Seguimiento de personas aplicando restricciones cinemáticas basadas en modelos de cuerpos rígidos articulados

Tesis Doctoral - Junio de 2017

Autor:
Enrique Martínez Bertí

Director:
Antonio José Sánchez Salmerón

Instituto de Automática e Informática Industrial

Universitat Politècnica de València

Doctorado en Automática, Robótica e Informática Industrial

El proyecto de tesis ha sido respaldado por el Programa VALi+d de La Generalitat Valenciana además de por el Plan Nacional de I+D, Comisión Interministerial de Ciencia y Tecnología (FEDER-CICYT) bajo el projecto DPI2013-44227-R.

El autor ha sido beneficiario de la beca de Formación de investigadores en fase pre-doctoral (ACIF/2012/043) por la Generalitat Valenciana. Además el trabajo de tesis ha sido respaldado por el departamento "Center for Research in Computer Vision" de la "University of Central Florida", Orlando (USA) gracias a la beca para estancias de becarios y contratados pre-doctorales en centros de investigación fuera de la Comunidad Valenciana (BEFPI/2014/062) de la Generalitat Valenciana, respaldando económicamente la estancia en la universidad de destino.

A mi familia.

Agradecimientos

Corría el año 2004 cuando pisé por primera vez la “Universitat Politècnica de València”. Atrás quedaban esos días de instituto con los amigos del pueblo, los de toda la vida. Empezaba una experiencia que jamás olvidare. Todo un mundo nuevo, mucha gente nueva que conocer, nuevas experiencias a descubrir. Aquel año, empecé a labrarme mi propio futuro. Por aquel entonces, nunca pensé que llegaría a estar donde estoy ahora, escribiendo estas palabras, escribiendo mi tesis doctoral. Si alguien me hubiera dicho aquel entonces que llegaría tan lejos, seguro que no le habría creído.

Pero sin embargo, hay unas personas que sí que creyeron en mi, mis padres. Esos padres que me han permitido llegar donde estoy ahora. Esos padres que me ayudaron a dar mis primeros pasos en la vida universitaria. Esos padres que sin pedirles nada, me han dado todo. Ellos me han visto caer una y otra vez y ellos me han hecho levantarme una y otra vez mas. De ellos nunca obtenía un “no” como respuesta, para ellos siempre era “lo que hagas, bien hecho esta, aquí estaremos nosotros para ayudarte”. Durante toda la vida ellos han sido mis pilares y gracias a ellos estoy aquí ahora. Ellos me han hecho crecer, ellos me han hecho la persona que soy ahora. Y nunca habrá suficientes palabras, para expresar todo lo que significan para mi. Esta tesis doctoral, va dedicada a ellos.

Por esto quiero agradecer a mis padres, Enrique Martínez Caballero y Concepción Bertí Lorente, todos aquellos consejos y todo el soporte que me han dado a lo largo de esta vida. Gracias a vosotros he llegado a donde estoy ahora, sin vosotros esto no hubiera sido posible. Gracias!

Hay otra persona en mi vida, una persona que me ha hecho llorar de alegría, sufrir en todas nuestras discusiones, reír en todos los buenos ratos que hemos pasado juntos,... Ella no puede ser otra persona que mi hermana, Esther Martínez Bertí. Miro hacia atrás y allá en el recuerdo quedan todas esas carreras donde me perseguías de arriba a abajo porque algo malo había hecho. Todos esos veranos "a la fresca" jugando a las cartas, al parchís o a lo que fuera. Esos días, nunca los olvido. Me has ayudado en cada uno de mis problemas, me has dado los mejores consejos de mi vida y también me has levantado cada vez que me has visto caer. Eres una de las responsables de ser lo que soy ahora, de ser como soy. Una de las responsables de las que este escribiendo estas líneas ahora. También gracias a ti he llegado donde estoy ahora, te estaré eternamente agradecido. No hay mejora hermana que uno pueda tener. Necesites lo que necesites, aquí estaré. Y aunque ha sido poco tiempo, no quiero olvidarme del nuevo miembro de la familia; gracias a ti, Vega, que desde el primer día ya me has robado el corazón. Has sido la distracción que necesitaba a lo largo de la redacción de la presente memoria de tesis. Espero que pronto puedas leer estas líneas, porque esto también va por ti. Gracias.

Durante toda la parte de investigación realizada durante la presente tesis, quiero dar las gracias a Antonio José Sánchez Salmerón, mi director de tesis. Él me ha enseñado a enfrentarme a cada uno de los problemas encontrados a lo largo de toda la tesis, y como no, a encontrar una solución. Te doy las gracias por todas esas horas que me has dedicado para corregir mi trabajo a lo largo de la tesis. Gracias por aguantarme en los momentos más difíciles y gracias por creer en mi y no perder nunca la paciencia conmigo. Gracias por ayudarme a crecer dentro de este mundo y gracias por tus críticas y tus consejos.

A lo largo de la vida universitaria, son dos las personas que también quiero agradecer el haber llegado donde estoy ahora, a mis compañeros Vicent Girbés Juan y Juan Ernesto Solanes Galbis. Gracias a los dos por todos estos años, por todos esos buenos ratos pasados juntos y por los no tan buenos. Gracias.

Resumen

La presente tesis trata sobre el estudio de técnicas de visión para la detección de la postura del esqueleto del cuerpo humano basada en el análisis de una sola imagen, además del seguimiento de estas posturas a lo largo de una secuencia de imágenes.

Se propone modelar la postura del esqueleto cuerpo humano mediante cuatro cadenas cinemáticas que modelan las cuatro extremidades articuladas. Estas cadenas cinemáticas y la cabeza permanecen unidas al cuerpo. Las cuatro cadenas cinemáticas se componen de tres puntos de interés. Por lo tanto, el modelo inicialmente dispone de un total de 14 puntos de interés.

Todos los métodos propuestos en este trabajo son implementados, validados y analizados utilizando la base de datos CAD60. La base de datos CAD60 es de dominio público y contiene imágenes en RGB y profundidad, junto con una base de datos creada expresamente para complementar a la base de datos CAD60.

En esta tesis se propone modificar la técnica denominada Deformable Parts Model (DPM), añadiendo el canal de profundidad denominado "Depth". Inicialmente el modelo DPM se definió sobre imágenes de tres canales RGB. Mientras que en esta tesis se propone trabajar sobre imágenes de cuatro canales RGBD, por ello a la ampliación propuesta se le denomina 4D-DPM. Los experimentos realizados con 4D-DPM demuestran una mejora en la precisión de la detección de la postura con respecto al modelo inicial DPM, a costa de incrementar su coste computacional al tratar un canal adicional.

Por otra parte, se propone reducir el coste computacional anterior simplificando el modelo que define la postura del cuerpo humano. La idea es reducir el número de variables a detectar con el modelo 4D-DPM, de tal manera que las variables suprimidas se puedan calcular a partir de las variables detectadas, utilizando modelos de cinemática inversa basados en cuaterniones duales. Los experimentos realizados demuestran que la combinación de estas dos técnicas permite, reduciendo el coste computacional del método original DPM, mejorar la precisión de la detección de postura debido a la información extra del canal de profundidad.

Adicionalmente, se propone utilizar modelos de filtros de partículas para continuar mejorando la precisión de la detección de las posturas humanas a lo largo de una secuencia de imágenes.

Atendiendo al problema de detección y seguimiento de las postura del esqueleto del cuerpo humano a lo largo de una secuencia de vídeo, esta tesis propone el uso del siguiente método.

1. Calibración de cámaras. Procesamiento de imágenes RGBD. Sustracción del fondo de la imagen con el método MSER.

2. 4D-DPM: método utilizado para detectar los puntos de interés (variables del modelo de postura) dentro de una imagen.

3. Filtros de partículas: se diseña este tipo de filtros para realizar el seguimiento de los puntos de interés a lo largo del tiempo y corregir los datos obtenidos por el sensor.

4. Modelado cinemático inverso: se realiza el control de cadenas cinemáticas con la ayuda de cuaterniones duales con el fin de obtener el modelo completo de la postura del esqueleto del cuerpo humano.

La contribución global de esta tesis es la propuesta del método anterior que, combinando los métodos anteriores, es capaz de mejorar la precisión en la detección y el seguimiento de la postura del esqueleto del cuerpo humano en una secuencia de vídeo, reduciendo además su coste computacional.

Esto es posible debido a la combinación del método 4D-DPM con la utilización de técnicas de cinemática inversa. El método original DPM debe detectar 14 puntos de interés sobre una imagen RGB para estimar la postura de un cuerpo humano. Sin embargo, el método propuesto, donde se elimina un punto de interés por cada extremidad, debe detectar 10 puntos de interés sobre una imagen RGBD. Posteriormente, los 4 puntos de interés eliminados se calculan mediante la utilización de métodos de cinemática inversa a partir de los 10 puntos de interés calculados.

Para resolver el problema de la cinemática inversa se propone utilizar cuaterniones duales para cada una de las 4 cadenas cinemáticas que modelan las extremidades del esqueleto del cuerpo humano.

El filtro de partículas se aplica sobre la secuencia temporal de los 10 puntos de interés del modelo de postura detectados a través del método 4D-DPM. Para diseñar estos filtros de partículas se propone añadir las siguientes restricciones para ponderar las partículas generadas:

1. Restricciones en los límites de articulaciones: la postura del esqueleto del cuerpo humano se modela con un conjunto de cadenas cinemáticas abiertas. De tal manera que los puntos de interés son las variables de articulación de cada una de las cadenas cinemáticas. Cada una de estas variables tiene un movimiento restringido en un rango determinado.

2. Restricciones de suavidad: se propone ponderar las partículas de manera inversamente proporcional a la distancia de la partícula generada con la solución en el instante de tiempo anterior.

3. Detección de colisiones: el modelado geométrico utilizado para modelar el esqueleto del cuerpo humano es un conjunto de poli-esferas ya que estas nos permiten realizar la detección de colisiones, entre los elementos del cuerpo, de manera muy eficiente. Se propone que el filtro de partículas no genere partículas en las cuales se produzcan colisiones imposibles entre estos elementos.

4. Proyección de las poli-esferas: se propone ponderar cada partícula generada de forma directamente proporcional al solapamiento de la proyección del modelo de las poli-esferas que define esta partícula con algún plano de la imagen RGBD.

Resum

La present tesi tracta sobre l'estudi de tècniques de visió per a la detecció de la postura de l'esquelet del cos humà basada en l'anàlisi d'una sola imatge, a més del seguiment d'estes postures al llarg d'una seqüència d'imatges.

Es proposa modelar la postura de l'esquelet del cos humà per mitjà de quatre cadenes cinemàtiques que modelen les quatre extremitats articulades. Estes cadenes cinemàtiques i el cap romanen unides al cos. Les quatre cadenes cinemàtiques es componen de tres punts d'interés. Per tant, el model inicialment disposa d'un total de 14 punts d'interés.

Tots els mètodes proposats en este treball són implementats, validats i analitzats utilitzant la base de dades CAD60. La base de dades CAD60 és de domini públic i conté imatges en RGB i profunditat, junt amb una base de dades creada expressament per a complementar a la base de dades CAD60.

En esta tesi es proposa modificar la tècnica denominada Deformable Parts Model (DPM) , afegint el canal de profunditat denominat "Depth". Inicialment el model DPM es va definir sobre imatges de tres canals RGB. Mentres que en esta tesi es proposa treballar sobre imatges de quatre canals RGBD, per això a l'ampliació proposada se la denomina 4D-DPM. Els experiments realitzats amb 4D-DPM demostren una millora en la precisió de la detecció de la postura respecte al model inicial DPM, a costa d'incrementar el seu cost computacional al tractar un canal addicional.

D'altra banda, es proposa reduir el cost computacional anterior simplificant el model que definix la postura del cos humà. La idea és reduir el nombre de varia-

bles a detectar amb el model 4D-DPM, de tal manera que les variables suprimides es puguen calcular a partir de les variables detectades, utilitzant models de cinemàtica inversa basats en quaternions duals. Els experiments realitzats demostren que la combinació d'estes dos tècniques permet, reduint el cost computacional del mètode original DPM, millorar la precisió de la detecció de la postura degut a la informació extra del canal de profunditat.

Addicionalment, es proposa utilitzar models de filtres de partícules per a continuar millorant la precisió de la detecció de les postures humanes al llarg d'una seqüència d'imatges.

Atenent al problema de detecció i seguiment de les postura de l'esquelet del cos humà al llarg d'una seqüència de vídeo, esta tesi proposa l'ús del següent mètode.

1. Calibratge de càmeres. Processament d'imatges RGBD. Sostracció del fons de la imatge amb el mètode MSER.

2. 4D-DPM: mètode utilitzat per a detectar els punts d'interés (variables del model de postura) dins d'una imatge.

3. Filtres de partícules: es dissenya este tipus de filtres per a realitzar el seguiment dels punts d'interés al llarg del temps i corregir les dades obtingudes pel sensor.

4. Modelatge cinemàtic invers: es realitza el control de cadenes cinemàtiques amb l'ajuda de quaternions duals a fi d'obtindre el model complet de l'esquelet del cos humà.

La contribució global d'esta tesi és la proposta del mètode anterior que, combinant els mètodes anteriors, és capaç de millorar la precisió en la detecció i el seguiment de la postura de l'esquelet del cos humà en una seqüència de vídeo, reduint a més el seu cost computacional.

Açò és possible a causa de la combinació del mètode 4D-DPM amb la utilització de tècniques de cinemàtica inversa. El mètode original DPM ha de detectar 14 punts d'interés sobre una imatge RGB per a estimar la postura d'un cos humà. No obstant això, el mètode proposat, on s'elimina un punt d'interés per cada extremitat, ha de detectar 10 punts d'interés sobre una imatge RGBD. Posteriorment, els 4 punts d'interés eliminats es calculen per mitjà de la utilització de mètodes de cinemàtica inversa a partir dels 10 punts d'interés calculats.

Per a resoldre el problema de la cinemàtica inversa es proposa utilitzar quaternions duals per a cada una de les 4 cadenes cinemàtiques que modelen les extremitats de l'esquelet del cos humà.

El filtre de partícules s'aplica sobre la seqüència temporal dels 10 punts d'interés del model de postura detectats a través del mètode 4D-DPM. Per a dissenyar estos filtres de partícules es proposa afegir les següents restriccions per a ponderar les partícules generades:

1. Restriccions en els límits d'articulacions: la postura de l'esquelet del cos humà es modela amb un conjunt de cadenes cinemàtiques obertes. De tal manera que els punts d'interés són les variables d'articulació de cada una de les cadenes cinemàtiques. Cada una d'estes variables té un moviment restringit en un rang determinat.

2. Restriccions de suavitat: es proposa ponderar les partícules de manera inversament proporcional a la distància de la partícula generada amb la solució en l'instant de temps anterior.

3. Detecció de col·lisions: el modelatge geomètric utilitzat per a modelar l'esquelet del cos humà és un conjunt de poli-esferes ja que estes ens permeten realitzar la detecció de col·lisions, entre els elements del cos, de manera molt eficient. Es proposa que el filtre de partícules no genere partícules en les quals es produïsquen col·lisions impossibles entre estos elements.

4. Projecció de les poli-esferes: es proposa ponderar cada partícula generada de forma directament proporcional al solapament de la projecció del model de les poli-esferes que definix esta partícula amb algun pla de la imatge RGBD.

Abstract

The present thesis deals with the study of vision techniques for the detection of human pose based on the analysis of a single image, as well as the tracking of these poses along a sequence of images.

It is proposed to model the human pose by four kinematic chains that model the four articulated extremities. These kinematic chains and head remain attached to the body. The four kinematic chains are composed by three keypoints. Therefore, the model initially has a total of 14 parts.

All methods proposed in this work are implemented, validated and analyzed using the public CAD60 dataset, dataset with images in RGB and depth, and other dataset created expressly to complement the CAD60 dataset.

In this thesis it is proposed to modify the technique called Deformable Parts Model (DPM), adding the depth channel. Initially, the DPM model was defined over three RGB channel images. While in this thesis it is proposed to work on images of four RGBD channels, so the proposed extension is called 4D-DPM. The experiments performed with 4D-DPM demonstrate an improvement in the accuracy of pose detection with respect to the initial DPM model, at the cost of increasing its computational cost when treating an additional channel.

On the other hand, it is defined to reduce the previous computational cost by simplifying the model that defines the human pose. The idea is to reduce the number of variables to be detected with the 4D-DPM model, so that the suppressed variables can be calculated from the detected variables using inverse kinematics models based on dual quaternions. The experiments show that the combination of these

two techniques allows, by reducing the computational cost of the original DPM method, to improve the accuracy of the pose detection due to the extra depth channel information.

In addition, it is proposed to use a particle filter models to continue improving the accuracy of detection of human poses along a sequence of images.

Considering the problem of detection and monitoring of human body pose along a video sequence, this thesis proposes the use of the following method.

1. Camara calibration. RGBD image processing. Subtraction of the image background with the MSER method.
2. 4D-DPM: method used to detect the keypoints (variables of the pose model) within an image.
3. Particle filters: this type of filter is designed to track the keypoints over time and correct the data obtained by the sensor.
4. Inverse kinematic modeling: the control of kinematic chains is performed with the help of dual cuaternions in order to obtain the complete pose model of the human body.

The overall contribution of this thesis is the proposal of the previous method that, combining the previous methods, is able to improve the accuracy in the detection and the follow up of the human body pose in a video sequence, also reducing its computational cost .

This is possible due to the combination of the 4D-DPM method with the use of inverse kinematics techniques. The original DPM method should detect 14 point of interest on an RGB image to estimate the human pose. However, the proposed method, where a point of interest for each limb is removed, must detect 10 point of interest on an RGBD image. Subsequently, the eliminated 4 point of interest are calculated by using inverse kinematics methods from the calculated 10 point of interest.

To solve the problem of inverse kinematics a dual quaternions methods is proposed for each of the 4 kinematic chains that model the extremities of the skeleton of the human body.

The particle filter is applied over the time sequence of the 10 points of interest of the posture model detected through the 4D-DPM method. To design these particle filters it is proposed to add the following restrictions to weight the particles generated:

1. Restrictions on joint limits: The human pose is modeled with a set of open kinematic chains. In such a way that the points of interest are the joint articulation variables of each of the kinematic chains. Each of these variables has a restricted movement in a given range.

2. Softness restrictions: it is proposed to weight the particles inversely proportional to the distance of the particle generated with the solution at the previous time instant.

3. Collision detection: the geometric modeling used to model the skeleton of the human body is a set of poly-spheres because they allow us to perform collision detection between body elements very efficiently. It is proposed that the particle filter does not generate particles in which there are collisions impossible between these elements.

4. Projection of poly-spheres: it is proposed to weight each particle generated directly proportional to the overlap of the projection of the poly-sphere model that defines this particle with some plane of the RGBD image.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1

Introducción

En este capítulo se presentan de forma concisa las ideas y objetivos principales del trabajo de tesis realizado.

1.1 Introducción.

En este capítulo se presentan de forma concisa las ideas y objetivos principales del trabajo de tesis realizado. En el apartado 1.2 se pretende dar la visión inicial del problema, a continuación en 1.3 se define el concepto de captura del movimiento humano. En 1.4 se presentan los objetivos planteados, finalmente en la sección 1.5 se introduce el esquema de este documento así como el esquema del algoritmo desarrollado.

1.2 Presentación

El avance de las tecnologías de la información y comunicación, la introducción de estas en todos los ámbitos de la vida cotidiana y el interés de las personas por interactuar con los dispositivos de forma eficiente y fiable obligan a los desarrolladores de nuevas máquinas a plantearse nuevas interfaces basadas en la voz y el comportamiento humano. El uso actual de los dispositivos electrónicos está orientado a botones o teclas (mandos a distancia de TV, microondas, móviles...), en los ordenadores personales es el ámbito donde existen las interfaces más sofisticadas: ratones, palanca de juego, volantes, ratones de bola, etc.

Existen otros medios de interacción de medio y alto coste, como es la localización por sistemas de infrarrojos, por ultrasonidos, por visión etc. Pero las interfaces mencionadas anteriormente se utilizan fundamentalmente en sistemas de realidad virtual avanzados, como visualizadores estereoscópicos, workbench, Cave3D, etc.

En esta tesis se intenta obtener un sistema automático de detección y seguimiento del esqueleto del cuerpo humano para realizar una estimación de la postura que permita definir nuevos métodos de interacción de forma natural con un robot, ordenador o con cualquier máquina.

Como objetivos primordiales o clave, se puede resaltar que el sistema sea no invasivo con el usuario, y que su funcionamiento sea lo más automático posible. Por ello se ha optado por las técnicas de visión y gráficos por ordenador avanzadas, dentro de esta las técnicas de análisis y síntesis de secuencias de imágenes 2D y modelos biomecánicos 3D. El sistema considera un proceso de análisis y síntesis de las secuencias de movimientos.

De lo anterior, se deducen las múltiples aplicaciones del sistema propuesto. Estas aplicaciones pueden ser: video-vigilancia, análisis del movimiento humano aplicado al deporte, aplicado a la rehabilitación, sistemas biométricos y obviamente cualquier interfaz que facilite la interacción hombre-máquina, conocido con las siglas IHM.

El análisis y síntesis del movimiento del esqueleto del cuerpo humano es un área que ha sido estudiada por múltiples culturas y civilizaciones remontándose sus primeros estudios a “Aristóteles” o “Leonardo Da Vinci”.

Centrándose en los primeros estudios cinematográficos se tiene que mencionar el gran trabajo de E.Muybridge a finales del siglo XIX. Donde se recopilan las primeras secuencias de personas y animales en movimiento sobre una película. A continuación, se define el concepto de captura del movimiento humano.

1.3 Definición

El ser humano posee una estructura ósea que le habilita para realizar una gran libertad de movimientos, esta libertad de movimientos le permite desplazarse y cambiar la postura (andar, correr, agacharse, saltar...), realizar acciones con la intención de manipular su entorno (empujar, tirar, recoger objetos, lanzarlos,...) e interactuar con otras personas (dar la mano, abrazarse...), a veces con malos fines (robar, agredir...). La captura del movimiento y de la acción que está llevando

a cabo una persona puede tener varias aplicaciones, cada aplicación tiene como interés un nivel de detalle distinto.

Resulta obvio que para permitir la interacción con todo el cuerpo es necesario obtener información del mismo, esta captura de información se puede llevar a cabo mediante diferentes medios: ultrasonidos, imágenes en color, imágenes térmicas, rayos X, sensores electromagnéticos, etc. Estos medios pueden complementarse con marcas situadas en la persona para ayudar al sistema a capturar o reconstruir la información (sistemas de captura con marcadores), siempre será más cómodo para la persona la no utilización de marcadores. Al colocar marcadores para medir se está alterando la medición, colocando más peso e incomodando, lo que supone una alteración del movimiento que realizará la persona. Otras veces resulta imposible colocar marcadores, por ejemplo en el estudio de competidores en el ámbito deportivo. Por el contrario, la no utilización de marcadores dificulta el procesamiento de la información, siendo necesarios algoritmos más robustos y costosos computacionalmente.

Un objetivo del análisis del movimiento humano puede ser la vídeo vigilancia, detectar las personas que hay en el entorno y cómo interactúan entre ellas, en este caso es de interés la situación de las personas y las interacciones anormales que puedan existir, la detección de un robo o de una situación anormal se realiza en un nivel de abstracción alto, siendo a veces más difícil definir la acción que reconocerla. Otro posible objetivo es la detección y seguimiento de las personas para realizar mediciones de tráfico, en este caso la interpretación es a un nivel de abstracción menor, solo un seguimiento.

Una aplicación más es el estudio biomecánico, en este tipo de aplicaciones lo importante es la obtención de las posiciones de las articulaciones con gran precisión. Los marcadores son de gran ayuda ya que proporcionan una gran precisión llegándose incluso a calcular en tiempo real la translación que sufre la rodilla al correr (distancia del fémur a la tibia).

Dentro de todo el rango de posibles aplicaciones derivadas de la captura del movimiento humano el interés de esta tesis se centra en el seguimiento del esqueleto humano y la interacción hombre-máquina (IHM), para llevar a cabo la interacción hombre-máquina nos centraremos en los movimientos y gestos que realiza el usuario. Los datos necesarios para esta aplicación son las posiciones y orientaciones de las articulaciones, la interpretación que se realiza sobre estos movimientos son las órdenes dadas a la máquina. La interacción, como su propio nombre indica supone una respuesta por parte de la máquina tras una petición de el/los usuario/s.

Restringiendo el concepto a la aplicación se define captura del movimiento humano como el proceso de posicionamiento y orientación a lo largo del tiempo de las articulaciones de una persona, esta persona será detectada inicialmente antes de empezar a realizar la captura del movimiento. No se entenderá como captura del movimiento humano localizar por primera vez la persona, pero si su seguimiento a lo largo del tiempo, la información resultante es la posición/configuración 3D de la persona a lo largo del tiempo. La captura del movimiento no incluye la interpretación de la acción, sino que sirve como fuente de datos de esta, que trabaja a un nivel de abstracción más alto. Los objetivos de este trabajo vienen detallados en el siguiente apartado.

1.4 Objetivos

El cuerpo humano es un sistema muy complejo con gran cantidad de grados de libertad y múltiples oclusiones en sus movimientos. El objetivo global del análisis del movimiento y su síntesis posterior se convierte a su vez en una tarea muy compleja. Por tanto para que el sistema sea manejable tanto en complejidad como en eficiencia es lógico introducir ciertas restricciones funcionales. Estas restricciones se introducen con la idea de realizar el sistema lo más robusto, fiable y económico posible. Las principales hipótesis de trabajo son:

- El sistema será no invasivo. Se utilizarán técnicas de visión por ordenador mediante el uso de cámaras digitales sobre imágenes RGB o de profundidad.
- El entorno de trabajo será flexible, aunque se consideran fundamentalmente los entornos cerrados, donde existe cierto control en la iluminación (oficinas, laboratorios, edificios, etc.).
- En la medida de lo posible, se intentará que los sistemas de captura y digitalización sean de medio y bajo coste, pensando en la aplicación futura a entornos domóticos o de realidad virtual doméstica.
- Se definirá y aplicará un modelo geométrico de la persona sujeta a estudio para validar los resultados.

El objetivo general de la tesis pretende desarrollar mejoras para realizar la detección y el seguimiento del esqueleto del cuerpo humano dentro de una imagen, desarrollando filtros de seguimiento de los puntos de interés e introducirles restricciones gracias al diseño de modelos geométricos. Los modelos geométricos servirán además para la representación de la solución aportada desarrollando el modelo cinemático del cuerpo humano.

Teniendo las hipótesis de trabajo presentes se definirán los objetivos básicos que debe cumplir nuestro sistema. De manera sintética se pueden resumir los siguientes objetivos primordiales de la tesis doctoral presentada:

- Realizar la detección del esqueleto del cuerpo humano con una sola imagen sin utilizar marcadores.
- Definir de forma precisa y robusta un módulo de seguimiento y estimación de la postura mediante el filtro de partículas. Realizando un estudio exhaustivo de búsqueda de la mejor solución y la consideración de ciertas restricciones.
- Diseñar un modelo geométrico de la persona para realizar una reconstrucción 3D del esqueleto del sujeto a estudio. Los parámetros de este modelo deben estimarse de forma automática.
- Aplicar modelos geométricos que faciliten la detección de colisiones.
- Desarrollar un modelo cinemático y aplicar las técnicas de cinemática directa e inversa necesarias para conseguir el seguimiento y reconstrucción del esqueleto del cuerpo humano.
- Evaluar cada una de las mejoras introducidas al modelo para corroborar su utilización.

Estas aportaciones facilitarán el desarrollo de un nuevo paradigma de interacción, de manera que la persona pueda actuar y comunicarse con una máquina de una forma más natural y eficiente. Se pretende avanzar del concepto ya convencional de sistema de interfaz gráfico hacia un sistema perceptual o multimodal, más habitual entre los seres humanos.

1.5 Estructura

Las aportaciones de esta tesis son una serie de pasos que van a ser definidos en los capítulos posteriores. Observando la figura 1.1, uno se hace la idea del método propuesto para dar solución al problema de detección y seguimiento del esqueleto del cuerpo humano.

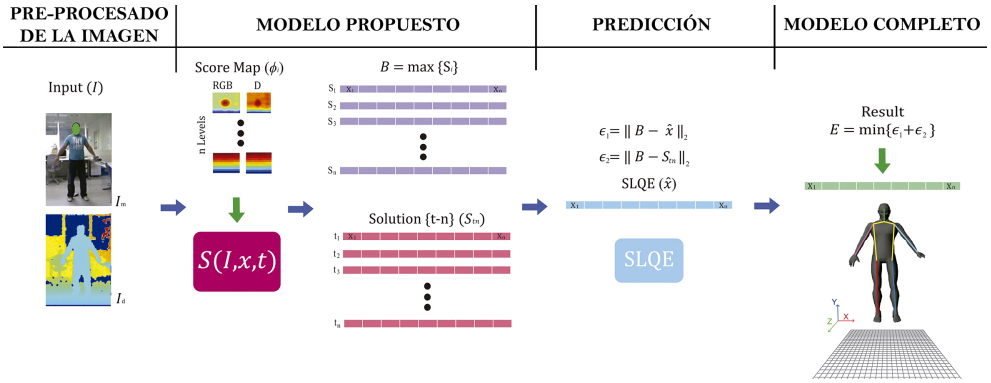


Figura 1.1: Esquema del método propuesto.

Atendiendo a la estructura de la figura 1.1, se definen cada uno de los capítulos de la presente tesis.

Previamente a la estructura anterior de la figura 1.1, se realiza el estudio del arte en el capítulo 2. El primer paso en el modelo descrito en la presente tesis consiste en el "pre-procesado de la imagen", donde se realiza al sustracción del fondo tanto de la imagen RGB como de la imagen de profundidad. Estas imágenes serán las que se utilicen como imágenes de entrada al "modelo propuesto". Tanto el "pre-procesado" como el "modelo propuesto" se definen en el capítulo 3. Para mejorar la precisión de los resultados, en el capítulo 4 se define el método de "predicción" donde se utiliza la información temporal para mejorar los resultados del modelo. Una vez se tienen los puntos de interés necesarios dentro de la imagen, se obtiene el "modelo completo" del esqueleto del cuerpo humano, donde en el capítulo 5 se define el modelo geométrico y cinemático utilizado para llegar a tal fin. Los resultados obtenidos son expuestos y analizados en el capítulo 6. Y en el capítulo 7 se detallan las conclusiones a las que se llegan tras analizar con detalle los resultados.

En la parte de anexos se encuentra: en el anexo A se hace una ampliación estudio del arte anterior a la tesis presentada. En el anexo B se habla sobre el modelo original DPM en el que se ha basado la presente tesis. El anexo C define el filtro de partículas utilizado. En el anexo D se definen los cálculos previos para dar solución a la restricción 3 introducida en el filtro de partículas. En el anexo E se definen los cálculos previos necesarios a utilizar posteriormente para llegar a la obtención de la solución cinemática utilizando cuaterniones duales.

Capítulo 2

Estado del Arte

En este capítulo se realiza el estudio del arte relacionado con la detección de la postura del esqueleto del cuerpo humano dentro de una imagen.

2.1 Estado del arte actual

2.1.1 Introducción

En Visión por Computador, los seres humanos son típicamente considerados como objetos articulados consistiendo en partes móviles rígidamente conectadas entre sí en ciertos puntos de articulación. Bajo este supuesto, la estimación de la postura del esqueleto del cuerpo humano a partir de imágenes monoculares tiene como objetivo recuperar la distribución representativa de las partes del cuerpo a partir de las características de la imagen. Tal y como se ha comentado en la introducción, algunas de las aplicaciones donde se utilizan las posturas del esqueleto del cuerpo humano extraídas a partir de imágenes pueden ser: analizar comportamientos humanos en sistemas de vigilancia inteligentes, controlar el movimiento del avatar en animaciones realistas, analizar la patología de la marcha en las prácticas médicas, interactuar con las computadoras,...

Tradicionalmente, una postura del esqueleto del cuerpo humano puede ser reconstruida con precisión a partir del movimiento capturado con marcadores ópticos unidos a las partes del cuerpo Cheng y Trivedi 2004. Estos sistemas basados en

marcadores suelen usar múltiples cámaras para capturar movimientos simultáneamente. Sin embargo, no son adecuados para aplicaciones no invasivas de la vida real, y el equipo es bastante caro, confinando sus aplicaciones a experimentos de laboratorio o a producciones costosas a largo plazo, como controlar los movimientos de avatares en animaciones Dinh y col. 2014.

Por lo tanto, un número creciente de estudios se han centrado en métodos sin marcadores. Las entradas del cuerpo humano a la escena también son capturadas por las cámaras, pero los seres humanos que actúan no están obligados a usar ningún marcador. Se pueden capturar varios tipos de imágenes: imágenes en RGB o en escala de grises, imágenes de infrarrojos Hirota, Nakajima y Saito 2003, imágenes de profundidad Buys y col. 2014 y otras. Las imágenes RGB captan la luz visible y son las imágenes más frecuentes en la web. Las imágenes infrarrojas captan la luz infrarroja, y las imágenes en profundidad contienen información sobre la distancia de los objetos de la imagen a las cámaras. Las imágenes infrarrojas son extremadamente útiles para la visión nocturna.

Mientras que las cámaras normales pueden capturar imágenes RGB, las imágenes en profundidad requieren equipo especializado. Los productos comerciales incluyen "Microsoft Kinect" "Meet Kinect forWindows." 2016, "Leap Motion" "Leap Motion" 2016, y "GestureTek" "GestureTek" 2016. Estos productos proporcionan interfaces de programación de aplicaciones (API) para la adquisición de datos de profundidad Oleinikov y col. 2014. La utilización de imágenes a color junto con imágenes de profundidad para la detección de la postura del esqueleto del cuerpo humano ha sido la solución más exitosa: la estimación en tiempo real de las articulaciones del esqueleto del cuerpo humano en 3D y el etiquetado de partes del cuerpo en píxeles (pixelwise) han sido posibles basándose en "bosques de decisión aleatorios" (randomized decision forests) Shotton y col. 2013b. La exactitud de la estimación de las imágenes de profundidad es comparativamente más precisa, pero estos dispositivos sólo pueden adquirir imágenes dentro de un determinado límite de distancia (unos ocho metros), y una gran mayoría de imágenes en la web son imágenes en RGB o imágenes en escala de grises sin información de profundidad.

La detección de la postura del esqueleto del cuerpo humano a partir de una sola imagen es un problema donde se aplican muchas restricciones a los movimientos del esqueleto del cuerpo humano debido a la naturaleza intrínseca de este problema. Una postura produce varias siluetas dentro de una imagen cuando se proyecta desde puntos de vista diferentes. Este problema ha sido ampliamente estudiado, pero todavía está lejos de ser resuelto completamente. Las soluciones eficaces para este problema necesitan abordar cambios de iluminación, problemas de sombreado y variaciones de puntos de vista. Además, los problemas de

estimación de la postura del esqueleto del cuerpo humano tienen características específicas. En primer lugar, el cuerpo humano tiene un número alto de grados de libertad, dando lugar a una solución de espacio dimensional muy elevada. En segundo lugar, la estructura compleja y la flexibilidad compleja de las partes del cuerpo humano hacen que las posturas del esqueleto del cuerpo humano parcialmente ocluidas sean extremadamente difíciles de reconocer. En tercer lugar, la pérdida de profundidad resultante de las proyecciones 3D en planos de imagen 2D hace que la estimación de las posturas en 3D sea extremadamente difícil.

2.1.1.1 Trabajos relacionados

En la literatura se pueden encontrar varias estudios sobre la estimación de las posturas humanas. Los autores de Krotosky y Trivedi 2004; Moeslund, Hilton y Kruger 2006; Poppe 2007; Li y Sun 2009 dan estudios de la visión basada en la estimación de la postura del esqueleto del cuerpo humano, pero estos trabajos se realizaron antes de 2009. Un estudio más reciente es Liu y col. 2015. Este estudio se centra en la estimación de la postura del esqueleto del cuerpo humano a partir de varios tipos de imágenes de entrada bajo diversos tipos de ajustes de cámara (tanto de vista única como de vista múltiple). Ver anexo A.

Otros estudios recientes se realizaron sobre metodologías específicas. Por ejemplo, el estudio de Lepetit y Fua 2005 y el estudio de Perez-Sala y col. 2014, estáns basados en modelos de estudio, que emplean el conocimiento del cuerpo humano, como la apariencia y la estructura del cuerpo humano para la mejora de la estimación de la postura. También hay estudios dedicados al análisis del movimiento humano donde la información del movimiento es un requisito previo Moeslund, Hilton y Kruger 2006; Poppe 2007; Zhu y col. 2006; Lepetit y Fua 2005.

Un área que está estrechamente relacionada con la estimación de la postura del esqueleto del cuerpo humano es el reconocimiento de la acción. Aunque los algoritmos y técnicas utilizadas en el reconocimiento de la acción humana son diferentes de los utilizados de la estimación de la postura del esqueleto del cuerpo humano, los resultados de reconocimiento de estos dos a veces se combinan, Yao y col. 2011; Yao, Gall y Van Gool 2012; Nie, Xiong y Zhu 2015; Gong, Gonzalez y Roca 2012. Los estudios sobre el reconocimiento de la acción incluyen Poppe 2010; Weinland, Ronfard y Boyer 2011; Aggarwal y Ryoo 2011; Chen, Wei y Ferryman 2013.

2.1.2 Características

Teniendo en cuenta las imágenes monoculares y de profundidad, el primer paso en la secuencia de procesos es extraer los puntos clave, describirlos y alimentar a la siguiente unidad de procesamiento. El rendimiento de varias características debe ser evaluado para determinar qué característica elegir dentro de un determinado contexto.

Los puntos de características (feature points) extraen la mayor parte de la información representativa en imágenes, pero son generalmente ruidosos y contienen información redundante. Estas características se describen a continuación.

2.1.2.1 Características de bajo nivel

Para capturar la apariencia, la geometría y la forma como información de las partes del cuerpo humano, las características comúnmente extraídas son siluetas "Inferring Body Pose without Tracking Body Parts." 2016; Agarwal y Triggs 2006c; Gavrila 2007; Elgammal y Lee 2014, contornos Viola, Jones y Snow 2005; Sapp, Toshev y Taskar 2010, bordes Dimitrijevic, Lepetit y Fua 2006; Weinrich, Volkhardt y Gross 2013, etc. Las siluetas (Silhouettes) extraen contornos de los objetos y estos son invariantes a la textura e iluminación Agarwal y Triggs 2006c; Sminchisescu y col. 2005; Grauman, Shakhnarovich y Darrell 2003; Kehl, Bray y VanGool 2005; Sminchisescu y Telea 2002. Los contornos (Contour) capturan el contorno (outline) de las partes del cuerpo y es un camino con bordes que unen los puntos de intersección de los límites de segmentación Sapp, Toshev y Taskar 2010. Los bordes extraen líneas muy variadas en las imágenes y normalmente se calculan por convolución.

En comparación, las siluetas son descriptores globales que incluyen una vista general de un objeto y usualmente requieren conocimientos previos del fondo para extraer el objeto en primer plano, Bosch y col. 2014; Bosch, Sanchez y Ricolfe-Viala 2012. Los contornos requieren un procesamiento previo (como la segmentación) y adjuntan detalles además de la información fuera de línea (outline information). Los bordes son características bastante dispersas y se pueden calcular directamente con filtros. El color, Sapp, Toshev y Taskar 2010; Wren y col. 1997; Kakadiaris y Metaxas 1995, y la textura, Felzenszwalb y col. 2010, son otras de las características que modelan la apariencia del cuerpo humano.

2.1.2.2 Características de nivel medio

Las características de la silueta extraída se codifican generalmente como descriptores de "Fourier" Zahn y Roskies 1972, contextos de la forma Mori, Belongie y Malik 2005, firmas geométricas Arkin y col. 1991, características de "Poisson" Gorelick y col. 2006, etc. El descriptor de contexto de forma más utilizado captura la distribución de los puntos relativos al punto actual que se describe.

Otras características basadas en bordes o gradientes se codifican como histogramas de gradientes orientados (HOG) Dalal y Triggs 2005a; Zhu y col. 2006; Shakhnarovich, Viola y Darrell 2003; Nayak, Sarkar y Loeding 2009, distribución de bordes relacionales Nayak, Sarkar y Loeding 2009, "transformación de características invariantes en escala" (SIFT) Lowe 1999; Lowe 2004, características "SIFT-like" Agarwal y Triggs 2006a; Scovanner, Ali y Shah 2007, características de bordes Wu, Lin y Huang 2005, características de forma Sabzmeydani y Mori 2007, etc. Mediante la medición en una serie de escalas, las características SIFT se pueden comparar con la varianza de escala. Las características HOG son características muy populares para la estimación de la postura del esqueleto del cuerpo humano, y normalmente se aprenden varias plantillas HOG (HOG templates) que representan varios estados de una parte del cuerpo.

Además de las características locales mencionadas anteriormente, existen muchas características globales que capturan características generales, por ejemplo, el mapa del primer plano del objeto Jiang 2011 y las características de la rejilla densa (dense grid features), como las rejillas de los descriptores HOG Dalal y Triggs 2005a o las rejillas de las características SIFT Agarwal y Triggs 2006a; Ionescu, Li y Sminchisescu 2011.

Las codificaciones jerárquicas multinivel, como el "Modelo Jerárquico y X" (HMAX) Serre, Wolf y Poggio 2005, las hiperfunciones Agarwal y Triggs 2006b, la pirámide espacial Lazebnik, Schmid y Ponce 2006, el árbol del vocabulario y los "Bloques Espaciales Multinivel" (MSB) Kanaujia, Sminchisescu y Metaxas 2007 son más estables en la preservación de la invariancia a las transformaciones geométricas. Otras características, tales como caminos locales Hara y Kurokawa 2011, secuencia de procesos de predicción Lallemand, Szczot e Ilic 2014 y "Extremal Human Curves" Slama, Wannous y Daoudi 2013 son también rasgos comunes en la estimación de la postura del esqueleto del cuerpo humano.

Una "red neuronal convolucional" (CNN, o ConvNet) es actualmente la característica más popular en visión por computador, inteligencia artificial, aprendizaje de máquina (machine learning), y en muchos otros campos. CNN es una extensión de una red neuronal. Las imágenes de entrada son procesadas por convolución y

muestreadas varias veces para extraer características. Los errores estimados son propagados hacia atrás y los parámetros de red se ajustan en consecuencia. Recientemente, algunos trabajos han utilizado características CNN extraídas para la estimación de la postura del esqueleto del cuerpo humano Li y Chan 2014; Li, Liu y Chan 2014; Pfister, Charles y Zisserman 2015.

2.1.2.3 Características de alto nivel

Varios descriptores tienen características de alto nivel, como "parches de partes del cuerpo" (body part patches), descriptores de geometría o características de contexto. Los parches de la parte del cuerpo asumen cualquiera de la orientación espaciada, y pueden tener cualquier posición dentro del parche. Son descriptores más generales en comparación con las partes del cuerpo, que se confinan dentro de una extremidad del cuerpo, entre las articulaciones del cuerpo, o en las proximidades de una articulación del cuerpo. Las partes del cuerpo combinadas, como un descriptor de geometría, contienen relaciones semánticas entre partes individuales, Roberts, McKenna y Ricketts 2004; Bourdev y col. 2010; Gkioxari y col. 2014b, usualmente codificadas como poner dos conjuntos de características juntas, incluyendo la ubicación y orientación de las partes del cuerpo Sapp, Toshev y Taskar 2010. El contexto, por otro lado, captura las correlaciones espaciales o temporales, y puede representar características específicas de la tarea Oleinikov y col. 2014. Las funciones de alto nivel codifican la co-ocurrencia semántica entre las unidades de composición. En comparación con las características de nivel medio, que son una codificación espacial o temporal en un patrón predefinido, las correlaciones de alto nivel extraen características de los datos de entrenamiento y dejan que los datos hablan por sí mismos.

2.1.2.4 Características de movimiento

Como se mencionó anteriormente, las posturas estimadas a partir de imágenes monoculares podrían utilizarse como una inicialización para el seguimiento de la postura en sistemas de vigilancia inteligentes. La consistencia temporal y espacial en videos podría ser extremadamente útil. Por ejemplo, puede utilizarse para corregir el fallo de la estimación en una sola imagen. A continuación, se revisa las señales de movimiento utilizadas por la estimación de la postura humana.

Las características de movimiento como el flujo óptico denso Zuffi, Freifeld y Black 2012, el flujo óptico robusto Lu y Jiang 2013, los límites de energía y movimiento de los bordes y sus combinaciones Sminchisescu y Triggs 2001 mejoran el desempeño de la estimación por correspondencia temporal. El flujo óptico Dalal, Triggs

y Schmid 2006 es el patrón de los movimientos de objetos, superficies y bordes causados por el movimiento relativo entre un observador y la escena. El gradiente en el flujo óptico está relacionado con los movimientos, y podría utilizarse para rastrear las posturas Bregler y Malik 1998; "Scene Constraints-aided Tracking of Human Body" 2016. También se utilizan características que representan similitudes de movimiento locales, tales como Wang y col. 2010; Sun y col. 2014 y parches de movimiento y apariencia basados en la diferencia de imagen Yang y Bissacco 2010.

Las características individuales son insensibles a las variaciones de fondo, lo que resulta en ambigüedades. Las características se pueden combinar para mejorar el rendimiento de la estimación de la postura del esqueleto del cuerpo humano Sedai y col. 2010; Sedai, Bennamoun y Huynh 2013. Las posturas del esqueleto del cuerpo humano en las imágenes monoculares se podrían estimar más exactamente combinando múltiples señales de imagen con diferentes rasgos, tales como "edge cue", "ridge cue" y "motion cue" Sidenbladh y Black 2001.

Por otra parte, utilizando secuencias continuas de imágenes de profundidad: en Jalala y col. 2017 se segmentan la siluetas del cuerpo humano en las imágenes de profundidad utilizando características temporales del movimiento humano, así como se obtienen las articulaciones del esqueleto humano utilizando características espacio-temporales del cuerpo humano. En Chen, Liu y Kehtarnavaz 2016 se utilizan "mapas de movimiento de profundidad" (DMMs) donde cada "frame" de profundidad en una secuencia de video es proyectado en tres planos ortogonales cartesianos, bajo cada vista de proyección, la diferencia absoluta entre dos mapas proyectados consecutivos se acumula a través de una secuencia de vídeo de profundidad total que forma un DMM.

2.1.3 Modelos del cuerpo humano

Una de las cuestiones clave en la estimación de la postura del esqueleto del cuerpo humano es cómo construir y describir modelos del cuerpo humano. Un cuerpo humano incluye la información de la estructura cinemática, la información de la forma y la información de la textura del cuerpo humano, si es posible. Por ejemplo, un modelo de articulación cinemática de alrededor de 30 parámetros de las articulaciones y 8 parámetros de proporción interna que codifican las posiciones de las articulaciones de la cadera, la clavícula y el cráneo y la forma del cuerpo humano se pueden denotar como 9 parámetros de forma deformables para cada parte del cuerpo, recogidos en un vector Sminchisescu y Telea 2002. En los métodos discriminativos, los modelos cinemáticos se utilizan para ensamblar las partes del cuerpo detectadas por separado o las articulaciones del cuerpo. Bajo

proyecciones geométricas, estos modelos con una postura pueden ser mapeados a un plano, y así comparar con la evidencia de la imagen para verificar la postura proyectada.

La configuración de una postura del esqueleto del cuerpo humano puede determinarse por la orientación de la parte del cuerpo. Una línea es capaz de especificar una orientación del miembro, por lo que un cuerpo humano puede ser modelado como una figura de líneas. Los volúmenes de las partes del cuerpo juegan un papel importante en la localización cuando el modelo humano volumétrico necesita ser proyectado en un plano de imagen 2D donde la efectividad de la postura es validada comparando con la evidencia de la imagen. En las secciones siguientes, discutimos varios tipos de modelos de cuerpo humano.

2.1.3.1 Modelo cinemático

Modelos que siguen la estructura del esqueleto se llaman modelos de cadena cinemática Lehrmann, Gehler y Nowozin 2013. El conjunto de posiciones de las articulaciones y las orientaciones de los miembros son representaciones efectivas de una postura humana. En Akhter y Black 2015 se presenta una representación libre de coordenadas: las coordenadas locales de la parte superior de los brazos, la parte superior de las piernas y la cabeza se pueden convertir en coordenadas esféricas, y se pueden definir los ángulos azimutales y polares discretizados de los huesos. El modelo cinemático nos permite incorporar creencias previas sobre ángulos articulares. Para lograr esto, un conjunto de los ángulos de las variables de articulación debe ser etiquetado con ejemplos positivos y negativos de la postura del esqueleto del cuerpo humano Demirdjian, Ko y Darrell 2003a.

Hay dos categorías del modelo cinemático. Uno es el modelo predefinido y el otro es la estructura del gráfico aprendido. Un modelo de gráfico muy popular es el "modelo de estructura pictórica" (PSM) Zuffi, Freifeld y Black 2012; Johnson y Everingham 2010. Un caso especial de PSM son los modelos estructurados en árbol. Gracias a sus soluciones únicas, los modelos de árboles estructurados se aplican con éxito en la estimación de la postura del esqueleto del cuerpo humano, ya sea en 2D o 3D Felzenszwalb y col. 2010; Felzenszwalb, McAllester y Ramanan 2008; Felzenszwalb y Huttenlocher 2005a; Andriluka, Roth y Schiele 2009; Andriluka, Roth y Schiele 2010; Sapp, Jordan y Taskar 2010; Yang y Ramanan 2011; Chen y Yuille 2015. Sin embargo, la inferencia es incapaz de capturar dependencias adicionales entre las partes del cuerpo, aparte de restricciones cinemáticas entre las partes conectadas. Por ejemplo, un modelo de árbol cinemático tiene sus limitaciones en representar equilibrio global y restricciones de gravedad. Además,

las partes del cuerpo no pueden detectarse completamente bajo la circunstancia de oclusión parcial Ramakrishna, Kanade y Sheikh 2013.

Muchos investigadores buscan una mejora de los modelos de árboles estructurados Sapp, Toshev y Taskar 2010; Wang y Mori 2008; Johnson y Everingham 2011; Tian, Zitnick y Narasimhan 2012; Duan, Batra y Crandall 2012; Sun y Savarese 2011; Xiao, Lu y Li 2012. Por ejemplo, los autores Wang y Mori 2008 resuelven la falta de descripción del modelo mediante la adición de modelos estructurados en árbol con diferentes formas, los autores de Johnson y Everingham 2011 añaden la restricción espacial de partes del cuerpo no conectadas al cambiar la función objetivo optimizada. Los autores de Sapp y Taskar 2013 mejoran la capacidad descriptiva mediante la adición de los estados de los modelos. Los autores de Wang y Mori 2008 utilizan múltiples modelos de árbol en lugar de un solo modelo de árbol para la estimación de postura del esqueleto del cuerpo humano. Los parámetros de cada modelo de árbol individual son entrenados a través de algoritmos de aprendizaje estándar en un solo modelo estructurado en árbol. Otro ejemplo de utilizar múltiples estructuras de árbol es Komodakis, Paragios y Tziritas 2011, donde se combinan diferentes modelos de árbol.

Más generales que los modelos de estructura predefinidos, los pares de relaciones de las partes del cuerpo pueden ser entrenadas a partir de las imágenes Chen y Yuille 2014. Adicionalmente, una estructura de árbol basada en redes bayesianas podría ser aprendida Lehrmann, Gehler y Nowozin 2013; Tashiro y col. 2014. Estos modelos son no-paramétricos con respecto a la estimación tanto de su estructura gráfica como de sus distribuciones locales.

Otra forma de utilizar los modelos cinemáticos es para corroborar que los puntos de características extraídos de una imagen de profundidad, correspondientes a la postura del esqueleto del cuerpo humano, son correctos. Estos modelos cinemáticos tratan al cuerpo humano como si de un conjunto de cadenas cinemáticas se tratase y utilizando la cinemática inversa se puede obtener el modelo 3D y posteriormente proyectar este modelo 3D sobre la imagen RGB y corroborar el solapamiento de la postura con la imágen 3D. Los modelos cinématicos que se pueden utilizar para llegar a tal fin son por ejemplo: métodos geométricos, Denavit-Hartenberg (DH), quaterniones simples (SQ), quaterniones duales (DQ), entre otros.

2.1.3.2 Modelo planar

Aparte de capturar las relaciones de conexión entre las partes del cuerpo, los modelos planares son también capaces de aprender la apariencia. Varios medios se utilizan para aprender la forma y la apariencia de las partes del cuerpo humano. Un ejemplo es "Active Shape Models" (ASMs). ASMs se utilizan para representar el cuerpo humano completo y capturar las estadísticas de las deformaciones de contorno de una forma media mediante el análisis de componentes principales (PCA) Cootes y col. 1995; Freifeld y col. 2010; Baumberg y Hogg 1994; Urtasun y Fua 2004a.

Otro ejemplo es el modelo de cartón (cardboard model) basado en formas rectangulares, compuesto de información sobre los colores del primer plano del objeto y las formas rectangulares de la parte del cuerpo. El modelo de cartón normalmente tiene un torso y ocho medias extremidades, la apariencia de cada parte del cuerpo está representada por el color RGB promedio, y el histograma de color de primer plano también se almacena. Por ejemplo, los autores de Jiang 2010 utilizaron el modelo de cartón para la estimación de la postura humana.

2.1.3.3 Modelo volumétrico

Los modelos volumétricos representan realistamente formas y posturas del cuerpo 3D. Las formas geométricas y las mallas son ambos modelos volumétricos eficaces. Cuando se usan formas geométricas como componentes del modelo, las partes del cuerpo humano se aproximan con cilindros, cónicas y otras formas, ensamblando los miembros del cuerpo. Por ejemplo, una persona podría ser modelada como un compuesto de cilindros, con cada cilindro conectado a uno o varios otros cilindros Sidenbladh, Torre y Black 2000. Cada unión de los cilindros tiene de 1 a 3 grados de libertad (DOF). El modelo es descrito por la translación y rotación global. El patrón de la extremidad se extrae de los parámetros del modelo, y el espacio superficial se puede determinar resolviendo el problema de mínimos cuadrados Gall y col. 2009. Las secciones cónicas también se utilizan para modelar formas de miembros humanos en 3D. Las secciones cilíndricas y cónicas conducen a formas proyectadas rectangulares o cuadriláteras. Tales modelos captan claramente la verdadera forma de las extremidades humanas dadas las variaciones anchas en la anatomía o la ropa, y son más exactas que las aproximaciones basadas en estructuras pictóricas.

Otra forma de modelar un cuerpo humano volumétrico es utilizando mallas. Las mallas son modelos deformables y triangulados, por lo que son más adecuados para la representación de cuerpos humanos no rígidos De Aguiar y col. 2007. Una

forma de adquirir modelos de malla es a través de escaneos 3D Allen, Curless y Popovic 2002a; Park y Hodgins 2006; Sand, McMillan y Popovic 2003. Para estimar las ubicaciones de las articulaciones, las mallas suelen estar segmentadas en varias partes del cuerpo. Un modelo de malla 3D ampliamente utilizado es "Shape Completion and Animation of People" (SCAPE) Anguelov y col. 2005; Peng y col. 2009; Balan y col. 2007a; Ge y Fan 2015a; Ge y Fan 2015b. Modelos de títeres cosidos Zuffi y Black 2015 mejoran el modelo SCAPE mediante la adición de potenciales por parejas. Definen un coste de costura (stitching cost) para separar las extremidades, y aprenden las relaciones de parejas (pairwise) de las imágenes.

Otra forma de modelar un cuerpo humano volumétrico es mediante poli-esferas. La combinación de un número dado de esferas a lo largo del esqueleto del cuerpo humano da como solución un método llamado poli-esferas. El método utilizado tendrá como esferas principales cada una de las partes definidas dentro de la imagen que se van a utilizar para estimar la postura del esqueleto del cuerpo humano, coincidiendo generalmente con las articulaciones. Posteriormente, y utilizando estas esferas como principales, se generan otras esferas necesarias para envolver cada parte del cuerpo humano. Las poli-esferas además tienen la ventaja de poder utilizarse fácilmente para la detección de colisiones entre cada una de las partes del cuerpo humano.

Además, los modelos de cuerpo humano 3D están incorporados con sombreado. Para una malla dada, los gradientes de deformación de forma se concatenan en un vector de una sola columna. Un modelo de "Blinn-Phong" con componentes difusos y especulares se puede utilizar para aproximar la reflectancia de un cuerpo cuando hay una sola fuente de luz Blinn 1977. Las sombras emitidas desde una fuente de luz puntual proporcionan restricciones adicionales sobre la postura y la forma Balan y col. 2007b. Después de estimar los parámetros de postura y forma, se determina la posición de la luz desde las sombras, y también se reestima la posición y la forma de las regiones de primer plano y las regiones de sombra.

Modelos que son lo suficientemente expresivos como para representar una amplia gama de cuerpos humanos y posturas con bajas dimensiones también se exploran Freifeld y col. 2010. Los autores de Anguelov y col. 2005 se basan en el modelo SCAPE y desarrollan una representación factorizada.

2.1.3.4 Conocimientos previos sobre la postura humana

La postura del cuerpo humano está limitada por varios factores, como la cinemática, los límites operacionales de las articulaciones y los patrones de comportamiento del movimiento en actividades específicas Cheung y col. 2004; Rius y col. 2009. Las restricciones cinemáticas, junto con un modelo dinámico, proporcionan suficiente información para estimar las posturas humanas Moeslund y Granum 2001a.

La disponibilidad de técnicas de captura de movimiento Wei y Chai 2010; Liu y col. 2011; Wang y Cheng 2010 permite que se aprendan posturas previas a partir de datos. Para aprender las restricciones de la postura del esqueleto del cuerpo humano eficientemente, los autores de Sminchisescu y Triggs 2003 recogen un conjunto de datos de captura de movimiento para explorar las posibilidades de la postura del esqueleto del cuerpo humano. Con los datos recogidos, se puede utilizar etiquetas de datos de las variables de articulación con ejemplos positivos y negativos Demirdjian, Ko y Darrell 2003a. Sin embargo, plantear posturas aprendidas de un movimiento tiene problemas posteriormente para generalizar estas posturas a nuevos movimientos Wang, Fleet y Hertzmann 2007.

Algunos estudios aprenden la postura humana como un modelo de postura dependiente de los límites de las variables de articulación Urtasun y col. 2005, y otros entrenan a los bosques aleatorios (random forests - RFs) y el análisis de la dirección principal para modelar los cuerpos humanos Dinh y col. 2014. En Kostrikov y Gall 2014 se extienden los bosques de regresión para inferir en las oclusiones de las características de una imagen de profundidad y la postura del esqueleto del cuerpo humano en 3D de forma simultanea. Para los modelos basados en la física con dinámica, obras relacionadas incluyen Brubaker, Fleet y Hertzmann 2007; Metaxas y Terzopoulos 1993. Cuando la información temporal está disponible, los modelos previos Jaeggli, Koller-Meier y Van Gool 2007 de movimiento humano se puede aprender para restringir la inferencia de las secuencias de postura en 3D para mejorar el seguimiento monocular de la postura humana.

2.1.4 Metodologías

Hay dos maneras principales de categorizar los algoritmos de estimación de la postura del esqueleto del cuerpo humano. Basándose en si la estimación de la postura del esqueleto del cuerpo humano es modelada como una proyección geométrica o si se trata como un problema específico de procesamiento de imágenes, los trabajos relacionados se pueden clasificar en dos grupos principales: métodos generativos o métodos discriminatorios.

Otra forma de categorización diferencia entre si el problema de la estimación de la postura del esqueleto del cuerpo humano se resuelve comenzando con una abstracción de alto nivel y trabajando hacia abajo o comenzando con evidencia de píxel de bajo nivel y trabajando hacia arriba.

2.1.4.1 Métodos Discriminativos y Métodos Generativos

El modelo generativo se define en términos de una representación gráfica por ordenador de posturas (computer graphics rendering of poses). Normalmente se requiere un modelo de cuerpo humano volumétrico, y el modelo se proyecta al espacio de imagen y se ajusta de tal manera que la proyección y la observación de la imagen son conformes. Mientras que en los métodos de aprendizaje se modelan las correspondencias entre las características de la imagen y las posturas humanas, el problema de la estimación de la postura humana en 3D se trata como un problema de búsqueda o de regresión.

El método de aprendizaje es generalmente más rápido, ya que sólo considera las observaciones de imágenes, mientras que el método generativo modela el proceso intrínseco de este problema. El modelo discriminativo consiste en un conjunto de funciones de mapeo que se construyen automáticamente a partir de un conjunto de entrenamiento etiquetado de posturas de cuerpo y sus características de imagen respectivas.

Una de las diferencias entre los métodos generativos y los métodos discriminativos es que la primera metodología parte de un modelo de cuerpo humano inicializado con una postura y proyecta la postura al plano de la imagen para verificar con evidencia de imagen, mientras que la segunda metodología parte de la evidencia de la imagen y usualmente aprende un mecanismo que modela las relaciones entre la evidencia de la imagen y las posturas humanas sobre la base de datos de entrenamiento. Sus direcciones de trabajo son completamente opuestas.

2.1.4.1.1 Métodos Discriminativos

Los enfoques discriminatorios parten de la evidencia de la imagen, la estimación de la postura se plantea mediante un mecanismo de mapeo o mediante un mecanismo basado en la búsqueda (search-based mechanism). El modelo que describe las relaciones entre la evidencia de la imagen y las posturas humanas podría ser aprendido de una base de datos de entrenamiento Bowden, Mitchell y Sarhadi 2000. Una vez entrenado el modelo, la prueba suele ser más rápida que los métodos generativos, ya que desciende a un cálculo de formulación o un problema

de búsqueda restringido en lugar de optimizar un espacio paramétrico de alta dimensión.

Los enfoques discriminatorios buscan las soluciones óptimas dentro de su alcance Bo y Sminchisescu 2010; Lee y Elgammal 2010; Sminchisescu y col. 2011; Memisevic, Sigal y Fleet 2012; Urtasun y Darrell 2008; Zhao y col. 2008. Ha habido muchos estudios utilizando esta metodología de métodos, y se pueden dividir en dos subcategorías principales: métodos basados en el aprendizaje Elgammal y Lee 2014; Flitti y col. 2010 y métodos basados en ejemplos Mori y Malik 2006; Toyama y Blake 2002. Estas subcategorías se dividen de la siguiente manera:

1. Métodos basados en el aprendizaje

    - Basados en métodos de mapeo (mapping). Un modelo extremadamente popular para el aprendizaje de estos tipos de mapas es Máquinas de vector soporte, "Support Vector Machines" (SVMs). SVMs Ronfard, Schmid y Triggs 2002a; Okada y Soatto 2008; Zhang y col. 2014 son clasificadores discriminantes que entrenan hiperplanos para la discriminación entre clases. Los ejemplos más decisivos en el entrenamiento se eligen como vectores de soporte. Para imágenes de profundidad, en Kim y col. 2015 se utilizan SVM y superpixeles para la estimación en tiempo real de la postura del esqueleto del cuerpo humano. Del mismo modo, en "Relevance Vector Machines" (RVMs), que es un método de kernel bayesiano, los ejemplos de entrenamiento más decisivos se eligen como vectores relevantes Agarwal y Triggs 2006c; "Metric Regression Forests for Human Pose Estimation" 2016; Sedai, Bennamoun y Huynh 2011; Sedai, Bennamoun y Huynh 2009. Los "modelos de mapeo no lineal" (non-linear mapping models) también se utilizan, por ejemplo, los procesos gaussianos Gong, Gonzalez y Roca 2012.

      Los mecanismos de mapeo más complejos pueden modelarse con un "modelo de mezcla de expertos" (Mixture of Experts - MoE), un "modelo Bayesiano de mezclas de expertos" (Bayesian mixtures of experts - BME) y otros modelos. Por ejemplo, los autores de Sigal y Black 2006b explotan un modelo de MoE aprendido que representa los condicionales Agarwal y Triggs 2004c; Agarwal y Triggs 2004b; Sminchisescu y col. 2005 para inferir una distribución de posturas 3D condicionadas a las posturas 2D. BME Jordan y Jacobs 1994; Ning y col. 2008 podría modelar la distribución multi-modelo del espacio de la postura del esqueleto del cuerpo humano 3D condicionado en el espacio de características, ya que la relación imagen-a-postura es poco lineal.

Los métodos basados en el mapeo también pueden categorizarse en métodos de mapeo directo y métodos de impulso de mejora 2D a 3D (2D-to-3D boosting methods). Una clase de enfoques de aprendizaje utiliza el mapeo directo a partir de las características de la imagen Agarwal y Triggs 2006c; Ionescu, Li y Sminchisescu 2011; Ionescu, Bo y Sminchisescu 2009; Bo y Sminchisescu 2009; Rosales y col. 2001; Hara y Chellappa 2013; Roth, Sigal y Black 2004, y otra clase de enfoques mapea las características de la imagen a las partes 2D y luego usa métodos de modelado o aprendizaje para mapear las partes 2D a las posturas 3D Andriluka, Roth y Schiele 2010; Barbulescu y col. 2012; Cour y col. 2009; Simo-Serra y col. 2013; Akhter y Black 2015.

Basándose en si el mapeo es entrenado con datos de la base de verdad (ground truth) o no, el mapeo puede ser tanto supervisado como sin ser supervisado Kanaujia, Sminchisescu y Metaxas 2007; Yang, Saleemi y Shah 2013. Además, se utilizan métodos semi-supervisados en Olshausen y Field 1997; Gong, Xiang y Hongeng 2010; Cour y col. 2008.

- Métodos basados en el aprendizaje espacial. Tanto el espacio como el subespacio de la topología se utilizan para entrenar el mapeo. Por ejemplo, en un método basado en el espacio de topología se podrían aprender deformaciones arbitrarias no rígidas de una superficie de malla 3D como multidimensional Yao, Gall y Van Gool 2012; Elgammal y Lee 2014; Taylor y col. 2012; Baak y col. 2013a; Freifeld y Black 2012; Christoudias y Darrell 2005; Morariu y Camps 2006; Gall, Yao y Van Gool 2010.

  Por otro lado, el subespacio también podría aprenderse para restringir el espacio de la solución. Por ejemplo, una inserción se puede aprender colocando imágenes en posturas similares cerca, evitando la estimación de las posiciones de la articulación del cuerpo Gupta y col. 2008; Mori y col. 2015. Las tecnologías de reducción dimensional también pueden utilizarse para eliminar información redundante Sminchisescu y Jepson 2014. También se pueden realizar algoritmos de codificación lineal (LLC) con limitación de localidad Wang y col. 2010; Sun y col. 2014 para aprender el mapeo no lineal con el fin de reconstruir posturas humanas en 3D.

  Otros métodos, como "Análisis de Componentes Relevantes" (RCA) Kanaujia, Sminchisescu y Metaxas 2007, "Análisis de Correlación Canónica" (CCA) y "Factorización de matriz no negativa" (NMF) Agarwal

y Triggs 2006a también son algoritmos típicos utilizados para extraer datos correlacionados.

- Métodos basados en la "bolsa de palabras" (Bag-of-words methods). La secuencia de procesos de la bolsa de palabras es la solución del algoritmo de visión por computador más popular antes del aprendizaje profundo. La idea principal de la secuencia de procesos de la bolsa de palabras es extraer primero las características más representativas como un vocabulario y luego denotar cada uno de los datos de entrenamiento basados en la evidencia de imagen y el vocabulario de manera estadística: la ocurrencia de cada palabra en la imagen es contada, todas las ocurrencias de palabras en el vocabulario forman un histograma, y este histograma se toma como la representación final de la imagen de entrada. Esta representación de las características alimenta a un clasificador o un modelo de regresión para completar la tarea Ning y col. 2008.

  Seleccionando las características más representativas como el vocabulario, seguido de una representación de histograma basada en el vocabulario, una imagen puede ser representada con un vector de longitud fija igual al tamaño del vocabulario. De esta manera, la imagen se representa con una ocurrencia estadística de las características más salientes y se comprime al tamaño del vocabulario.

- Métodos basados en el aprendizaje profundo. El aprendizaje profundo es un método de aprendizaje de extremo a extremo que aprende automáticamente la información clave en las imágenes. Las "Redes Neuronales Convolucionales" (CNN) Jain y col. 2014a; Toshev y Szegedy 2014; Pinto, Cox y DiCarlo 2008; Jain y col. 2014b son modelos populares de aprendizaje profundo que tienen múltiples capas, con cada capa compuesta de múltiples convoluciones y algunas otras arquitecturas híbridas. En Schwarz, Schulz y Behnke 2015 se utilizan redes neuronales para la estimación de la postura utilizando imágenes monoculares y de profundidad. La estimación de la postura humana basada en el aprendizaje profundo tiene tres categorías principales: (1) la detección de partes combinadas con la localización exacta de partes del cuerpo humano a través de redes de aprendizaje profundo Li, Liu y Chan 2014; Ouyang, Chu y Wang 2014; Gkioxari, Girshick y Malik 2015; (2) características de aprendizaje a través de redes neuronales convolucionales profundas y el aprendizaje de la cinemática del cuerpo humano a través de la modelización gráfica Tompson y col. 2014; Jain y col. 2014a; (3) el aprendizaje de ambas características y localización de las partes del

cuerpo a través de redes de aprendizaje profundo Chen y Yuille 2014; Toshev y Szegedy 2014; Pfister y col. 2014; Fan y col. 2015.

Los métodos de regresión, Hara y Chellappa 2013, basados en el aprendizaje profundo tienen varias extensiones, como una "mezcla de Redes Neuronales" (mixture of Neural Networks - NNs) Flitti y col. 2010 que utiliza una red hacia adelante (feedforward) de dos capas y neuronas de salida lineales como un modelo para la regresión NN local. Los autores de Tompson y col. 2014 también proponen una arquitectura combinada que implica una red convolucional profunda y un modelo de "Markov Random Field" (MRF). Los autores de Gkioxari y col. 2014a presentan una CNN que incluyen aprendizaje y regiones con detectores de características CNN (R-CNN) con menos funciones. Recientemente, los autores de Carreira y col. 2016 adoptan un error iterativo de retroalimentación que cambia una solución inicial mediante la alimentación de las predicciones de error.

2. Métodos basados en ejemplos:

   Los enfoques basados en ejemplos estiman la postura de una imagen de entrada visual desconocida Babagholami Mohamadabadi y col. 2014 basada en un conjunto discreto de posturas específicas con sus correspondientes representaciones Flitti y col. 2010. Árboles aleatorios (Randomized trees) Amit y Geman 1997 y bosques aleatorios (Random forests - RF) Breiman 2001; Chang y Nam 2013 son rápidas y robustas técnicas de clasificación que pueden manejar este tipo de problema Lepetit y Fua 2006. El RF es un clasificador de conjuntos que consiste en varios árboles de decisión aleatorios Taylor y col. 2012; Belagiannis y col. 2014b y tiene un nodo no terminal que contiene una función de decisión para predecir las correspondencias por regresión de imágenes a nodos terminales, como los vértices de malla Shotton y col. 2011. Los RF mejorados fueron utilizados por Dantone y col. 2013, que empleó RF de dos capas como regresores de la conjunción, con la primera capa que actúa como un clasificador de la parte del cuerpo discriminativo y el segundo que predice posiciones de las variables de articulación según los resultados de la primera capa.

   Otro tipo de enfoque se basa en los bosques de "Hough". Los bosques de "Hough" son combinaciones de bosques de decisión, y los nodos de hoja en cada árbol son un nodo de clasificación o un nodo de regresión. El conjunto de nodos hoja puede considerarse como un libro de códigos discriminativo. Los autores Girshick y col. 2011 retrocedieron directamente un desplazamiento a varias ubicaciones de las variables de articulación en cada píxel.

Las versiones mejoradas incluyen un objetivo optimizado, como un objetivo de partes basado en la ganancia discreta de información Shotton y col. 2011, mientras que otras obras informan del problema de generalización del objetivo especificado Buntine y Niblett 1992; Nowozin 2012. Además, se utiliza la representación escasa (SR) para extraer las muestras de entrenamiento más significativas y, posteriormente, todas las estimaciones se realizan sobre la base de estas muestras Chen y col. 2011; Wright y col. 2009; Huang y Yang 2009; Huang y Yang 2010.

2.1.4.1.2 Métodos Generativos

Las predicciones realizadas en el nivel de píxeles producen un conjunto de señales independientes de posición local que es poco probable que respeten restricciones cinemáticas. Al ajustar un modelo generativo a estas señales, en Taylor y col. 2012; Baak y col. 2013b; Ganapathi y col. 2010 se resuelve este problema.

Los enfoques generativos Lepetit y Fua 2005; Parameswaran y Chellappa 2004; Pons-Moll y Rosenhahn 2011; Güdükbay, Demir y Dedeoglu 2013; Zhang, Shang y Chan 2014; Babagholami Mohamadabadi y col. 2014; Zhu, Dariush y Fujimura 2008 modelan la probabilidad de las observaciones dada una estimación de pose. La inferencia implica una búsqueda compleja sobre el espacio de estado para localizar los picos de la probabilidad Sminchisescu y col. 2005. Los métodos generativos son susceptibles a mínimos locales, y por lo tanto requieren buenas estimaciones de la postura del esqueleto del cuerpo humano inicial, independientemente del esquema de optimización utilizado. La postura suele inferirse utilizando la optimización local Bregler, Malik y Pullen 2004; Brubaker, Fleet y Hertzmann 2010; Ganapathi, Plagemann y Koller 2012; Pons-Moll y col. 2011a; Stoll y col. 2011 o la búsqueda estocástica Deutscher y Reid 2005; Gall y col. 2010; Pons-Moll y col. 2011b. En Probst, Fossati y Gool 2016 proponen utilizar métodos generativos para estimar la postura y la apariencia del esqueleto del cuerpo humano utilizando imágenes de profundidad.

2.1.4.1.3 Métodos combinados de métodos discriminatorios y generativos

Los métodos generativos proyectan el modelo humano en el espacio de imagen 2D y miden una distancia entre ellos Flitti y col. 2010, mientras que los métodos discriminativos detectan las partes del cuerpo humano para reconstruir la postura del esqueleto del cuerpo humano. Los métodos generativos sufren de baja eficiencia, mientras que los métodos discriminativos luchan por generalizar las posturas no presentes en los datos de entrenamiento Ning y col. 2008.

Para sacar provecho de ambas metodologías y evitar sus defectos, se han realizado investigaciones explorando la combinación de estos dos tipos de métodos juntos. La combinación se implementa generalmente mediante la inicialización de la postura con la estimación de los métodos discriminatorios Kanaujia 2014 y la optimización de la postura del esqueleto del cuerpo humano en un área local a través de métodos generativos Orrite-Urunuela, Herrero-Jaraba y Rogez 2004; Sigal, Balan y Black 2007; Agarwal y Triggs 2005. A través de la optimización iterativa en el proceso generativo, las posturas del modelo humano 3D se ajustan comparando con la evidencia de la imagen en el proceso discriminativo.

En los métodos generativos, el espacio de las siluetas puede proyectarse desde las posturas humanas 3D. Una postura genera varias siluetas diferentes bajo varios puntos de vista Sidenbladh, Black y Fleet 2010. Los parámetros estructurales del modelo 3D volumétrico articulado contribuyen a la proyección del modelo geométrico del cuerpo humano 3D Sminchisescu y Telea 2002; Wei y Chai 2009, y la regla de Bayes podría ser utilizada para estimar los parámetros del modelo y lograr una interpretación probabilística. Una postura estimada con el método discriminativo podría utilizarse como inicialización, y el espacio de la silueta (manifold of silhouette space) podría ser utilizado para optimizar la solución Gall, Yao y Van Gool 2010; Lee y Elgammal 2007.

Otros métodos combinados incluyen el modelado probabilístico de Gauss y otros "Integrating Bottom-up / Top-down for Object Recognition by Data Driven Markov Chain Monte Carlo" 2016; Kuo, Makris y Nebel 2011; Kanaujia 2014. Estos dos modelos también podrían combinarse para inferir la postura humana articulada derivando en una formulación combinada Rosales y Sclaroff 2006.

2.1.4.2 Métodos ascendentes y métodos descendentes

Consideramos una segunda manera de categorizar, basada en la dirección que los algoritmos de la estimación de la postura del esqueleto del cuerpo humano trabajan semánticamente. Es decir, el método funciona desde la abstracción semántica de nivel superior hasta el nivel bajo, o funciona al revés. Las imágenes se consideran como el nivel más bajo en la jerarquía semántica, la configuración de la postura del esqueleto del cuerpo humano se considera como en el nivel más alto, y también los tipos de la acción humana a los cuales pertenecen las posturas del esqueleto del cuerpo humano.

2.1.4.2.1 Métodos ascendentes

En los métodos ascendentes se recogen y describen fragmentos de evidencia de imagen para formar rasgos descriptivos. Estas características a veces se utilizan directamente para predecir las posturas humanas, y algunas veces se utilizan para localizar partes del cuerpo cuyas ocurrencias en las imágenes se ensamblan para formar un evento humano. En la Sección 2.1.4.1, se discuten los mecanismos que modelan las representaciones de las imágenes y las correspondencias de las posturas del esqueleto del cuerpo humano. En esta sección, se recopilan y se comparan métodos que fusionan pruebas de imágenes de bajo nivel para formar una semántica de alto nivel. Basándose en el tamaño de la unidad, los métodos ascendentes se pueden dividir de la siguiente manera:

1. Métodos basados en píxeles o superpíxeles

   La información de píxeles también se puede utilizar para aumentar la precisión de estimación de la postura del esqueleto del cuerpo humano Wang, Wang y Yuille 2013. Por ejemplo, la información de píxeles se utiliza como entrada a un proceso de análisis iterativo, que aprende mejores características ajustadas a una imagen en particular Ramanan 2007.

   Los píxeles o superpíxeles de una imagen también se pueden utilizar para formular una función de segmentación y ser integrados en la estimación de la posición. Por ejemplo, pueden utilizarse para formular la función energética de los algoritmos de segmentación e integrar la segmentación de objetos con una optimización conjunta Eichner, Ferrari y Zurich 2009; Wang y Koller 2011; Lu, Shao y Xiao 2013.

   Los métodos basados en píxeles también se pueden combinar con otros métodos. Por ejemplo, los autores de Hernández-Vela y col. 2012 amplían el

método de clasificación por píxel con la poda de grafos (graph-cut) de optimización, que es un marco de minimización de energía. Además, los resultados de la segmentación se pueden utilizar para mejorar la estimación a nivel de píxeles. Los autores de Ferrari, Marin-Jimenez y Zisserman 2008 proponen un enfoque que reduce progresivamente el espacio de búsqueda de partes del cuerpo mediante el empleo de "grabcut" inicializado en las regiones detectadas para reducir aún más el espacio de búsqueda Ferrari, Marin-Jimenez y Zisserman 2009; Ferrari, Marín-Jiménez y Zisserman 2009. Los enfoques basados en partes y basados en píxeles también se pueden combinar en un solo marco de optimización Ladicky, Torr y Zisserman 2013.

Los superpixels también son útiles para restringir las posiciones de las articulaciones en el modelo del esqueleto del cuerpo humano Mori 2005. En los métodos basados en superpixeles, la optimización de la correlación de partes del cuerpo y del primer plano obtenida mediante el etiquetado del superpíxel podría optimizarse, por ejemplo, con un algoritmo de ramificación y poda (branch and bound - BB) Jiang 2010; Tian y Sclaroff 2010; Sun y col. 2012; Nakariyakul 2014. Además, los autores de Hao, Fanhui y Baofu 2014 comparan la calidad de la segmentación derivada de los modelos de apariencia generados por varios enfoques.

2. Métodos basados en partes

   Los métodos basados en partes resuelven problemas de la estimación de la actitud a través del aprendizaje de la apariencia de la parte del cuerpo y los modelos de posición. En métodos basados en partes, los candidatos de la parte del cuerpo son detectados por primera vez a partir de la evidencia de la imagen, y luego las partes del cuerpo detectadas se montan para ajustarse a las observaciones de la imagen y un plan del cuerpo Rogez y col. 2008. Como un trabajo icónico, se introdujo un modelo de mezcla flexible de partes en Yang y Ramanan 2011, que extiende el "Modelo de Partes Deformables" (DPM) Felzenszwalb y col. 2010 para la estimación de la posición del cuerpo 2D articulado. Se mejoró aún más utilizando un modelo de composición gráfico Rothrock, Park y Zhu 2013.

   Una cuestión clave en los métodos basados en partes es decidir cómo fusionar las respuestas de cada parte del cuerpo en un todo, y esto está relacionado con cómo el cuerpo humano es modelado. Organizamos lo siguiente basándonos en las características de los modelos de cuerpo humano, y dividimos aún más los métodos basados en partes.

- Estructuras Pictóricas. Las estructuras pictóricas Sapp, Toshev y Taskar 2010; Andriluka, Roth y Schiele 2009; Sapp, Jordan y Taskar 2010; Ferrari, Marin-Jimenez y Zisserman 2009; Andriluka, Roth y Schiele 2008; Belagiannis y col. 2014a; Penmetsa y col. 2014; Eichner y Ferrari 2012 son un tipo de modelo cinemático gráfico sobre los métodos de detección, con los nodos de la gráfica que representan las partes del objeto, y las ariustas entre las partes que codifican las relaciones geométricas.

  Se han desarrollado diferentes deformaciones de los modelos clásicos de Estructuras Pictóricas, como las "Estructuras Pictóricas Adaptativas" (APS) Sapp, Jordan y Taskar 2010, las "Estructuras Pictóricas Múltiples" (MPS) Eichner y Ferrari 2010, las "Estructuras Pictóricas Condicionadas de Poselet" Pishchulin y col. 2013, los "Campos de Partes" (FOP) Kiefel y Gehler 2014, y otros.

  La estructura en árboles es una de las estructuras pictóricas más aplicadas con éxito. El modelo descompone una estructura de árbol en términos de apariencia unaria y pares potenciales entre pares de partes físicamente conectadas. Con los métodos de ventanas corredizas, las plantillas de partes de cuerpo entrenadas se comparan con las características de la imagen. Las respuestas de todas las partes del cuerpo se pasan a través de la estructura del árbol, y una puntuación final se calcula en la raíz del árbol.

- Modelos cinemáticos mejorados. Los modelos cinemáticos mejorados suelen tener un mejor aspecto y son más expresivos al describir las limitaciones de la postura. Por ejemplo, se incluyen una variedad de modos para mejorar las capacidades de representación del modelo cinemático, como el modelo "Multimodal decomposable Model" (MODEC), que tiene un modo izquierdo y derecho y modos de cuerpo completo y medio cuerpo.

  También se han realizado muchos estudios sobre la mejora de los modelos cinemáticos con estructuras en cascada. Por ejemplo, los autores de Sapp, Toshev y Taskar 2010 proponen una cascada gruesa a fina de modelos de estructura pictórica. Los estados del marco en cascada podían ser podados y calculados Bo y Jiang 2013. Recurriendo a múltiples árboles, la estructura estima los parámetros para todos los modelos, requiriendo sólo un aumento lineal en el cálculo sobre el aprendizaje o infiriendo en un solo sub-modelo manejable Weiss, Sapp y Taskar 2010. Los autores de Tian, Zitnick y Narasimhan 2012 proponen un nuevo modelo espacial jerárquico que puede capturar un número exponen-

cial de posturas con una representación de mezclas compactas en cada parte. Usando nodos latentes, representa una relación espacial de alto orden entre las partes con inferencia exacta.

Además, en lugar de predefinir un modelo cinemático, un modelo de árbol latente Wang y Li 2013 puede recuperar un modelo gráfico estructurado en árbol que se aproxima mejor a las distribuciones de un conjunto de observaciones. Además, al modificar los métodos de regresión, se puede mejorar la precisión de la estimación de la posición. Por ejemplo, los autores de Eichner, Ferrari y Zurich 2009 introducen regresores articulares del cuerpo dependientes de las partes para clasificar las partes del cuerpo y predecir las ubicaciones de las articulaciones.

Las funciones de coste locales de los hijos en los modelos estructurados en árbol podrían ser correctamente trasladados a sus padres, mientras que en el caso de oclusión, la función de coste puede pasar al padre equivocado, resultando en partes perdidas y detección inexacta, convirtiendo la estructura del árbol en un grafo Radwan, Dhall y Goecke 2013. También se proponen modelos mejorados de estructura de árbol para tratar este problema. El método de corrección de la oclusión basado en la regresión podría detectar la oclusión mediante la codificación de las configuraciones cinemáticas en un árbol. Como las partes no adyacentes son independientes, podrían estimarse las partes ocluidas Sigal y Black 2006a. Los problemas de escorzo (foreshortening) y la variación de parte de la escala se pueden abordar mediante la definición de una parte del cuerpo con las articulaciones del cuerpo en lugar de miembros del cuerpo, Rogez y col. 2008; Urtasun y Darrell 2008; Lee y Nevatia 2006.

Recientemente se han propuesto métodos no basados en árboles para facilitar restricciones estructurales más fuertes, y pueden ser optimizados usando programación convexa o propagación de creencias Ning y col. 2008. Se cree que los modelos gráficos "loopy" son necesarios cuando las partes combinadas se utilizan para manejar gran variación en el aspecto Wang, Tran y Liao 2011. Los "Modelos Gráficos Loopy", Sapp, Weiss y Taskar 2011; Cherian y col. 2014, comienzan enviando mensajes desde los nodos hoja a la raíz, y luego desde el nodo raíz hasta el resto. Los modelos de gramática articulada son otro ejemplo de modelos no-árbol. Los autores de Rothrock, Park y Zhu 2013 presentan un marco utilizando el modelo de gramática articulada para integrar un modelo de fondo en la gramática para mejorar el rendimiento de la localización.

2.1.4.2.2 Métodos descendentes

El método descendente se utiliza para referirse a los métodos generativos en Torres y Kropatsch 2013; Brox, Rosenhahn y Weickert 2005, pero en este estudio usamos este término para denotar el proceso de resolución de problemas de trabajo de semántica de alto nivel a evidencia de imagen de nivel inferior Torres y Kropatsch 2013, donde la semántica de alto nivel se utiliza para guiar el reconocimiento de bajo nivel. Con esta noción, los métodos descendentes se utilizan mas en combinación con los métodos ascendentes que utilizándolos solo como métodos separados, ya que generalmente lo que queremos lograr es la semántica de nivel más alto.

2.1.4.2.3 Métodos ascendentes y descendentes combinados

La forma en que se combinan los métodos ascendentes y los métodos descendentes es más flexible que la forma en que se combinan los métodos discriminatorio y generativo:

1. Combinación entre métodos de detección y de reconocimiento.

   Motivado por la literatura extensa tanto sobre la detección Gavrila 2007; Viola, Jones y Snow 2005; Zhu y col. 2006; Wu, Lin y Huang 2005; Sabzmeydani y Mori 2007; Weiss, Sapp y Taskar 2010 y el reconocimiento Agarwal y Triggs 2006c; Shakhnarovich, Viola y Darrell 2003; Yang y Bissacco 2010; Mori y Malik 2006; Rogez, Orrite-Urunuela y Rincón 2008; Hernández y col. 2008; Yacoob y Black 1998, muchos trabajos exploran la posibilidad de combinar estos dos tipos de métodos juntos para mejorar la precisión de la estimación Dimitrijevic, Lepetit y Fua 2006; Bissacco, Yang y Soatto 2006. Por ejemplo, al combinar los modelos cinemáticos gráficos con los métodos de detección, la detección y las posturas 3D podrían obtenerse simultáneamente Ionescu, Li y Sminchisescu 2011; Bray, Kohli y Torr 2006; Rogez y col. 2008; Kohli, Rihan y Bray 2008. Por otra parte, los autores de Yu, Kim y Cipolla 2013 introducen un método de estimación monocular de la postura 3D a partir de un video usando la detección de acciones sobre partes deformables en 2D.

2. Combinación entre métodos basados en píxeles y basados en partes.

   La correspondencia y segmentación simultánea de objetos de optimización permite resultados más sólidos, ya que los dos métodos estrechamente relacionados basados en píxeles y en partes se apoyan entre sí, Jiang 2011; Lu,

Shao y Xiao 2013; Ladicky, Torr y Zisserman 2013. Por ejemplo, se pueden obtener etiquetas de partes del cuerpo en píxeles combinando enfoques basados en partes y en píxeles en un solo marco de optimización, Ladicky, Torr y Zisserman 2013.

Los autores de Bray, Kohli y Torr 2006 utilizan poda de grafos (graph-cut) para optimizar los parámetros de la postura del esqueleto del cuerpo humano para realizar la segmentación integrada y la estimación de la postura 3D del esqueleto del cuerpo humano. Los mínimos globales de las energías se pueden encontrar por la poda de grafos (graph-cut), Kolmogorov y Zabin 2004, y el cálculo de la poda de grafo se hace perceptiblemente más rápido usando el algoritmo del la poda de grafo dinámico (graph-cut algorithm), Kohli y Torr 2005.

2.1.4.3 Métodos basados en el movimiento

Con la información temporal, la estimación de la postura del esqueleto del cuerpo humano podría potenciarse con coherencia temporal y espacial, y la estimación de la postura del esqueleto del cuerpo humano también podría considerarse como seguimiento de la postura humana. En este caso, no sólo se aprenden la forma y la apariencia de la parte del cuerpo, sino que también se debe extraer el movimiento de la parte del cuerpo.

Con las señales de movimiento, los puntos de articulación del esqueleto del cuerpo humano se pueden estimar por el movimiento de las partes rígidas, y las restricciones entre las partes adyacentes en modelos basados en partes se modelan principalmente como modelos gráficos Felzenszwalb y col. 2010; Ferrari, Marin-Jimenez y Zisserman 2008; Cheung, Baker y Kanade 2003; Wang y col. 2012. Los autores de Ju, Black y Yacoob 1996 modelan el cuerpo humano como una colección de parches planares sometidos a un movimiento afín y las restricciones blandas penalizan la distancia entre los puntos de articulación previstos por los modelos afines adyacentes. En un enfoque similar, los autores Datta, Sheikh y Kanade 2008 obligan a los desplazamientos de las articulaciones del cuerpo a ser iguales en los modelos afines de las partes adyacentes, lo que resulta en una simple optimización de mínimos cuadrados lineales limitados para el seguimiento de partes restringidas cinemáticamente.

Los parámetros del modelo de movimiento también se pueden optimizar directamente. Por ejemplo, el Algoritmo de Densidad de Curva de Contratante (CCD) Hahn y col. 2007 refina un conjunto de parámetros inicial para ajustar un modelo de curva paramétrica a una imagen. Adicionalmente, el modelo “Wandering-

Stable-Lost" (WSL) Balan y Black 2006 se desarrolló en el contexto de la estimación de movimiento paramétrico. La información de movimiento también se puede extraer como campos de flujo. Por ejemplo, los campos de flujo articulado se infieren mediante el uso de la segmentación del etiquetado de la postura Fragkiadaki, Hu y Shi 2013. También se proponen métodos de estimación de movimiento de partes Bregler y Malik 1998; Rehg y Kanade 1995; Ghosh y col. 2015.

El muestreo es otra manera de resolver los modelos de movimiento. La técnica de la cadena de "Markov Monte Carlo" (MCMC) se utiliza con frecuencia en el movimiento basado en la estimación de la postura humana como un método de muestreo. El conjunto de muestras de soluciones generadas por la cadena de "Markov" converge débilmente en una distribución estacionaria equivalente a la distribución posterior. El marco de MCMC basado en datos "Integrating Bottom-up / Top-down for Object Recognition by Data Driven Markov Chain Monte Carlo" 2016; Lee y Cohen 2004a permite diseñar buenas funciones de propuesta derivadas de observaciones de imagen como el contorno de la cara, el hombro y la piel y las manchas de color de la piel. El paso de mensajes de partículas (PAMPAS) también se puede utilizar para resolver problemas basados en el movimiento en forma de propagación no paramétrica de creencias "Attractive People: Assembling Loose-Limbed Models Using Non-Parametric Belief Propagation" 2016; Sigal y col. 2012.

Adicionalmente, se propone un algoritmo de verificación y ajuste de escala para ajustar automáticamente las escalas de perspectiva durante el proceso de seguimiento para abordar el problema de las escalas de perspectiva múltiple Tian, Li y Liu 2014.

Los "Procesos Gaussianos" (GP), que pueden usarse para especificar la distribución sobre la función, son generalizaciones de distribuciones gaussianas definidas sobre conjuntos de índices infinitos Zhao y col. 2008; Guo y Qian 2008; Rasmussen 2006.

Después de incorporar la información temporal, se propone el "Modelo Gaussiano de Variable Latente del Proceso" (GPLVM) Ek, Torr y Lawrence 2007; Urtasun, Fleet y Lawrence 2007; Hou y col. 2007; Tian, Li y Sclaroff 2005; Tian y col. 2010 para conocer las distribuciones de estilos de movimiento humano con correspondencia multifactorial con las variables latentes. Además, se ha propuesto el uso de "Modelos Dinámicos de Proceso Gaussiano" (GPDM) Urtasun, Fleet y Fua 2006 para el aprendizaje de la postura y el movimiento de objetos humanos para el seguimiento de personas en 3D Urtasun, Fleet y Fua 2005. Además, basándose en modelos dinámicos de aprendizaje, los procesos auto regresivos gaussianos pueden aprenderse dividiendo automáticamente el espacio de parámetros en regiones con características dinámicas similares Agarwal y Triggs 2004d. Para una secuencia

de movimiento particular, se utiliza un modelo de dinámica de círculo (CDM) cuando el estilo se supone constante en el tiempo para restringir el contenido de diferentes estilos para que se encuentren en la misma trayectoria Wang, Fleet y Hertzmann 2007.

El algoritmo “locality-constrained linear coding” (LLC) Sun y col. 2014 es otra manera de codificar los atributos de movimiento en dimensiones reducidas. LLC se realiza para aprender el mapeo no lineal con el fin de reconstruir una postura 3D humana. Una nueva codificación para LLC se propone en un marco discriminativo utilizando “motionlets” como libros de códigos en Wang y col. 2010.

Capítulo 3

Modelo 4D - DPM

En este capítulo se habla sobre el modelo propuesto para la detección de cada uno de los puntos de interés dentro de la imagen. Estos puntos de interés coinciden con las variables de articulación del esqueleto del cuerpo humano.

3.1 Introducción

En el anexo B se presenta el modelo original utilizado,Felzenszwalb, McAllester y Ramanan 2008, donde se utiliza un modelo de partes deformables (DPM) utilizando 6 partes para la detección de objetos dentro de una imagen fija. Posteriormente, en Yang y Ramanan 2013 utiliza una nueva representación de modelos de partes deformables basada en en el estado anterior para la detección de la postura del esqueleto del cuerpo humano, donde se utilizan 14 partes en el modelo DPM para la detección de la postura.

Hasta ahora, el método de Yang y Ramanan 2013 ha sido el estado del arte para la estimación de la postura del esqueleto del cuerpo humano en imágenes monoculares. Sin embargo, como podemos ver en las imágenes del capítulo 6, el método de Yang y Ramanan 2013 obtiene resultados erróneos en las imágenes que difieren bastante de las imágenes con las que se ha entrenado el modelo, e incluso después de re-entrenar el modelo con imágenes similares, el método solo mejora en un margen muy pequeño.

Durante el estudio del estado del arte hemos visto otros algoritmos que han mejorado el método de Yang y Ramanan 2013, como es el caso de Wang y Li 2013; Pishchulin y col. 2013; Ramakrishna y col. 2014, todos estos métodos, incluyendo Yang y Ramanan 2013, utilizan el método de mezcla de partes (mixture of parts) solo en imágenes RGB, 3 canales. En cambio, el método propuesto utiliza una mezcla multi-canal de partes (multi-channel mixture of parts) que nos ayuda a extender el número de mezclas de partes a la dimensión de profundidad en imágenes RGBD. En Shotton y col. 2013c se utilizan solo imágenes de profundidad, en comparación de nuestro modelo que utiliza imágenes de profundidad y RGB. La utilización de ambas imágenes, RGB y de profundidad, radica en que ambas pueden ser complementarias en aquellos casos en donde una imagen tiene más probabilidad de aportar soluciones erróneas. Por ejemplo en las imágenes de profundidad donde la persona puede tener los brazos delante del cuerpo y la resolución de esta no permita diferenciarlas los brazos del cuerpo, o por otro lado donde una parte del cuerpo es confundido con el mismo color con otra parte de la imagen. Es por ello por lo que se ha decidido utilizar imágenes RGB y de profundidad, para que las desventajas de la utilización de una de ellas, sean las ventajas de la utilización de la otra.

En el presente capítulo se presenta en la sección 3.2 el primer paso a realizar, que consiste en la calibración de la cámara, es dicir, calibrar los sensores RGB y de profundidad. En la sección 3.3 se describe el procesamiento de la imagen para la sustracción del fondo utilizando el método MSER. En la sección 3.4 se presenta el método DPM desarrollado en Yang y Ramanan 2013, utilizando solo imágenes monoculares y 14 partes, el cual ha sido modificado para obtener el método 4D-DPM propuesto en la sección 3.5, el cual utiliza imágenes monoculares y de profundidad y se han reducido el número de partes a 10. Como se a mencionado anteriormente, el anexo B describe el modelo DPM original

3.2 Calibración de la cámara

Primero es necesario calibrar los parámetros intrínsecos e extrínsecos de la cámara a utilizar corrigiendo las distorsiones, Ricolfe y Sanchez 2011; Ricolfe, Sanchez y Valera 2013 son ejemplos de calibración de los parámetros de la cámara en cámaras de gran angular, para poder relacionar los píxeles de la imagen de profundidad con los píxeles de la imagen RGB.

Con la ayuda de la calibración de la cámara se obtienen los parámetros necesarios para pasar un píxel de la imagen RGB a la imagen de profundidad y viceversa

sa, del mismo modo se puede obtener un píxel de la imagen de profundidad y representarlo en coordenadas 3D respecto de la cámara.

La figura 3.1 muestra en qué consiste la calibración de la imagen RGB y de profundidad.

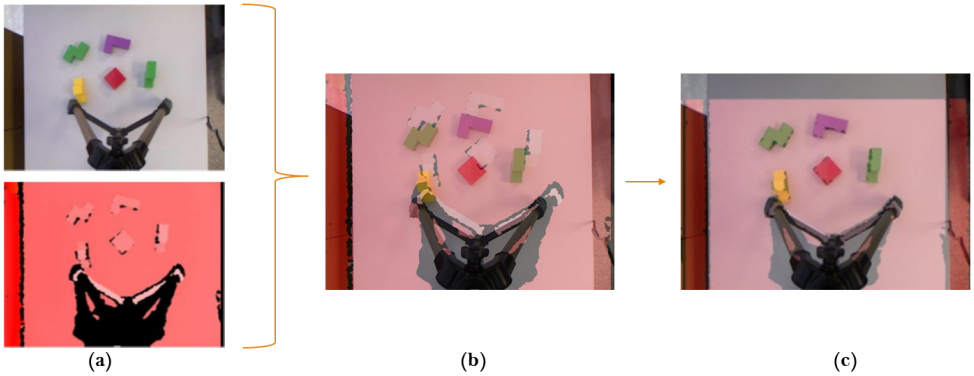


Figura 3.1: Calibración de los sensores RGB y de profundidad. (a) Imágenes de entrada. (b) Imágenes solapadas, los píxeles no se corresponden. (c) Imágenes solapadas tras realizar la calibración de ambos sensores, los píxeles se corresponden entre ambas imágenes.

Observando la figura 3.1 se tiene como imágenes de entrada la imagen RGB y la de profundidad (a). Las dos imágenes son solapadas en (b) y se observa que ambas imágenes no se corresponden, las piezas de colores están desplazadas entre ambas imágenes. La imagen (c) muestra la superposición de ambas imágenes, RGB y profundidad, tras realizar la calibración de ambos sensores, se aprecia que las piezas en ambas imágenes están situadas en los mismos píxeles.

3.3 MSER

Una vez realizada la calibración de la cámara, en nuestro caso la Kinect, se realiza la sustracción del fondo en las imágenes de entrada al modelo 4D-DPM, para eliminar el ruido de la imagen. La eliminación del fondo de la imagen se realiza mediante la eliminación de aquellas regiones en la imagen de profundidad que son más inestables a diferentes umbrales que pertenecen al fondo. Esta plantilla, a partir de la imagen de profundidad, es transferida a la imagen RGB, eliminando así el ruido que confundiría a las características del modelo 4D-DPM y dificultaría la detección de los puntos de interés. La figura 3.3 muestra de forma gráfica el proceso en cuestión.

En la figura 3.2 se observa en qué etapa dentro del esquema se encuentra el procesamiento de imágenes utilizando MSER.

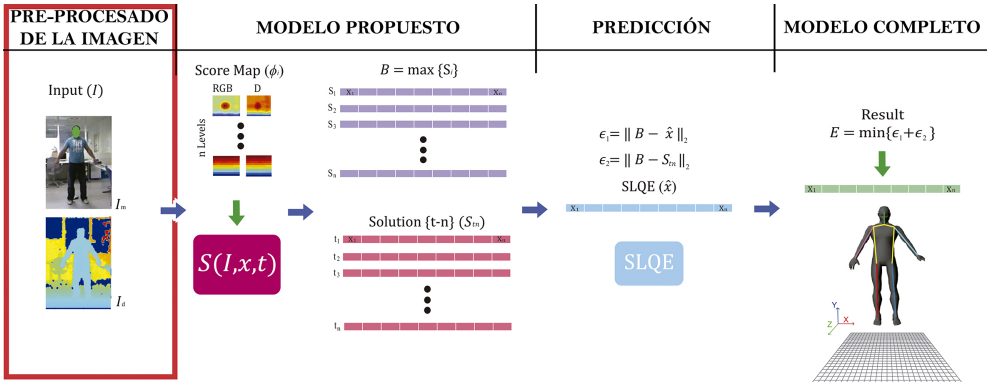
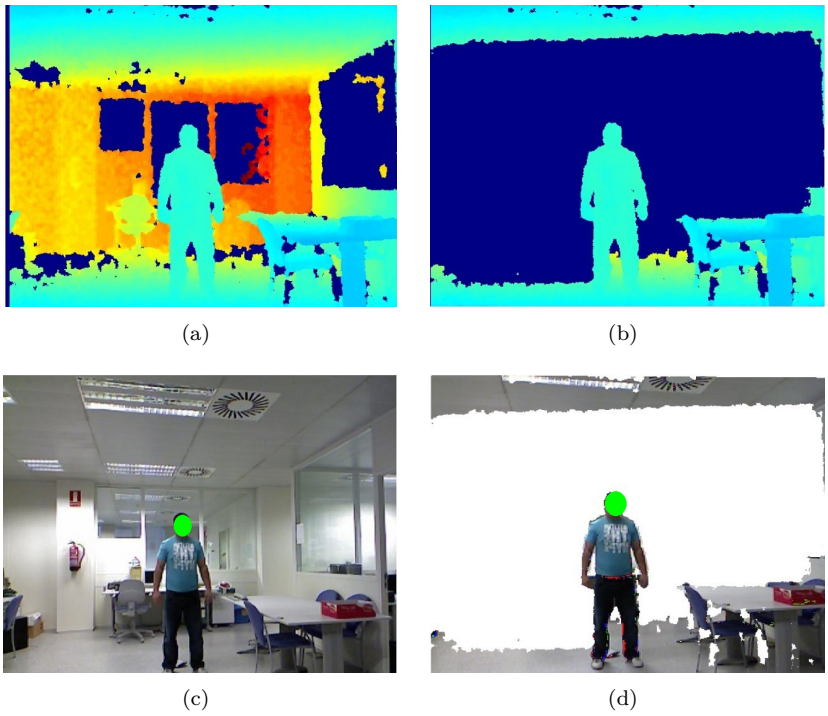


Figura 3.2: Introducción del procesamiento de imágenes.

La intuición detrás de este enfoque es que los objetos o personas en primer plano visto a través del sensor de profundidad comparten áreas con intensidades de píxeles similares. La razón de esto es que los rayos infrarrojos (IR) que se reflejan en los objetos en primer plano se reflejan más o menos al mismo tiempo y con la misma intensidad. Otros objetos o áreas mucho más alejadas de la cámara de IR reflejan de manera desigual y estas áreas son más ruidosas y con intensidades variables. La figura 3.3 muestra las diferentes intensidades reflejadas por el sensor IR que representa las coordenadas de profundidad de los objetos.



(a) (b)

(c) (d)

Figura 3.3: (a) Imagen de profundidad original; (b) Profundidad tras aplicar MSER; (c) Imagen original de RGB; (d) Combinamos las imágenes (b) y (c).

Debido a esta propiedad de las intensidades de píxeles en las imágenes de profundidad, el método de sustracción de fondo que se utiliza para imágenes de profundidad y que se aplica posteriormente a imágenes RGB, utiliza el método "maximally stable extremal regions" (MSER) basado en Matas y col. 2004. Estas regiones son las más estables a través de un rango de todos los posibles umbrales que se les aplican. La variable de estabilidad δ de cada una de las regiones en los canales de profundidad es calculada como $\delta = \frac{|\Delta R - R|}{|R|}$, donde $|R|$ representa el área de la región en cuestión y Δ representa la variación de intensidad para los diferentes umbrales. Por lo tanto, se eliminan las regiones MSER cuyas áreas están por encima de un umbral T. Los parámetros que afectan al método MSER son entrenados utilizando un subconjunto de imágenes que se utilizarán posteriormente para entrenar el método propuesto. Se puede observar en la figura 3.3 los resultados del método de sustracción de fondo y se aprecia que los píxeles que forman parte del fondo de la imagen son removidos.

3.4 Modelo original DPM

En la figura 3.4 se observa en qué etapa dentro del esquema utilizado se encuentra la utilización del modelo DPM.

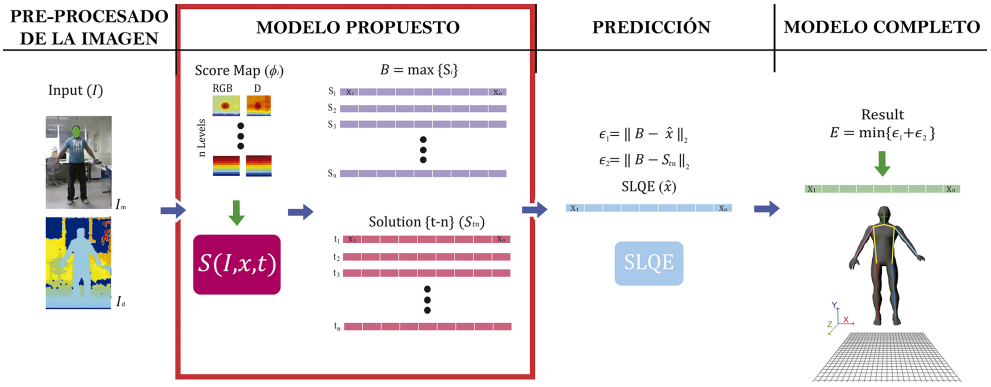


Figura 3.4: Introducción del método utilizado para el seguimiento de los puntos de interés.

3.4.1 Motivación

El modelo descrito es una aproximación para capturar una familia continua de deformaciones. El enfoque clásico de usar un conjunto finito de plantillas es también una aproximación. En esta sección, se presenta un análisis teórico directo de ambos. Para simplificar, se restringe a las deformaciones afines, aunque una derivación similar se aplica a cualquier función de deformación débil, incluyendo las deformaciones de perspectiva.

Se escribe x para la posición de los píxeles en $2D$ dentro de la plantilla, y $\omega(x) = (I + \Delta A)x + b$ para su nueva posición bajo una pequeña deformación afín $A = I + \Delta A$ y una translación b. Se utiliza ΔA para parametrizar la desviación de la deformación de una deformación de identidad. Se define $s(x) = \omega(x) - x$ para ser el cambio de una posición x. El cambio de una posición cercana $x + \Delta x$ puede ser escrito como:

$$s(x+\Delta x) = \omega(x+\Delta x) - (x+\Delta x) = (I+\Delta A)(x+\Delta x) + b - x - \Delta x = s(x) + \Delta A \Delta x \tag{3.1}$$

Los dos píxeles x y $x + \Delta x$ cambian en la misma medida, y pueden ser modelados como una parte simple, si el producto $\Delta A \Delta x$ es pequeño, esto es cierto si el de-

terminante de ΔA es pequeño o la norma de Δx es pequeña. Los modelos clásicos articulados utilizan una gran familia de articulaciones discretizadas, en las que cada plantilla discreta sólo necesita explicar una pequeña gama de rotaciones (por ejemplo, un valor de ΔA poco determinante). Aquí se toma el enfoque opuesto, haciendo Δx pequeño utilizando partes pequeñas. Ya que es preferible una norma pequeña para Δx, esto sugiere que las partes circulares funcionarían mejor, pero se usan partes cuadradas como una aproximación discreta. En el caso extremo, se podría definir un conjunto de partes de un solo píxel. Tal representación es de hecho la más flexible, pero resulta difícil de entrenar dada la formulación de aprendizaje descrita a continuación.

3.4.2 Modelo

Se escribe I para una imagen, $l_i = (x, y)$ para la localización de un píxel de la parte i y t_i para la componente mezcla (mixture) de la parte i. Se escribe $i \in \{1, ..., K\}$, $l_i \in \{1, ..., L\}$ y $t_i \in \{1, ..., T\}$. Se llama t_i al "tipo" de la parte i. Por conveniencia notacional, se define la falta de subíndice para indicar un conjunto abarcado por ese subíndice (ejemplo, $t = \{t_1, ..., t_k\}$). Para simplificar, se define el modelo para una escala fija. En el tiempo de testeo se detectan personas de diferentes tamaños buscando sobre una pirámide de imágenes con múltiples escalas,ver figura B.3.

3.4.2.1 Modelo de co-ocurrencia

Para anotar una configuración de partes, primero se define una función de compatibilidad para tipos de partes:

$$S(t) = \sum_{i \in V} b_i^{t_i} + \sum_{ij \in E} b_{ij}^{t_i, t_j} \tag{3.2}$$

El parámetro $b_i^{t_i}$ favorece asignaciones de tipos particulares para la parte i, mientras el parámetro por parejas $b_{ij}^{t_i, t_j}$ favorece asignaciones de co-ocurrencia de la parte tipos. Por ejemplo, si la parte tipo corresponde a orientaciones y la parte i y j están en la misma extremidad rígida, entonces $b_{ij}^{t_i, t_j}$ favorecería a asignaciones consistentes de orientación. Específicamente, $b_{ij}^{t_i, t_j}$ debería ser un número positivo grande para orientaciones consistentes t_i y t_j, y un número negativo grande para orientaciones inconsistentes t_i y t_j.

3.4.2.2 Rigidez

Se escribe $G = (V, E)$ para (estructura de árbol) un grafo de relación K-nodo, cuyas aristas especifican qué pares de partes están limitadas para tener relaciones consistentes. Este grafo todavía puede codificar relaciones entre partes distantes a través de la transitividad. Por ejemplo, el modelo puede obligar a una colección de partes a compartir la misma orientación, siempre y cuando las partes formen una conexión *subtree* de $G = (V, E)$. Esta propiedad se utiliza para modelar múltiples partes en el torso. Dado que los parámetros de co-ocurrencia son aprendidos, el modelo aprende qué subconjuntos de partes deben ser rígidas.

Ahora se puede escribir la notación completa asociada a la configuración del tipo de partes y posiciones:

$$S(I, l, t) = S(t) + \sum_{i \in V} \omega_i^{t_i} \cdot \phi(I, l_i) + \sum_{ij \in E} \omega_{ij}^{t_i, t_j} \cdot \psi(l_i - l_j) \tag{3.3}$$

donde $\phi(I, l_i)$ es la función de apariencia representada por HOG, Dalal y Triggs 2005b, que extrae las características de una imagen monocular (I) en la localización del píxel l_i. Se escribe $\psi(l_i - l_j) = [dx, dx^2, dy, dy^2]^T$, donde $dx = x_i - x_j$ y $dy = y_i - y_j$. La localización relativa es definida con respecto a la rejilla de píxeles y no a la orientación de la parte i (como en las estructuras pictóricas articuladas).

3.4.2.3 Modelo de apariencia

La primera suma en la ecuación 3.3 es el modelo de apariencia que computa la asignación local de colocar una plantilla $w_i^{t_i}$ para la parte i, sintonizado para el tipo t_i, en la localización l_i.

3.4.2.4 Modelo de deformación

El segundo término de la ecuación 3.3 puede ser interpretado como un modelo de muelle (spring) de conmutación que controla la colocación relativa de la parte i y j conmutando entre una colección de muelles. Cada muelle está adaptado para un par particular de tipos (t_i, t_j) y es parametrizado por su posición de reposo y rigidez, que están codificados por $\omega_{ij}^{t_i, t_j}$. El modelo de muelle de conmutación codifica la dependencia de la apariencia local en geometría, desde que diferentes pares de mezclas locales (local mixtures) están restringidos a la utilización de diferentes muelles. Junto con el término de co-ocurrencia, especifica una imagen independiente sobre las ubicaciones de las partes y tipos.

3.4.3 Simplificaciones del modelo

Ahora se describe varios casos especiales del modelo.

3.4.3.1 Modelos humanos estirables

Sapp, Weiss y Taskar 2011 describe un modelo de partes del ser humano que consiste en una parte simple para cada variable de articulación. Esto es equivalente al modelo original con $K = 14$ partes, cada parte con una sola mezcla $T = 1$. Similar al modelo original, Sapp, Weiss y Taskar 2011, argumentan que una representación céntrica conjunta capta eficientemente el efecto de articulación. Sin embargo, los modelos locales de mezcla propuestos (para $T > 1$) también capturan la dependencia de la geometría global sobre la apariencia local.

3.4.3.2 Modelos de partes semánticas

Epshtein y Ullman 2007a argumenta que las apariencias parciales deben captar las clases semánticas y no las clases visuales. Esto se puede hacer con un modelo tipo. Considere un modelo facial con partes de los ojos y la boca. Uno puede querer modelar diferentes tipos de ojos (abiertos y cerrados) y bocas (sonriendo y frunciendo el ceño). La relación espacial entre los dos probablemente no depende de su tipo, pero los ojos abiertos tienden a co-ocurrir con las bocas sonrientes. Esto se puede obtener como un caso especial del modelo usando un único resorte para todos los tipos de un par particular de partes:

$$\omega_{ij}^{t_i,t_j} = \omega_{ij} \tag{3.4}$$

3.4.3.3 Modelo de partes deformables (DPM)

Felzenszwalb y col. 2010 define mezclas de modelos, donde cada modelo es una estructura pictórica basada en estrellas. Esto puede lograrse restringiendo el modelo de co-ocurrencia para permitir sólo tipos consistentes a nivel global:

$$b_{ij}^{t_i,t_j} = \begin{cases} 0 & if \quad t_i = t_j \\ -\text{ínf} & \text{otherwise} \end{cases} \tag{3.5}$$

3.4.3.4 Articulación

En el modelo original de DPM, se explora una versión simplificada de la ecuación 3.3 con un conjunto reducido de resortes:

$$\omega_{ij}^{t_i,t_j} = \omega_{ij}^{t_i} \tag{3.6}$$

La simplificación anterior establece que la ubicación relativa de la parte con respecto a su matriz depende del tipo de parte, pero no del tipo padre. Por ejemplo, sea i una parte de la mano, j su parte del codo padre, y asuma que los tipos de parte capturan la orientación. El modelo relacional anterior indica que una mano orientada hacia los lados debería tender al lado del codo, mientras que una mano orientada hacia abajo debe estar debajo del codo, independientemente de la orientación del brazo.

3.4.4 Inferencia

La inferencia corresponde en maximizar $S(I, l, t)$ en la ecuación 3.3 sobre l y t. Cuando el grafo relacional $G = (V, E)$ es un árbol, esto se puede hacer eficientemente con la programación dinámica. Para ilustrar la inferencia, se vuelve a escribir la ecuación 3.3 definiendo $z_i = (l_i, t_i)$ para denotar tanto la ubicación de los píxeles discretos como el tipo de mezcla discreta de la parte i:

$$S(I, z) = \sum_{i \in V} \phi(I, z_i) + \sum_{ij \in E} \psi(z_i, z_j) \tag{3.7}$$

donde $\phi(I, z_i) = \omega_i^{t_i} \cdot \phi(I, l_i) + b_i^{t_i}$, $\psi(z_i, z_j) = \omega_{ij}^{t_i,t_j} \cdot \psi(l_i, l_j) + b_{ij}^{t_i,t_j}$.

Con esta perspectiva, el modelo final original DPM es un modelo "pairwise Markov Random Field" (MRF). Cuando $G = (V, E)$ es un árbol estructural, uno puede computar $max_z S(I, z)$ con programación dinámica.

Para ser precisos, se itera sobre todas las partes partiendo de las hojas y moviéndonos "aguas arriba" hacia la parte de la raíz. Se define $kids(i)$ como el conjunto de hijos de la parte i, que es el conjunto vacío para las partes de la hoja. Se calcula la parte de mensaje i que pasa a su padre j por la siguiente ecuación.

$$score_i(z_i) = \phi_i(I, z_i) + \sum_{k \in kids(i)} m_k(z_i) \tag{3.8}$$

$$m_i(z_j) = \max_{z_i} \left[score_i(z_i) + \psi_{ij}(z_i, z_j) \right] \tag{3.9}$$

La ecuación 3.8 calcula la puntuación local de la parte i, para todas las localizaciones de los píxeles l_i y para todos los posibles tipos t_i, recogiendo mensajes de los hijos de i. La ecuación 3.9 calcula para cada localización y posible tipo de la parte j, la mejor ubicación y el tipo de las partes de sus padres i. Una vez que los mensajes se pasan a la parte raíz ($i = 1$), $score_1(z_1)$ representa la mejor configuración para cada raíz posición y tipo. Se puede usar estas puntuaciones de raíz para generar múltiples detecciones en la imagen I mediante el umbral de ellas y la aplicación de supresión no máxima (NMS). Al hacer un seguimiento de los índices $argmax$, se puede retroceder para encontrar la ubicación y el tipo de cada parte en cada configuración máxima. Para encontrar múltiples detecciones ancladas en la misma raíz, se pueden utilizar las extensiones $Nbest$ de programación dinámica.

3.4.4.1 Cálculo

La porción computacionalmente tributable de la programación dinámica es la ecuación 3.9. Se reescribe este paso en detalle:

$$m_i(t_j, l_j) = \max_{t_i}[b_{ij}^{t_i,t_j} + \max_{l_i} score_i(t_i, l_i) + \omega_{ij}^{t_i,t_j} \cdot \psi(l_i - l_j)] \tag{3.10}$$

Se tiene que hacer un bucle sobre LxT posibles ubicaciones y tipos de los padres, y calcular un máximo sobre LxT posibles ubicaciones y tipos de hijos, haciendo el cálculo $O(L^2T^2)$ para cada parte. Cuando $\psi(l_i - l_j)$ es una función cuadrática (como es el caso), la maximización interna en la ecuación 3.10 puede ser calculada eficientemente para cada combinación de t_i y t_j en $O(L)$ con una transformación de máxima convolución o distancia, Felzenszwalb y Huttenlocher 2005a. Puesto que uno tiene que realizar T^2 transformaciones de distancia, el paso de mensajes se reduce a $O(LT^2)$ por parte.

3.4.4.2 Casos especiales

El modelo de la ecuación 3.4 mantiene un solo resorte por pieza, por lo que el paso del mensaje se reduce a $O(L)$.

Los modelos de las ecuaciones 3.5 y 3.6 mantienen solo T resortes por parte, reduciendo el paso de mensajes a $O(LT)$. Vale la pena señalar que el modelo articulado no es más complejo computacionalmente que las mezclas de partes deformables en Felzenszwalb y col. 2010. Pero es considerablemente más flexible porque busca sobre un número exponencial (T^K) de mezclas globales. En la práctica, el tiempo de cálculo está dominado por el cálculo de las puntuaciones locales de cada

modelo de apariencia específica $\omega_i^{t_i} \cdot \phi(I, l_i)$. Dado que esta puntuación es lineal, puede calcularse eficientemente para todas las posiciones l_i mediante rutinas de convolución optimizadas.

3.4.5 Entrenamiento

Se asume un paradigma de aprendizaje supervisado. Dado ejemplos positivos etiquetados $\{I_n, l_n, t_n\}$ y ejemplos negativos $\{I_n\}$, se definirá una función objetivo de predicción estructurada similar a la propuesta en Felzenszwalb y col. 2010 y Kumar, Zisserman y Torr 2009. Para ello, se escribe $z_n = (l_n, t_n)$ y tenga en cuenta que la función de puntuación de la ecuación 3.7 es lineal en los parámetros del modelo $\beta = (\omega, b)$, y puede ser escrita como $S(I, z) = \beta \cdot \Phi(I, z)$. Se aprendería un modelo de la forma:

$$\begin{aligned} &\arg \min_{\omega, \xi_n \geq 0} \frac{1}{2}\beta \cdot \beta + C \sum_n \xi_n \\ &\text{s.t. } \forall n \in \text{pos } \beta \cdot \Phi(I_n, z_n) \geq 1 - \xi_n \\ &\forall n \in \text{neg}, \forall z \ \beta \cdot \Phi(I_n, z) \leq -1 + \xi_n \end{aligned} \tag{3.11}$$

La restricción anterior establece que los ejemplos positivos deben obtener mejores resultados que 1 (el margen), mientras que los ejemplos negativos, para todas las configuraciones de posiciones y tipos de partes, deben obtener menos de -1. La función objetivo penaliza las infracciones de estas restricciones usando variables flojas ξ_n.

3.4.5.1 Detección vs estimación de la postura

Las tareas de predicción estructurada tradicional no requieren un conjunto de entrenamiento negativo explícito y, en su lugar, generan restricciones negativas a partir de ejemplos positivos con etiquetas erróneamente estimadas z. Esto corresponde a la formación de un modelo que tiende a marcar una postura de la base de verdad (ground-truth) correctamente y alternar posturas erróneamente. Mientras que esto se traduce directamente a una tarea de estimación de la postura, la formulación anterior también incluye un componente de "detección": entrena a un modelo que obtiene puntuaciones altas en posturas de la base de verdad (ground-truth), pero genera puntuaciones bajas en imágenes sin personas. Se encuentra que lo anterior funciona bien tanto para la estimación de la postura como para la detección de personas.

3.4.5.2 Optimización

La optimización anterior es un programa cuadrático (QP) con un número exponencial de restricciones, ya que el espacio de z es $(LT)^K$. Afortunadamente, sólo una pequeña minoría de las restricciones será activa en problemas típicos (por ejemplo, los vectores de soporte), haciéndolos solubles en la práctica. Esta forma de problema de aprendizaje se conoce como SVM estructural, y existen muchos solucionadores bien ajustados, como el "SVMStruct" Tsochantaridis y col. 2004 y el "SGD" en Felzenszwalb y col. 2010. Para permitir una mayor flexibilidad en la programación de las actualizaciones de los modelos y la poda de conjuntos activos, se implementa un solucionador propio de coordenadas descendentes, descrito brevemente a continuación.

3.4.5.3 Descenso de coordenadas duales

El solucionador actualmente más rápido para SVM lineales parece ser "liblinear", Fan y col. 2008, que es un método de descenso de coordenadas duales. Una implementación ingenua de un solucionador SVM dual requeriría mantener una matriz MxM, donde M es el número total de restricciones activas (vectores de soporte).

La innovación de "liblinear" es la realización de que uno puede representar implícitamente la matriz del núcleo para SVMs lineales manteniendo el vector de peso primal β, que es típicamente mucho más pequeño. En la práctica, los métodos de descenso de coordenadas duales son lo suficientemente eficientes para alcanzar soluciones casi óptimas en un solo paso a través de grandes conjuntos de datos Bordes y col. 2007. Algorítmicamente, este paso no requiere más cálculo que SGD, pero se garantiza que el objetivo siempre aumentará el doble, mientras que los métodos estocásticos pueden tomar pasos erróneos en el camino. Se ha derivado una extensión de esta visión para SVMs estructurales, descrito en Ramanan 2012. En pocas palabras, la principal modificación requerida es la capacidad de las restricciones lineales para compartir la misma variable. Específicamente, los ejemplos negativos de la ecuación 3.11 que corresponden a una sola ventana I_n con diferentes variables latentes z comparten la misma holgura ξ_n. Esto complica un poco el paso de coordenadas duales, pero se aplica el mismo principio. Se soluciona el problema dual en coordenadas, una variable a la vez, representando implícitamente la matriz del núcleo con β. También se encuentra que se alcanzan soluciones óptimas en un solo paso a través de nuestro conjunto de entrenamiento.

3.4.5.4 Entrenamiento en practica

La mayoría de los conjuntos de datos de posturas humanas incluyen imágenes con las posiciones de las variables de articulación marcadas Bourdev y Malik 2009; Ferrari, Marin-Jimenez y Zisserman 2008; Ramanan 2007. Se definen las partes que deben situarse en las variables de articulación, esto proporciona las etiquetas de las posiciones de las partes l, pero no etiquetas de tipo de parte t. Ahora se describe un procedimiento para generar etiquetas de tipo para el modelo articulado, ecuación 3.6.

Primero se define manualmente la estructura de las aristas E conectando las posiciones de las variables de articulación en función de la proximidad media. Debido a que se desea modelar la articulación, se puede asumir que los tipos de parte deben corresponder a diferentes localizaciones relativas de una parte con respecto a su padre en E. Por ejemplo, las manos orientadas hacia los lados ocurren al lado de los codos, mientras que las manos hacia abajo ocurren por debajo de los codos . Esto significa que se puede usar la ubicación relativa como una señal de supervisión para ayudar a derivar las etiquetas de tipo que capturan la orientación.

3.4.5.5 Derivación de la parte tipo desde la posición

Supongamos que la n^{th} imagen de entrenamiento I_n ha etiquetado las posiciones de las variables de articulación l_n. Sea l_i^n la posición relativa de la parte i con respecto a su padre en la imagen I_n. Por cada parte i, agrupamos su posición relativa sobre el conjunto de entrenamiento $\{l_i^n : \forall n\}$ para obtener los T clusteres. Utilizamos $K - means$ con $K = T$. Cada cluster corresponde a una colección de instancias de parte con ubicaciones relativas consistentes, y por lo tanto, orientaciones consistentes por los argumentos anteriores. Se definen las etiquetas de tipo para las partes t_i^n basadas en la pertenencia al cluster.

3.4.5.6 Supervisión parcial

Debido a que el tipo de parte se deriva heurísticamente arriba, se podría tratar t_i^n como una variable latente que también se optimiza durante el aprendizaje. Este problema SVM latente puede ser resuelto por “descenso de coordenadas” Felzenszwalb y col. 2010 o el algoritmo “CCP” Yuille y Rangarajan 2003.

3.4.5.7 Problema de tamaño

En los conjuntos de datos de formación, el número de ejemplos positivos varía de $200 - 1000$ y el número de imágenes negativas es aproximadamente 1000. Tratamos cada posible ubicación de la raíz en una imagen negativa como un ejemplo negativo único x_n, lo que significa que tenemos millones de restricciones negativas. Además, se consideran modelos con cientos de miles de parámetros. Entonces se tiene que para manejar el aprendizaje de esta escala, se necesita un seleccionador cuidadoso optimizado.

3.4.6 Representación de las partes

Tras realizar el entrenamiento del modelo original DPM, se obtienen las siguientes representaciones mostradas en la figura 3.5, que representan el modelo entrenado.

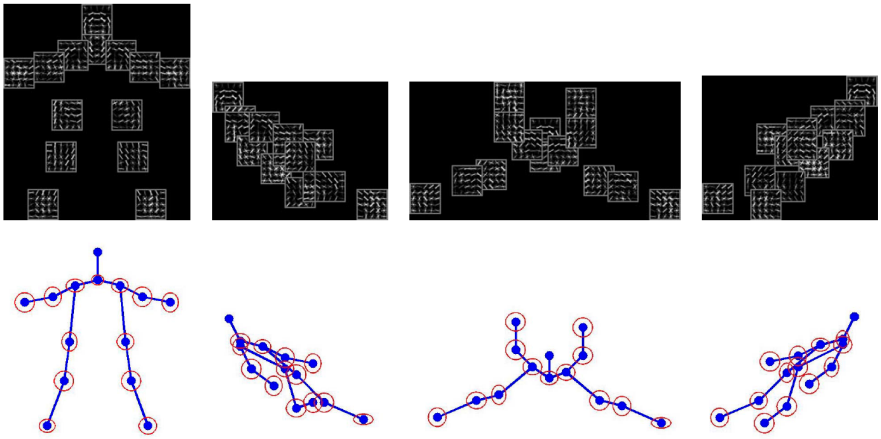


Figura 3.5: Visualización del modelo utilizando 14 partes y 4 mezclas locales. En la fila superior se muestran las plantillas locales, y en la fila inferior se muestran las estructuras de árbol utilizadas. Fuente: Yang y Ramanan 2013.

Se establecen todas las partes para que sean regiones HOG de tamaño $5x5$. Para visualizar el modelo, se muestran 4 árboles generados seleccionando uno de los cuatro tipos de cada parte, y colocándolo en su posición de clasificación máxima. Recuerde que cada tipo de parte tiene su propia plantilla de apariencia y el resorte que codifica su posición relativa con respecto a su matriz.

3.5 Modelo 4D-DPM

3.5.1 Introducción

A continuación se observan los puntos que se ven afectados sobre el método original DPM tras añadir el canal de profundidad.

Uno de los aspectos que distingue el método original DPM del propuesto, radica en que se utiliza una dimensión más, la imagen de profundidad, con lo cual tenemos 4 canales de información para nuestro método. Al aumentar la dimensión del método estamos aumentando también el tiempo de computo del moldeo. Para dar solución a este inconveniente, y gracias a la utilización de la cinemática utilizando cuaterniones duales, capítulo 5, podemos inferir sobre el número de puntos de interés a encontrar dentro de la imagen, que a diferencia del método original que utiliza 14 partes para la representación completa de la postura del esqueleto del cuerpo humano, se propone utilizar solo 10 partes reduciendo considerablemente el tiempo de computo contrarrestando el tiempo añadido tras incluir una dimensión más. Posteriormente, utilizando la cinemática podemos encontrar los partes que faltan para representar completamente la postura del esqueleto del cuerpo humano.

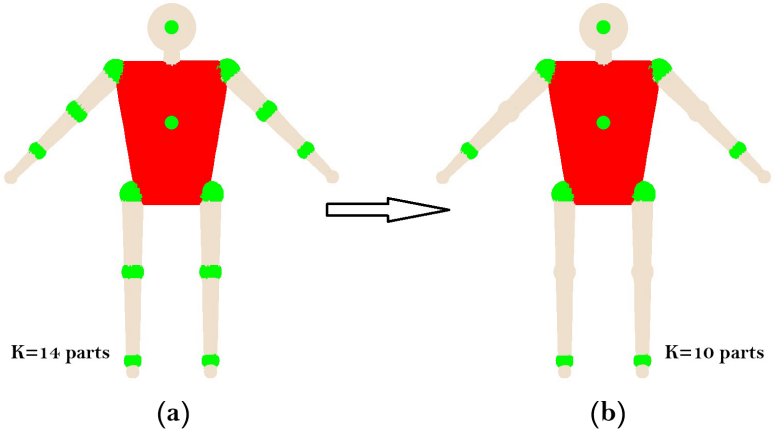


Figura 3.6: (a) modelo DPM en el que nos hemos basado utilizando 14 partes. (b) modelo DPM propuesto reducido utilizado 10 partes.

La figura 3.6 muestra en la imagen (a) las 14 partes del modelo DPM del cual hemos partido para llegar a las 10 partes, imagen (b), que se utiliza en el modelo propuesto.

3.5.2 Formulación introducida

En el método propuesto, se formula la función de coste S para cada una de las partes o variables de articulación que pertenecen a la postura y las funciones de apariencia y deformación como:

$$S(I, x, t) = \sum_{i \in V} \phi_i(I, x_i, t_i) + \sum_{ij \in E} \psi_{i,j}(I, x_i, t_i, x_i^{'}) \tag{3.12}$$

donde I corresponde con la imagen RGBD, x es la localización de la variable de articulación i que se corresponde con el tipo de variable de articulación detectado, j es la variable de articulación potencialmente conectada con i, y $t = 1, ..., T$ es la componente mezcla de la variable de articulación i y que se expande a partes que han experimentado diferentes transformaciones tales como rotación, traslación, orientación y otras y donde $x_i^{'} = (x_j, t_j)$. Los términos ϕ y ψ en la ecuación 3.12 corresponden con los modelos de apariencia y deformación respectivamente. El modelo de apariencia calcula el coste para las características de asignación de tipo t_i mientras que el modelo de deformación proporciona un coste para la distancia de deformación de asignación de tipo t_i y t_j. Estos modelos son restringidos con el árbol de estructura representado por $G(V, E)$, donde un vértice $i \in V$ representa una parte y la arista $(i, j) \in E$ representa la co-ocurrencia de la parte i y j para el propósito de optimización ya que el cálculo de todas las asignaciones posibles es exponencial.

Para obtener las características y deformaciones en todos los canales RGBD, se formula ϕ y ψ como mezcla multi-canal de partes (multi-channel mixture of parts) de la siguiente manera:

$$\begin{aligned} \phi_i\left(I, x_i, t_i\right) &= \begin{bmatrix} \omega_{i\ m}^{t_i} \cdot \phi\left(I_m, x_i\right) + b_{i\ m}^{t_i} \\ \omega_{i\ d}^{t_i} \cdot \phi\left(I_d, x_i\right) + b_{i\ d}^{t_i} \end{bmatrix} \\ \psi_{ij}\left(I, x_i, t_i, x_j, t_j\right) &= \begin{bmatrix} \omega_{ij}^{t_i, t_j} \cdot \psi(x_i - x_j)_m + b_{ij\ m}^{t_i t_j} \\ \omega_{ij}^{t_i, t_j} \cdot \psi(x_i - x_j)_d + b_{ij\ d}^{t_i t_j} \end{bmatrix} \end{aligned} \tag{3.13}$$

donde $\phi_i\left(I, x_i, t_i\right)$ es la función de apariencia representada por HOG, Dalal y Triggs 2005b, que extrae las características de una imagen monocular (I_m) o de profundidad (I_d) en la localización del píxel x_i. m representa la parte monocular

y d representa la parte de profundidad. ω son los filtros previamente entrenados. $b_i^{t_i}$ es el parámetro que se corresponde con el asignación de la parte i en el canal, $b_{ij}^{t_i,t_j}$ es otro parámetro que describe la asignación de co-ocurrencia de la parte i y j. En contraste con Yang y Ramanan 2013, el número de mezclas de partes en la ecuación 3.13 es el doble que el método original a causa de añadir el canal de profundidad. Este número extra de componentes de mezclas complementan a los componentes de mezclas de las componentes RGB y ayudan a mejorar los pesos de detección para todos los canales RGBD. Esta propiedad puede verse también en la figura 3.7 que muestra los diferentes costes obtenidos de diferentes imágenes.

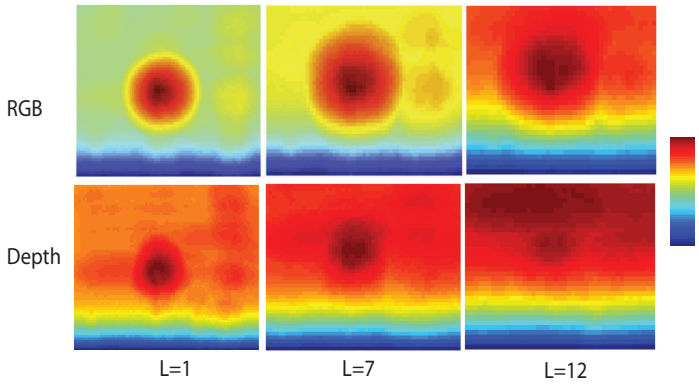


Figura 3.7: Mapa de pesos de los componentes a diferentes niveles. La figura muestra que la mezcla de partes en RGBD es complementaria.

Un ejemplo más concreto para ver que la mezcla de partes en RGBD es complementaria se muestra en la figura 3.8, donde en (a) se tiene la solución dada para la imagen RGB y en (b) la solución para la imagen de profundidad. Se observa en la imagen (a) una zona delimitada en donde se observan los valores máximos (colores mas oscuros) centrados en la parte superior, mientras que en la imagen (b) los valores máximos están dispersos. Esto puede ser debido a una mala información suministrada por el sensor de profundidad, por ejemplo la parte representada esta al mismo nivel de distancia que otros objetos a su alrededor. Con lo cual la imagen RGB facilita la tarea para obtener una mejor solución.

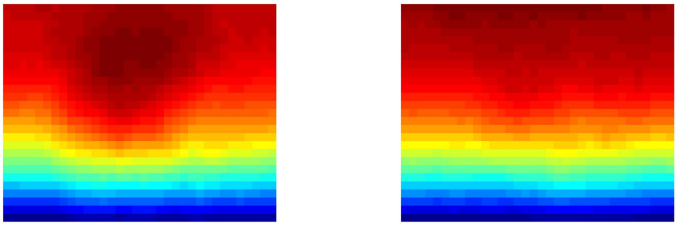


Figura 3.8: Mapa de pesos de los componentes a diferentes niveles. La figura muestra que la mezcla de partes en RGBD es complementaria.

La función de deformación es dada por $\psi(x_i - x_j) = [dx, dx^2, dy, dy^2]$, donde $dx = x_i - x_j$ y $dy = y_i - y_j$, corresponde a la localicación de la parte i respecto a j en la imagen I_c para los respectivos tipos de la imagen c.

Debido a que la estructura $G(V, E)$ es un árbol, se utiliza la programación dinámica para calcular S para cada nodo en el árbol con un segundo termino extra comparado con Yang y Ramanan 2013 para calcular los pesos y el paso de mensajes para los canales de profundidad. Entonces $kids(i)$ es el conjunto de hijos de la parte i en G. Computamos el mensaje de la parte i que pasa a sus parientes j de la siguiente forma:

$$score_i\left(t_i, x_i\right) = b_i^{t_i} + \begin{bmatrix} \omega_{t_i m}^{i} \cdot \phi\left(I_m, p_i\right) \\ \omega_{t_i d}^{i} \cdot \phi\left(I_d, p_i\right) \end{bmatrix} + \sum_{k \in kids(i)} m_k\left(t_i, x_i\right) \tag{3.14}$$

$$m_i\left(t_j, x_j\right) = \max_{t_i} \left[b_{ij}^{t_i, t_j} + \max_{x_i} score\left(t_i, x_i\right) + \begin{bmatrix} w_{ij}^{t_i, t_j}{}_{m} \cdot \psi(x_i - x_j)_m \\ w_{ij}^{t_i, t_j}{}_{d} \cdot \psi(x_i - x_j)_d \end{bmatrix} \right] \tag{3.15}$$

La ecuación 3.14 calcula los pesos locales de la parte i, para todos los píxeles p_i y para todos los posibles tipos t_i, recolectando los mensajes de los hijos de i. La ecuación 3.15 calcula cada posición y tipo de la parte i de sus hijos. Cuando los mensajes son pasados al origen $i = 1$, $score_1(c_1, x_1)$ representa el mejor peso de configuración para cada uno de los tipos de origen y posición.

En contraste con Yang y Ramanan 2013, parametrizamos la ecuación 3.12 como $S(I, x, t) = \alpha \cdot \Phi(I, x, t)$ y $\alpha = (\omega, b)$ para solucionar la siguiente estructura de "support vector machine" (SVM) con las siguientes condiciones para el procesamiento de muestras positivas y negativas que ayudan a resolver las restricciones mas violadas como pasos independientes i, mejorando el tiempo comparado con Yang y Ramanan 2013.

$$\arg \min_{w,\xi \geq 0} \frac{1}{2}\alpha \cdot \alpha + C \sum_{n} \xi_n \tag{3.16}$$
$$\text{s.t. } \forall n \in \text{pos } \alpha \cdot \Phi(I_{ni}, x_{ni}, t_{ni}) \geq 1 - \xi_{ni}$$
$$\forall n \in \text{neg}, \forall x_n, t_n \ \alpha \cdot \Phi(I_n, x_n, t_n) \leq -1 + \xi_n$$

3.5.3 Representación de las partes

Tras realizar el entrenamiento del modelo 4D-DPM, se obtienen las siguientes representaciones mostradas en la figura 3.9, que representan el modelo entrenado.

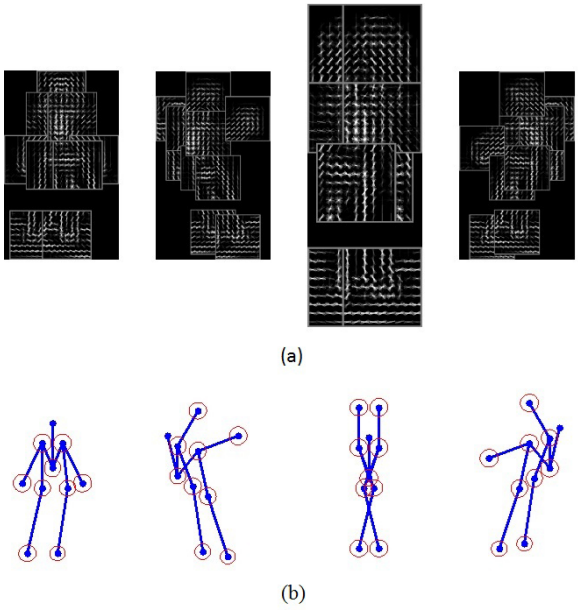


Figura 3.9: Visualización del modelo utilizando 10 partes y 4 mezclas locales. En la fila superior (a) se muestran las plantillas locales, y en la fila inferior (b) se muestran las estructuras de árbol utilizadas.

En comparación con la figura 3.5, donde se utilizan 14 partes, se observa que los modelos obtenidos son diferentes.

Capítulo 4

Restricciones en el filtro de partículas

En este capítulo se introducen las restricciones introducidas en el filtro de partículas para aumentar la precisión del filtro. Las restricciones introducidas son: límites en las variables de articulación, detección de colisiones y la proyección del modelo 3D sobre el plano 2D para observar la correspondencia de la proyección 2D con un plano de la imagen RGBD.

4.1 Introducción

Una de las características vistas sobre el modelo DPM es que es un modelo discriminativo. Tras incluir el filtro de partículas al modelo, el modelo resultante además de discriminativo es un modelo generativo.

El filtro de partículas utilizado se basa en el método explicado en el anexo C. A este filtro de partículas se le han añadido restricciones para obtener mayor precisión en los resultados obtenidos. Eso es debido a que gracias a estas restricciones introducidas, las partículas generadas por el filtro, según la restricción adecuada, son o bien generadas todas ellas dentro de unos límites posibles o bien se eliminan aquellas partículas como solución posible debido al incumplimiento de alguna de las restricciones introducidas.

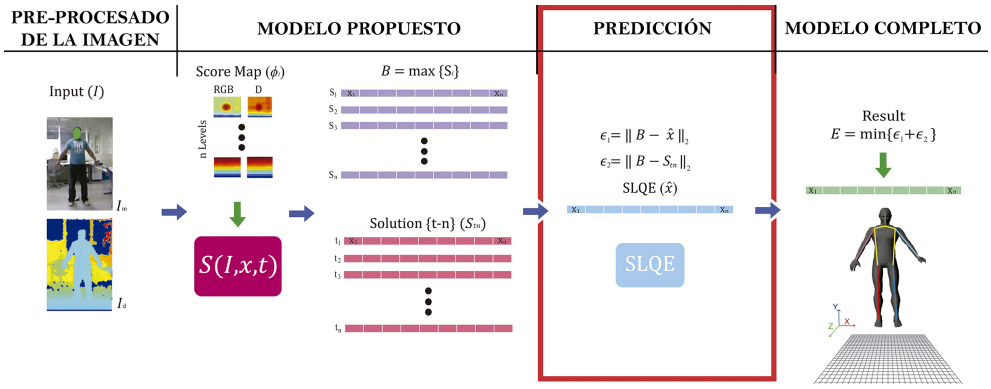


Figura 4.1: Introducción del filtro de predicción utilizado.

La figura 4.1 muestra la etapa en la que se emplea el filtro de partículas dentro de nuestro esquema.

El siguiente capítulo se divide entonces en las 3 restricciones introducidas: Restricción 1, límites de las variables de articulación, sección 4.2. Restricción 2, Detección de colisiones, sección 4.3. Restricción 3, Proyección 2D de las poli-esferas, sección 4.4, el cual en el anexo D se detallan las ecuaciones previas. Y por último, en la última sección, sección 4.5, se detalla el algoritmo del filtro de partículas utilizado.

4.2 Restricción 1: límites de las variables de articulación

El esqueleto del cuerpo humano se ha dividido en 5 partes, tal y como se observa en el capítulo 5, estas partes son: brazo derecho, brazo izquierdo, pierna derecha, pierna izquierda y el tronco.

Como pasa en todas las cadenas cinemáticas, hay lugares que una variable de articulación no puede alcanzar, es decir, las variables de articulación están limitadas a un rango específico. Esta restricción en el movimiento del esqueleto del cuerpo humano nos ayuda a introducir la siguiente restricción, limitar las variables de articulación para cada una de las partes en que se ha dividido el cuerpo humano. Estos límites son observados en la siguiente tabla, Tabla 4.1:

Tabla 4.1: Límites de las variables.

Parte	Límite	q_1	q_2	q_3	q_4	q_5	q_6
Brazo derecho	Superior	180	0	135	180	65	180
	Inferior	-45	-180	0	-180	-65	-180
Brazo izquierdo	Superior	180	180	0	180	65	180
	Inferior	-45	0	-135	-180	-65	-180
Pierna derecha	Superior	200	0	0	45	45	180
	Inferior	-20	-90	180	-45	-45	-180
Pierna izquierda	Superior	200	90	0	45	45	180
	Inferior	-20	0	-180	-45	-45	-180

Cada columna de la tabla 4.1 representa una variable de articulación para cada una de las partes descritas. Donde para cada parte se representa el límite superior (fila superior) y el límite inferior (fila inferior) de cada variable de articulación.

Esto permitirá comprobar si las variables de articulación calculadas en el filtro de partículas son las correctas o no y por lo tanto dar por buenas las partículas generadas para obtener la solución dada.

Su funcionalidad consiste en que las partículas que genera aleatoriamente el filtro de partículas tengan un rango de valores correctos, es decir, que ninguna partícula contenga valores en los que no se puede actuar, es decir, si se sabe que una de las variables a controlar tiene un rango de 0 a 100, el filtro de partículas genera todas las partículas y estas son modificadas para que no excedan del rango deseado. De esta forma todas las partículas generadas son correctas y dentro de los valores deseados.

4.3 Restricción 2: Detección de colisiones

Como se observa en el siguiente capítulo, capítulo 5, la representación del cuerpo humano se ha reducido a la utilización de poli-esferas que alberguen en su interior todo el cuerpo humano. Gracias a esta representación, es fácil de calcular colisiones entre cada una de las partes en que se ha dividido el cuerpo humano. Estas partes en que se ha dividido el cuerpo humano son:

- 2 brazos x 3 partes móviles = 6.
- 2 piernas x 3 partes móviles = 6.

- 1 cuerpo = 1.
- 1 cabeza = 1.

Con lo cual se dispone de 14 partes con las que se puede detectar colisión. En el caso de los brazos y las piernas, cada una de las partes está definida por dos esferas principales, inicial y final, con sus respectivos radios. A lo largo del eje que une estas dos esferas se definen un número determinado de esferas que engloben cada una de las partes a la cual modelan. En el caso del tronco, se utilizan 4 esferas principales colocadas en los hombros y las caderas, y se generan n esferas que engloban todo el tronco. En el caso de la cabeza, solo tenemos una esfera que engloba la cabeza. En total se obtienen las 14 partes.

La detección de colisión consiste en determinar, para cada uno de los conjuntos de esferas que modelan una parte del cuerpo humano, si hay colisión con algún otro conjunto de esferas que modelan otra de las partes restantes. La forma para comprobar la colisión radica en comprobar para cada una de las esferas que forman un conjunto, comprobar si existe colisión con otra esfera que forma el otro conjunto. Esto lleva a un elevado número de operaciones.

Para ello se ha definido una tabla donde se puede ver los casos con los que se puede encontrar colisión y en qué casos no, como por ejemplo las piernas es muy improbable que colisiones con la cabeza o incluso con el cuerpo. Otro caso es evitar repetir los cálculos de detección de colisión, por ejemplo si calculamos si el brazo izquierdo colisiona con el brazo derecho, no hace falta calcular si el brazo derecho colisiona con el izquierdo, ya que se obtendría el mismo resultado. La siguiente tabla, tabla 4.2, representa las posibles colisiones entre partes:

Tabla 4.2: Tabla de colisiones.

		brazo izquierdo			brazo derecho			pierna izquierda			pierna derecha			cuerpo	cabeza
		brazo	antebrazo	mano	brazo	antebrazo	mano	pierna	antepierna	pie	pierna	antepierna	pie	cuerpo	cabeza
brazo izquierdo	brazo	0	0	0	1	1	1	1	1	1	1	1	1	0	1
	antebrazo	0	0	0	1	1	1	1	1	1	1	1	1	1	1
	mano	0	0	0	1	1	1	1	1	1	1	1	1	1	1
brazo derecho	brazo	1	1	1	0	0	0	1	1	1	1	1	1	0	1
	antebrazo	1	1	1	0	0	0	1	1	1	1	1	1	1	1
	mano	1	1	1	0	0	0	1	1	1	1	1	1	1	1
pierna izquierda	pierna	1	1	1	1	1	1	0	0	0	1	1	1	0	0
	antepierna	1	1	1	1	1	1	0	0	0	1	1	1	0	0
	pie	1	1	1	1	1	1	0	0	0	1	1	1	0	0
pierna derecha	pierna	1	1	1	1	1	1	1	1	1	0	0	0	0	0
	antepierna	1	1	1	1	1	1	1	1	1	0	0	0	0	0
	pie	1	1	1	1	1	1	1	1	1	0	0	0	0	0
cuerpo	cuerpo	1	1	1	1	1	1	0	0	0	0	0	0	0	0
cabeza	cabeza	1	1	1	1	1	1	0	0	0	0	0	0	0	0

La tabla 4.2, muestra las colisiones entre las diferentes partes. Los cuadros en azul o con un 0 indican que no hay colisión mientras que los cuadros en blanco o con un 1 indican una posible colisión. De esta forma se ha reducido casi a la mitad el número de operaciones a realizar para detectar colisión.

En aquellos casos donde las partículas generadas dan como solución una posible colisión, estas partículas son eliminadas de la solución final, aumentando de esta forma la precisión del filtro de partículas.

4.4 Restricción 3: Proyección 2D de las poli-esferas

Obtenidas las partículas dadas, se puede calcular donde esta cada una de las variables de articulación en el espacio 3D, y a su vez representar mediante esferas cada una de las variables de articulación. Estas esferas pueden ser proyectadas al plano 2D obteniendo elipses como muestra la figura 4.2. Estas elipses representan sobre la imagen RGB dónde se encuentra cada variable de articulación deseada. Si unimos cada una de las esferas, de forma análoga a la vista anteriormente para definir cada una de las partes en que se a dividido el cuerpo humano, nos lleva a

una representación en 2D de la pose del cuerpo humano. Teniendo en cuenta que se tienen n soluciones, estas soluciones se pueden filtrar si para cada solución se observa la superposición de la pose con un plano de la imagen RGBD, eliminando las soluciones donde la superposición sea pobre.

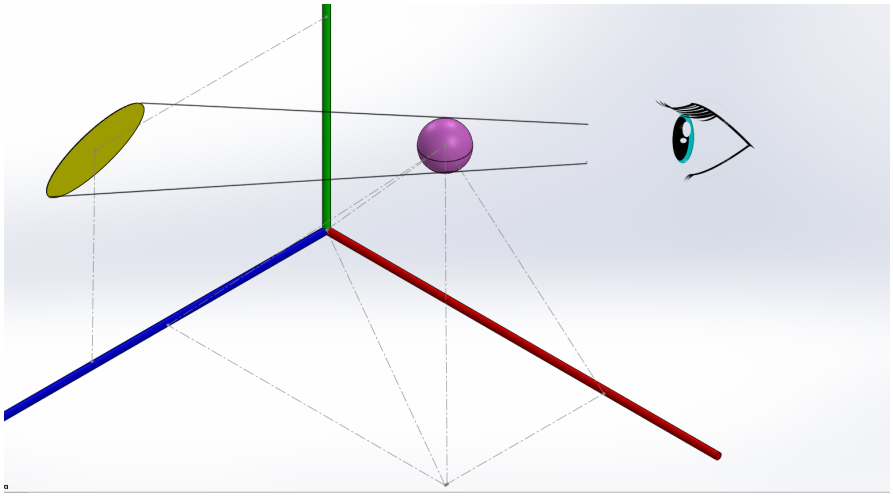


Figura 4.2: Proyección de una esfera sobre el plano 2D para obtener una elipse.

Para realizar estas operaciones, se necesita de ciertas operaciones básicas vistas en el anexo D. Con la ayuda de estos cálculos, se va a dar solución a los diferentes casos que se encuentran para representar la postura del esqueleto del cuerpo humano según la posición de las elipses.

Todos los siguientes cálculos ayudan a ponderar cada una de las soluciones según su solapamiento en la imagen RGBD. Esta restricción todavía no ha sido analizada. Con lo cual a falta de obtener los resultados, se dejarán estas soluciones para trabajo futuro.

4.4.1 Envolvente a dos elipses

Figura 4.3: Envolvente entre dos elipses.

Para calcular la envolvente a dos elipses tenemos que realizar diferentes pasos.

4.4.1.1 Paso 1: Tangentes a dos elipses y puntos de tangencia

Para calcular la envolvente de dos elipses hay que calcular las rectas tangentes entre ellas y obtener los puntos de intersección de las rectas con las elipses. Cómo realizar este cálculo ha sido visto en el anexo D, sección D.1.5.

4.4.1.2 Paso 2: Puntos exteriores a la nube de puntos obtenida

Una vez se disponen de los puntos de tangencia, se utiliza esta nube de puntos para calcular la envolvente. Se trata de seleccionar los puntos exteriores de la nube de puntos, es decir, seleccionar aquellos puntos que, trazando líneas rectas entre ellos, hacen que todos los otros puntos estén dentro del polígono formado. Para ello existe un comando de "MATLAB" que nos ayudara a realizar esta operación: "convhull".

El número de puntos tangentes será el doble al número de rectas tangentes, debido a que cada recta tiene dos puntos tangentes.

4.4.1.3 Paso 3: Punto medio de la recta que uno los centros de las elipses

Análogamente a los cálculos vistos en el anexo D, sección D.1.6, se calcula el punto medio de todos los centros de las elipses.

4.4.1.4 Paso 4: Ángulo que forman los puntos de tangencia con respecto a sus elipses

Cada punto de tangencia pertenece a una elipse en concreto, se trata de calcular el ángulo de este punto de tangencia con respecto al eje paralelo al eje OX que pasa por el centro de la elipse. Los cálculos necesarios han sido vistos en el anexo D, sección D.1.7.

4.4.1.5 Paso 5: Ángulo que forman los puntos de tangencia con respecto a un eje paralelo a OX que pasa por el punto medio de los centros de las elipses

Para cada punto de tangencia hay que calcular el ángulo que forma este con el eje paralelo a OX que pasa por el punto medio de los centros de las elipses. Estos cálculos pueden ser vistos en el anexo D, sección D.1.8.

Una vez calculado estos ángulos, se ordenan todos los puntos según este ángulo de forma creciente.

Con este procedimiento lo que se debe hacer es ordenar los puntos que van unidos entre sí, cada punto va unido con el punto siguiente y el anterior, de forma que cada punto va unido a dos más. Esta matriz presenta el siguiente aspecto:

$$\begin{bmatrix} \alpha_1 & c_{x_1} & c_{y_1} & \beta_1 & \text{elipse}_1 \\ \alpha_2 & c_{x_2} & c_{y_2} & \beta_2 & \text{elipse}_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_i & c_{x_i} & c_{y_i} & \beta_i & \text{elipse}_i \end{bmatrix} \tag{4.1}$$

donde α es el ángulo con respecto al centro de la recta que une las dos elipses, c_x y c_y son las coordenadas del centro de la elipse a la que pertenece el punto, β es el ángulo que forma el punto con respecto al centro de la elipse a la que pertenece, *elipse* es el número de la elipse a la que pertenece el punto tratado.

4.4.1.6 Paso 6: Calcular los puntos de cada tramo

Una vez obtenida la matriz, se dibujan los tramos deseados. Se unirá cada punto de la matriz anterior, con el punto anterior y siguiente, de esta forma cada punto pertenece a dos tramos, bien sea un tramo que pertenezca a una recta o bien un tramo que pertenezca a una elipse.

Para conocer si el tramo a dibujar es una recta o una elipse, se observa la matriz en la componente elipse_i. Si los dos puntos que definen un tramo, este valor es el mismo, entonces se trata de un tramo perteneciente a una elipse, si $\text{elipse}_1 = \text{elipse}_2 \Rightarrow$ tramo_elipse, se utilizarán los cálculos vistos en el anexo D, sección D.1.11. Si los dos puntos que definen un tramo, este valor es diferente, entonces se trata de un tramo perteneciente a una recta, si $\text{elipse}_1 \neq \text{elipse}_2 \Rightarrow$ tramo_recta, se utilizarán los cálculos vistos en la sección D.1.10.

Se obtienen dos vectores de coordenadas, un vector se corresponde con las coordenadas de las x, *vect_x*, el otro vector se corresponde con las coordenadas de las y, *vect_y*. Se utilizarán estos dos vectores para dibujar la envolvente de las elipses.

4.4.2 Envolvente a "N" elipses

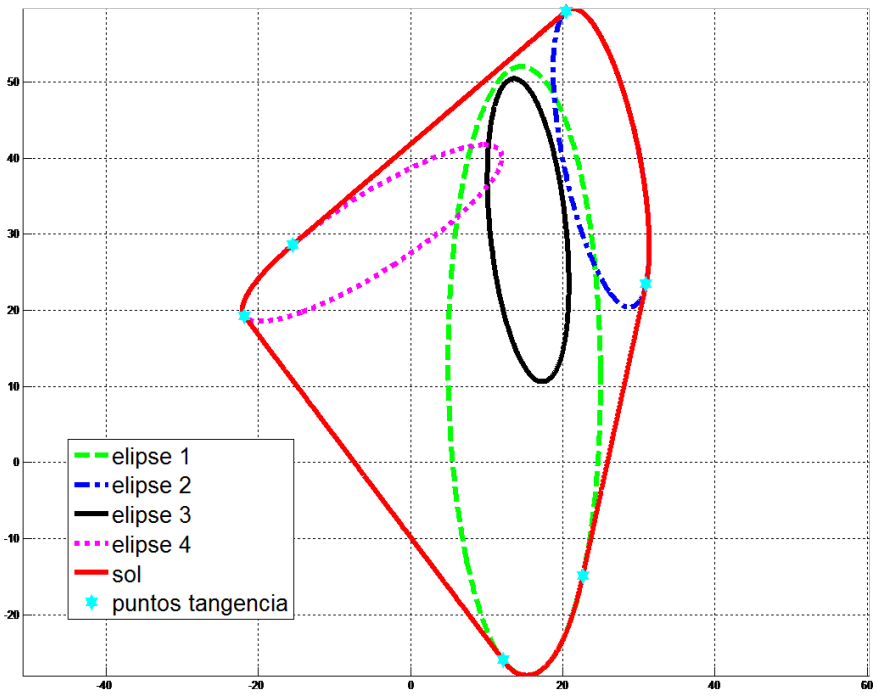


Figura 4.4: Envolvente entre "N" elipses.

Para calcular la envolvente a "n" elipses, el proceso es similar al visto en la sección 4.4.1. Los pasos a seguir son:

4.4.2.1 *Paso* 1*: Tangentes a las elipses entre ellas*

Calcular para cada elipse las rectas tangentes entre esta y todas las demás. Los cálculos a realizar han sido vistos en el anexo D, sección D.1.5. Se obtiene una nube de puntos tangentes para todas las elipses.

4.4.2.2 Paso 2: Puntos exteriores a la nube de puntos

De la nube de puntos obtenida, se seleccionarán los puntos exteriores a ella, es decir, se seleccionan aquellos puntos que, trazando líneas rectas entre ellos, hacen que todos los puntos restantes estén dentro del polígono formado. Para ello existe un comando de "MATLAB" que nos ayudara a realizar esta operación: "convhull". Se obtiene una matriz de los puntos deseados junto con la información de a que elipse pertenece cada punto.

4.4.2.3 Paso 3: Punto medio de los centros de todas las elipses

Análogamente a los cálculos vistos en el anexo D, sección D.1.6, se calcula el punto medio de todos los centros de las elipses.

4.4.2.4 Paso 4: Ángulo que forman los puntos de tangencia con respecto a sus elipses

Cada punto de tangencia pertenece a una elipse en concreto, se trata de calcular el ángulo de este punto de tangencia con respecto al eje paralelo al eje OX que pasa por el centro de la elipse. Los cálculos necesarios han sido vistos en el anexo D, sección D.1.7.

4.4.2.5 Paso 5: Ángulo que forman los puntos de tangencia con respecto a un eje paralelo a OX que pasa por el punto medio de los centros de las elipses

Para cada punto de tangencia hay que calcular el ángulo que forma este con el eje paralelo a OX que pasa por el punto medio de los centros de las elipses. Estos cálculos pueden ser vistos en el anexo D, sección D.1.8.

Una vez calculado estos ángulos, se ordenan todos los puntos según este ángulo de forma creciente.

Con este procedimiento lo que se hace es ordenar los puntos que van unidos entre sí, cada punto va unido con el punto siguiente y el anterior, de forma que cada punto va unido a dos más. Esta matriz presenta el siguiente aspecto:

$$\begin{bmatrix} \alpha_1 & c_{x_1} & c_{y_1} & \beta_1 & \text{elipse}_1 \\ \alpha_2 & c_{x_2} & c_{y_2} & \beta_2 & \text{elipse}_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_i & c_{x_i} & c_{y_i} & \beta_i & \text{elipse}_i \end{bmatrix} \tag{4.2}$$

donde α es el ángulo con respecto al eje paralelo a OX que pasa por el punto medio de los centro de todas las elipses, c_x y c_y son las coordenadas del centro de la elipse a la que pertenece el punto, β es el ángulo que forma el punto con respecto al eje paralelo a OX que pasa por el centro de la elipse a la que pertenece, *elipse* es el número de la elipse a la que pertenece el punto tratado.

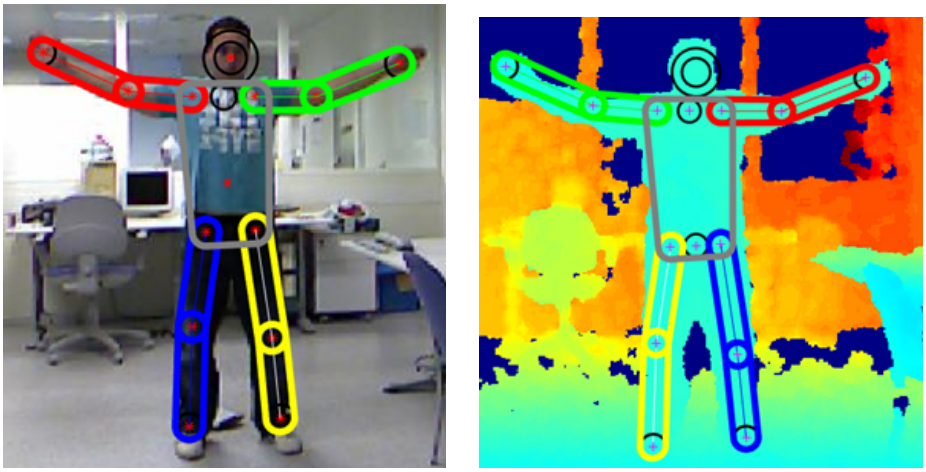
4.4.2.6 Paso 6: Calcular los puntos de cada tramo

Una vez obtenida la matriz, se dibujan los tramos deseados. Se unirá cada punto de la matriz anterior, con el punto anterior y siguiente, de esta forma cada punto pertenece a dos tramos, bien sea un tramo que pertenezca a una recta o bien un tramo que pertenezca a una elipse.

Para conocer si el tramo a dibujar es una recta o una elipse, se observa la matriz en la componente elipse_i. Si los dos puntos que definen un tramo, este valor es el mismo, entonces se trata de un tramo perteneciente a una elipse, si $\text{elipse}_1 = \text{elipse}_2 \Rightarrow$ tramo_elipse, se utilizarán los cálculos vistos en el anexo D, sección D.1.11. Si los dos puntos que definen un tramo, este valor es diferente, entonces se trata de un tramo perteneciente a una recta, si $\text{elipse}_1 \neq \text{elipse}_2 \Rightarrow$ tramo_recta, se utilizarán los cálculos vistos en el anexo D, sección D.1.10.

Se obtienen dos vectores de coordenadas, un vector se corresponde con las coordenadas de las x, *vect_x*, el otro vector se corresponde con las coordenadas de las y, *vect_y*. Se utilizan estos dos vectores para dibujar la envolvente de las elipses.

El cálculo de la envolvente a "n" elipses es utilizado para realizar la representación de la pose del esqueleto del cuerpo humano tal y como muestra la figura 4.5. Donde cada parte está formada por un conjunto de 2 elipses principales excepto el cuerpo que es representado por 4 elipses principales.



(a) (b)

Figura 4.5: Proyección de la envolvente de cada parte sobre la imagen RGB y de profundidad.

Tras obtener esta representación se puede realizar el estudio de la superposición de esta solución dada con respecto a la imagen de bordes, por ejemplo, para calcular la distancia de cada envolvente con respecto a los bordes mas cercanos, que definirá la superposición de la pose con la pose de bordes de la imagen.

4.4.3 Diferencia entre conjuntos de elipses

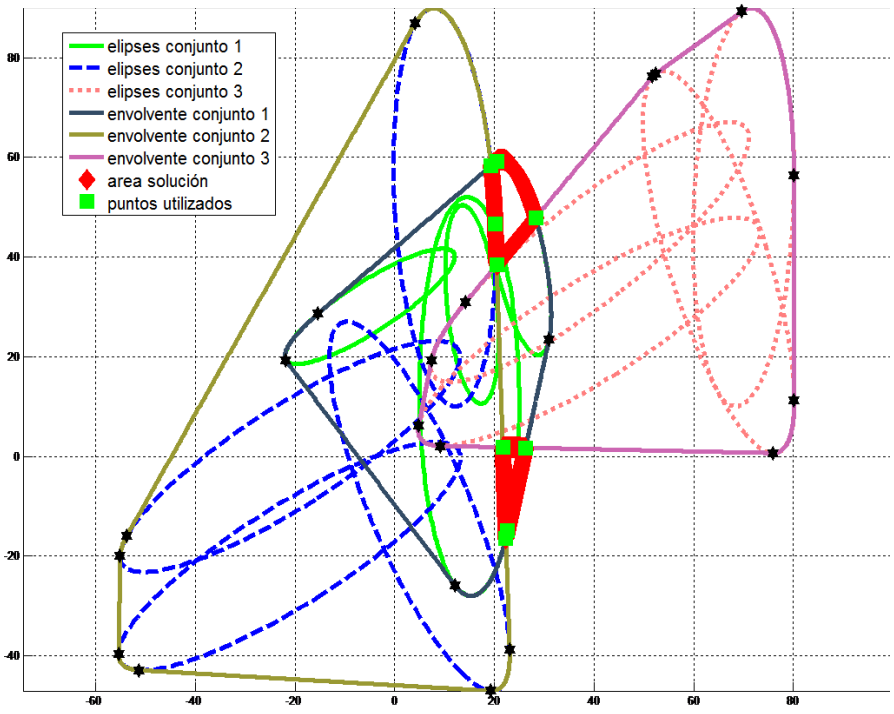


Figura 4.6: Diferencia entre conjuntos de elipses.

Dado "n" conjuntos de elipses, restar al conjunto principal todos los otros conjuntos y obtener el área del conjunto de elipses principal que no ha sido ocupada por ningún otro conjunto. Esta representación se utiliza para modelar las oclusiones sobre el plano 2D. Tras realizar las proyecciones de las esferas, se debe realizar la diferencia entre conjuntos para conocer de una forma visual qué parte del cuerpo esta por delante y qué parte del cuerpo está por detrás. De forma que si el brazo esta por delante del cuerpo en la proyección en el plano 2D de la imagen RGB, sea el brazo el que predomine sobre el área que define el cuerpo.

Los pasos a seguir para obtener la solución deseada son:

4.4.3.1 Paso 1: Obtener las envolventes de cada conjunto

Como se ha visto en la sección 4.4.2, para cada conjunto de elipses se tiene que calcular su envolvente. Esta solución aporta los puntos utilizados para calcular la envolvente de cada conjunto.

De estos puntos se tiene la información de a qué conjunto pertenecen, a qué elipse de cada conjunto pertenece, los ángulos de estos puntos con respecto al eje paralelo a OX que pasa por el centro de la elipse a la que pertenece y el ángulo con respecto al eje paralelo a OX que pasa por el punto medio de los centros de las elipses para el conjunto en que se encuentra. Esta información esta ordenada en tramos para cada conjunto de forma que es conocido si el tramo es una recta o una elipse.

4.4.3.2 Paso 2: Intersección con cada uno de los tramos de los polígonos

Una vez se tiene la envolvente para cada conjunto de elipses, se obtienen también todos los tramos ordenados de cada conjunto. Se tienen que calcular la intersección de cada uno de los tramos de un conjunto con todos los tramos de cada uno de los conjuntos de elipses restantes. SE coge el primer conjunto de elipses y se calcula la intersección de cada uno de sus tramos con los tramos de todos los otros conjuntos de elipses. Para realizar la intersección entre los diferentes tramos, se pueden encontrar 4 casos diferentes.

1. Intersección entre elipses:

   Para realizar la intersección entre elipses se utilizan los cálculos vistos en el anexo D, sección D.1.3.

2. Intersección entre rectas:

   Para realizar la intersección entre rectas se utilizan los cálculos vistos en el anexo D, sección D.1.2.

3. Intersección elipse-recta:

   Para realizar la intersección entre elipse - recta se utilizan los cálculos vistos en el anexo D, sección D.1.4.

4. Intersección recta-elipse:

   Para realizar la intersección entre recta - elipse se utilizan los cálculos vistos en el anexo D, sección D.1.4.

Una vez calculadas todas las intersecciones de todos los tramos de un conjunto de elipses, con todos los tramos de los conjuntos de elipses restantes, se obtiene una matriz de puntos con toda la información necesaria: conjunto al que pertenecen, tramo al que pertenece la intersección, en qué elipses se encuentra el tramo en cuestión, *cdots* esto ayudará a seleccionar los puntos deseados para poder dibujar el área resultante al realizar la diferencia entre conjuntos.

4.4.3.3 Paso 3: Descartar puntos no deseados

Para seleccionar qué puntos de intersección interesan y cuáles no, según el tipo de caso anterior se realizan diferentes cálculos.

1. Descartar puntos de la intersección entre elipses:

   Para descartar los puntos intersección entre elipses que no interesan, se calcula para cada punto intersección el ángulo que tiene este con respecto al eje paralelo a OX que pasa por el centro de la elipse, anexo D, sección D.1.7. Cada punto intersección pertenece a dos elipses, se tiene que calcular dicho ángulo con respecto a las dos elipses.

   Para eliminar los puntos deseados se dispone de: para cada elipse, ángulo inicial y final que pertenece a uno de los tramos para dibujar le envolvente, para cada punto intersección, el ángulo que forma este con sus dos respectivas elipses. Hay que comprobar que el ángulo del punto intersección está dentro de los dos rangos pertenecientes a los tramos de cada elipse.

   Es decir, se tiene un punto intersección entre dos elipses $p_1 = [x_1, y_1]$, se obtiene el ángulo con respecto al eje paralelo a OX que pasa por el centro de la elipse α, se cogen los rangos de las dos elipses que pertenecen a la envolvente $\text{ang\_el}_1 = [\text{ang\_in}_1, \text{ang\_fin}_1]$ y $\text{ang\_el}_2 = [\text{ang\_in}_2, \text{ang\_fin}_2]$, el punto intersección es deseado si se cumple que:

   $$\begin{cases} \text{ang\_in}_1 < \alpha < \text{ang\_fin}_1 \\ \text{ang\_in}_2 < \alpha < \text{ang\_fin}_2 \end{cases} \tag{4.3}$$

   Se descartan todos los puntos intersección entre elipses que no cumplan esta condición.

2. Descartar puntos de la intersección entre rectas:

   Tras calcular la intersección entre elipses, todos los puntos que están comprendidos entre los dos puntos extremos de la recta son correctos. De igual

forma que en la intersección entre elipses, se cogen los puntos correctos siempre y cuando estén entre los dos extremos de ambas rectas, en este caso no se descarta ningún punto ya que todos los puntos son correctos.

3. Descartar puntos de la intersección entre recta - elipse:

   Para descartar los puntos intersección entre recta - elipse, se tiene que realizar un procedimiento análogo al caso 1.

   En este caso se dispone de la recta r_1 definida por el punto inicial y punto final $\{p_1, p_2 | p_i = [x_i, y_i]\} / p_1, p_2 \in r_1$, del rango de la elipse que pertenece a la envolvente $\text{ang\_el}_1 = [\text{ang\_in}_1, \text{ang\_fin}_1]$ y el punto intersección $p_3 = [x_3, y_3]$. Se calcula el ángulo del punto intersección con respecto al eje paralelo a OX que pasa por el centro de la elipse α, anexo D, sección D.1.7. Un punto intersección es correcto si se cumple que:

   $$\begin{cases} p_3 \in r_1 / |p_1| < |p_3| < |p_2| \\ \text{ang\_in}_1 < \alpha < \text{ang\_fin}_1 \end{cases} \tag{4.4}$$

   Si no se cumplen estas dos condiciones, el punto intersección es descartado.

4. Descartar puntos de la intersección entre elipse - recta:

   Análogamente igual al caso 3. Se utilizará el mismo procedimiento para descartar los puntos no deseados.

Tras descartar todos los puntos no deseados, se obtiene una matriz de puntos intersección que van a ser utilizados para calcular los tramos que darán solución al área deseada.

4.4.3.4 Paso 4: Introducir los tramos derivados de las intersecciones

Los puntos intersección calculados cortan a los tramos ya definidos al calcular las envolventes de cada conjunto. Para cada punto intersección se conoce el conjunto al que pertenece y a que tramo pertenece. Se tiene que conocer a qué tramos cortan los puntos intersección, para cortar estos tramos y convertirlos en los nuevos.

Se parte de la matriz de puntos obtenida al calcular las envolventes, donde los puntos utilizados para calcular la envolvente están ordenados según el ángulo con respecto al eje paralelo a OX que pasa por el punto medio de los centros de todas las elipses. Esta matriz tiene la siguiente forma:

$$\begin{bmatrix}\alpha & c_x & c_y & ang & elip\end{bmatrix} \tag{4.5}$$

donde α indica el ángulo del punto con coordenadas c_x y c_y con respecto al eje OX que pasa por el centro del conjunto de las elipses. La variable *ang* indica el ángulo formado por el punto con coordenadas c_x y c_y con respecto al centro de la elipse al que pertenecen. La variable *elip* indica la elipse a la que pertenece el punto con coordenadas c_x y c_y.

Se dispone de la matriz de puntos intersección entre los conjuntos, esta matriz tiene la siguiente forma:

$$[c_x\ c_y\ tram_{11}\ tram_{12}\ tram_{21}\ tram_{22}\ ang_1\ ang_2\ tipo\_inter\ conj_1\ conj_2] \tag{4.6}$$

donde c_x y c_y son las coordenadas x e y respectivamente del punto intersección entre los conjuntos. Las variables $conj_1$ y $conj_2$ indican a qué dos conjuntos pertenece el punto intersección. Las variables $tram_{11}$ y $tram_{12}$ indican en qué tramo del $conj_1$ se encuentra el punto intersección. Las variables $tram_{21}$ y $tram_{22}$ indican en qué tramos del $conj_2$ se encuentra el punto intersección. Si $tram_{11} = tram12$ se trata de un tramo de la elipse $tram_{11}$, con lo cual ang_1 contiene el ángulo del punto intersección con respecto al eje paralelo al eje OX que pasa por el centro de la elipse. Si $tram_{11} \neq tram12$ se trata de una recta y el valor de la variable ang_1 es 0. Análogamente ocurre para las variables $tram_{21}$, $tram_{21}$ y ang_2. La variable *tipo_inter* indica el tipo de intersección (1 - elipses, 2 - rectas, 3 - elipse-recta, 4 - recta-elipse), es decir, entre qué tipo de tramos se produce la intersección.

Para cada punto intersección se calcula el ángulo formado con respecto al eje paralelo a OX que pasa por el punto medio de los centros de las elipses para el conjunto a los que pertenece, anexo D, sección D.1.8.

Conocido este ángulo, se introducen estos puntos intersección en la matriz de puntos obtenida al calcular las envolventes. Gracias a las variables $conj_1$ y $conj_2$ se puede conocer en qué conjunto debemos introducir los nuevos tramos.

El resultado es una nueva matriz que contiene los puntos iniciales de la envolvente con los nuevos puntos intersección.

Para conocer los nuevos tramos, se coge esta nueva matriz, y cada tramo estará formado por el punto actual y el punto siguiente en la matriz, donde cada uno de los puntos definirá dos tramos, donde acaba un tramo y donde empieza el siguiente.

Se repite este paso para cada uno de los conjuntos de elipses.

4.4.3.5 Paso 5: Seleccionar los tramos deseados

A partir de las matrices con los nuevos tramos, se tiene que descartar aquellos tramos no deseados. Para descartar estos tramos pueden darse dos casos: Caso 1) Se tiene que eliminar todos aquellos tramos de los conjuntos de elipses que no sean el principal, que no estén dentro del conjunto de elipses principal. Caso 2) Para todos aquellos tramos restantes al caso 1, comprobar que ningún tramo perteneciente a un conjunto de elipses dado, este dentro de otro conjunto de elipses a parte del conjunto de elipses principal.

Para conocer si un punto está dentro o fuera de un conjunto de elipses dado, la idea es coger el punto a tratar $p_1 = [x_1, y_1]$ y el punto medio a todos los centros de las elipses de ese conjunto $c_1 = [x_2, y_2]$ y crear la recta r_1 donde $p_1, c_1 \in r_1$. Calcular la intersección de la recta r_1 con todos los tramos que definen el conjunto de elipse dado. Pueden darse dos casos diferentes de intersección: intersección entre dos rectas, visto en el anexo D, sección D.1.2, o intersección entre recta - elipse, visto en el anexo D, sección D.1.4. En ambos casos la solución son los puntos intersección.

Una vez calculados los puntos intersección, se calcula la distancia de c_1 al punto p_1, $d_1 = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$, y la distancia de c_1 a todos los puntos intersección calculados, se obtiene d_2. Con la ayuda de los parámetros d_1 y d_2 se puede conocer si el punto está dentro o fuera del conjunto.

Para el caso 1, el punto p_1 tiene que estar dentro del conjunto de elipses principal, para ello la condición que se ha de cumplir es que por lo menos uno de los puntos intersección calculados tenga una distancia mayor al punto c_1, se debe cumplir que $d_1 < d_2$, si se cumple esta condición, el punto p_1 está dentro del conjunto de elipses principal, se trata de un punto deseado, en el caso contrario el punto se marcará como no deseado, y el tramo formado por ese punto será descartado.

Para el caso 2, se parte de los puntos resultantes del caso 1, se tiene que comprobar para cada punto p_1 que no esté dentro de ningún conjunto de puntos que no sea el principal, para ello ninguno de los puntos de intersección calculados para ese punto tiene que tener una distancia mayor al punto c_1, se debe cumplir que $d_1 > d_2$, si se cumple esta condición, el punto p_1 no está dentro de ningún otro conjunto de elipses, se trata de un punto deseado, en el caso contrario se marcará como no deseado, y el tramo formado por ese punto será descartado.

4.4.3.6 Paso 6: Calcular los puntos de cada tramo de las envolventes

Este proceso es igual al visto en el anexo D, sección 4.4.2.6.

Se trata de calcular todos los puntos para dibujar las envolventes para cada uno de los "n" conjuntos. Se utilizan todos los tramos de cada conjunto.

4.4.3.7 Paso 7: Calcular los puntos de los tramos resultantes como diferencia de los "n" conjuntos

Este proceso es igual al visto en el anexo D, sección 4.4.2.6.

En este caso los tramos a dibujar son los tramos resultantes como la diferencia de los "n" conjuntos.

La utilización de la diferencia entre conjunto de elipses es debido a que, utilizando el mismo fin visto en la utilización de la envolvente de "n" elipses, en este caso puede ser que una parte de la pose esta ocluida por algún objeto dentro de la escena, con lo que el estudio de la superposición de la envolvente con la imagen no seria correcta ya que hay objetos por delante, pero utilizando este método, si tenemos modelado la escena en la que nos encontramos, se puede eliminar aquellas partes de la envolvente calculada que estan ocluidas por algún objeto, tal y como muestra la figura 4.7.

Figura 4.7: Eliminación de la parte de la envolvente que esta ocluida por un objeto.

4.5 Utilización del filtro de partículas

Se ha implementado un filtro recursivo de estimación bayesiana. La solución bayesiana para realizar los posteriores cálculos de distribución, $p(x_t|Y_t)$, del vector de estados, dadas las observaciones pasadas, es dado por Konietschke y col. 2006:

$$\begin{aligned} p(x_{t+1}|Y_t) &= \int p(x_{t+1}|x_t)p(x_t|Y_t)dx_t = \\ &= \int p_{f_t}(B_f^{\dagger}(x_{t+1} - Ax_t - B_u u_t))p(x_t|Y_t)dx_t \end{aligned} \tag{4.7}$$

$$p(x_t|Y_t) = \frac{p(y_t|x_t)p(x_t|Y_{t-1})}{p(y_t|Y_{t-1})} = \frac{p_{e_t}(y_t - h(x_t))p(x_t|Y_{t-1})}{c_t} \tag{4.8}$$

$$\hat{x}_t^{MMS} = \int x_t p(x_t|Y_t)dx_t \tag{4.9}$$

$$P_t^{MMS} = \int (x_t - \hat{x}_t^{MMS})(x_t - \hat{x}_t^{MMS})^T p(x_t|Y_t)dx_t \tag{4.10}$$

donde † denota la pseudo-inversa de "Moore-Penrose", c_t es una constante de normalización y $\hat{x}_t^{MMS}$ la media mínima de estimación de cuadrados.

El algoritmo utilizado para la implementación del filtro de partículas es:

1. Inicialización: Se generan $x_0^i \sim p_{x_0}, i = 1, \ldots, N$. Cada muestra del vector de estado es referida como una partícula.

2. Actualizaciones de mesura: Se actualizan los pesos con la probabilidad (más generalmente, cada función de importancia)

$$w_t^i = w_{t-1}^i p(y_t|x_t^i) = w_{t-1}^i p_{e_t}(y_t - h(x_t^i)), i = 1, \ldots, N \tag{4.11}$$

   Y se normaliza para $w_t^i := w_t^i / \sum_i w_t^i$. Como una aproximación de la ecuación 4.9, se coge:

$$\hat{x}_t \approx \sum_{i=1}^{N} w_t^i x_t^i \tag{4.12}$$

3. "Re-Sampling":

   - "Bayesian bootstrap". Se cogen N muestras para reemplazarlas por el conjunto $\{x_t^i\}_{i=1}^N$ donde la probabilidad para coger la muestra i es w_t^i. Dado $w_t^i = 1/N$. Este paso es llamado "Sampling Importance Resampling" (SIR).

- "Importance Sampling". Solo se vuelve a muestrear como antes cuando el número efectivo de las muestras es inferior a un umbral N_{th}:

$$N_{eff} = \frac{1}{\sum_i (w_t^i)^2} < N_{th} \tag{4.13}$$

  Ver Doucet, De Freitas y Gordon 2001; Gonzalez y col. 2008; Kong, Liu y Wong 1994; Kruger, Lien y Verl 2009. Aquí $1 \leq N_{eff} \leq N$, donde el límite superior es alcanzado cuando todas las partículas tienen el mismo peso, y el límite inferior cuando toda la masa de probabilidad es una partícula. El umbral puede ser elegido como $N_{th} = 2N/3$.

4. Predicción: Se coge $f_t^i \sim p_{f_i}$ y se simula:

$$x_{t+1}^i = Ax_t^i + B_u u_t + B_f f_t^i, i = 1, \ldots, N \tag{4.14}$$

5. Entonces $t := t + 1$ y se itera al paso 2.

El punto clave con el "resampling" es prevenir la alta concentración de la masa de probabilidad en unas pocas partículas. Sin este paso, algunos w_t^i convergerán a 1 y el filtro cae en una simulación pura. El "resampling" puede ser eficientemente implementado utilizando el algoritmo clásico para el muestreo N.

Capítulo 5

Modelado de las cadenas cinemáticas utilizadas

En este capítulo se describe el modelo geométrico y cinemático utilizado para la representación de los resultados. El modelo cinemático se basa en la solución aportada por los cuaterniones duales, mientras que el modelo geométrico utiliza poli-esferas para la representación del modelo obtenido.

5.1 Introducción

Hoy en día la simulación de elementos mecánicos se hace imprescindible debido a la complejidad de los prototipos diseñados, así como del coste al que se arriesgan los creadores, por ello, poder comprobar en un ambiente de realidad virtual lo que sucedería con ciertos elementos, variándolos, y haciendo infinidad de pruebas, ahorra muchos recursos y cantidad de tiempo.

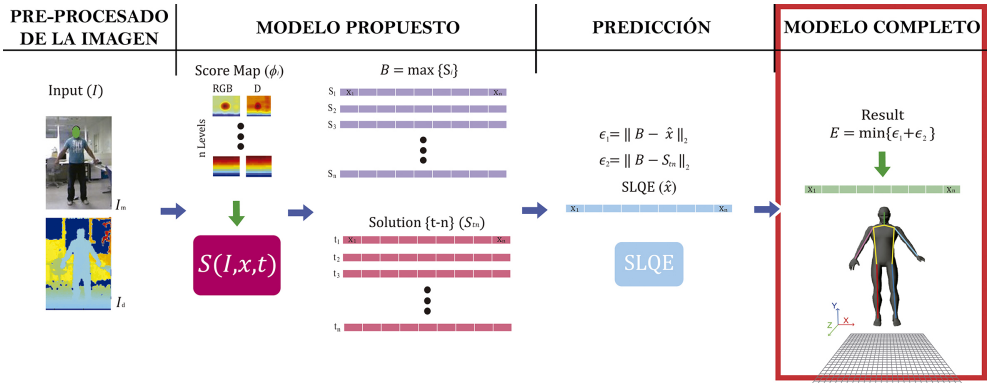


Figura 5.1: Introducción modelo geométrico y cinemático utilizado.

La figura 5.1 muestra la etapa en la que nos encontramos dentro de nuestro esquema.

Un simulador será necesario para:

- Observar el movimiento de los mecanismos
- Reproducir cada una de las partes del sistema o proceso de la forma más realista posible, pero siempre manteniendo bajo control la complejidad del modelo.

Actualmente existen herramientas de simulación virtual ajustadas al nivel del usuario, con una interfaz entendible y previsible, la cual facilita el trabajo con el software y el control por parte del usuario, y no es necesario conocer lenguajes de programación informáticos ni adquirir conocimientos avanzados.

Este capítulo tiene como objetivo realizar el modelado del esqueleto del cuerpo humano. Dicho modelado servirá para realizar las simulaciones oportunas y poder realizar los cálculos para la cinemática directa e inversa de cada una de las cadenas cinemáticas utilizando álgebra de cuaterniones. De esta forma se puede realizar simulaciones utilizando la cinemática y poder comprobar el correcto funcionamiento del mismo. Al mismo tiempo, el modelado utilizado servirá para realizar la detección de colisiones entre las diferentes partes del cuerpo.

Primero, en la sección 5.2 se describe el modelo geométrico utilizado para definir el esqueleto del cuerpo humano y el modelo cinemático utilizando cuaterniones duales aplicado al esqueleto del cuerpo humano tratando a este como un conjunto

de cadenas cinemáticas. Una vez definido el modelado del esqueleto del cuerpo humano, en la sección 5.3 se aplica los cuaterniones duales a la cadena cinemática de un brazo, los cálculos previos necesarios para dar solución a la cinemática los encontramos en el anexo E.

5.2 Modelado

El modelado del cuerpo humano no es nada trivial. Por eso siempre existen ciertas simplificaciones con lo que respecta al movimiento que un humano puede realizar. Para ello se ha realizado el modelado geométrico y cinemático del esqueleto del cuerpo humano.

5.2.1 Modelado geométrico

Uno de los objetivos de la presente tesis es la detección de colisiones entre las diferentes partes del cuerpo humano. Por esta razón se ha decidido realizar una aproximación al modelo del cuerpo humano utilizando poli-esferas. La utilización de poli-esferas tiene su justificación debido a que es sencillo realizar detección de colisiones entre esferas.

Para realizar el modelado geométrico se ha optado por dividir el esqueleto del cuerpo humano en diferentes cadenas cinemáticas. De esta forma se puede utilizar la ventaja de la aproximación a estas cadenas cinemáticas posteriormente para realizar el control del esqueleto del cuerpo humano.

Las cadenas cinemáticas en las que se ha dividido el esqueleto del cuerpo humano son las siguientes, Figura 5.2:

- Brazo derecho.
- Brazo izquierdo.
- Pierna derecha.
- Pierna izquierda.
- Tronco.

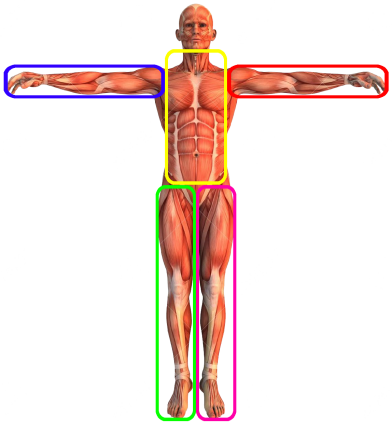


Figura 5.2: Partes en que hemos dividido el cuerpo.

Una vez dividido el esqueleto del cuerpo humano en las cadenas cinemáticas anteriormente citadas, ahora cada una de estas cadenas cinemáticas es dividida otra vez en varias partes. Lo que se ha realizado es dividir cada uno de las cadenas cinemáticas en tantas partes móviles como contenga:

- Brazos: 3 partes x brazo
- Piernas: 3 partes x pierna
- Cuerpo: 1 parte.
- Cabeza: 1 parte.

En la figura 5.3 se observa cada una de estas partes:

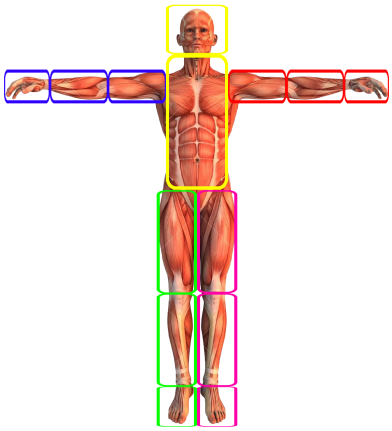


Figura 5.3: Partes en que se ha dividido cada cadena cinemática.

Para cada parte se necesitan dos puntos, punto inicial y punto final, donde en cada uno de estos puntos se dibujará una esfera. Para crear el modelo en poli-esferas lo que se realiza es una interpolación entre estas dos esferas, lo que conseguimos es unir estas dos esferas mediante un número infinito de esferas para envolver por completo la parte representada. Hay que realizar este procedimiento para cada una de las partes de cada una de las cadenas cinemáticas.

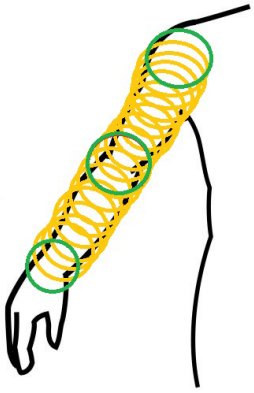


Figura 5.4: Modelado geométrico del brazo utilizando poli-esferas.

La figura 5.4 muestra un ejemplo de como funciona el modelo geométrico utilizando poli-esferas para una de las cadenas cinemáticas. En verde se encuentran

las esferas principales y en amarillo las "n" esferas creadas para envolver cada una de las partes de la cadena cinemática.

La representación final del esqueleto del cuerpo humano se observa en la figura 5.5:

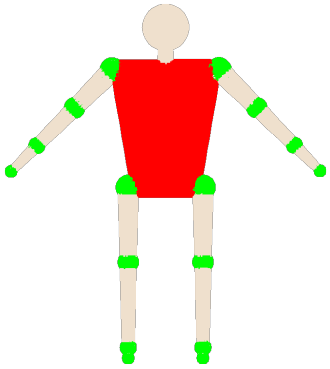


Figura 5.5: Modelo del esqueleto del cuerpo humano.

De la misma forma que se ha modelado el esqueleto del cuerpo humano utilizando poli-esferas, también se dispone de la representación del esqueleto del cuerpo humano utilizando líneas tal y como se observa en la figura 5.6. Esta representación consiste en unir en líneas cada una de las esferas que forman cada cadena cinemática.

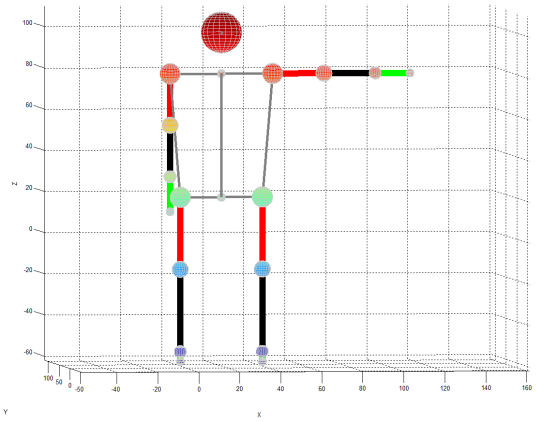


Figura 5.6: Modelo del cuerpo humano utilizando líneas.

5.2.2 Modelado cinemático - Cuaterniones duales

Para realizar el modelado cinemático de cada una de las cadenas cinemáticas en que se ha dividido el esqueleto del cuerpo humano se ha utilizado el álgebra de cuaterniones. En concreto se ha utilizado los cuaterniones duales para dar solución a la cinemática. Los cálculos previos necesarios se desarrollan en el anexo E. En la sección 5.3 de este capítulo, se aplican los cálculos necesarios para dar solución a una cadena cinemática.

Para dar solución a la cinemática utilizando cuaterniones duales, es necesario definir los sistemas de coordenadas que se van a utilizar. Anteriormente, en la figura 5.3, se ha visto en cuantas partes se ha dividido cada una de las cadenas cinemáticas. Si se entra más en detalle, se puede ver que realmente se tienen 3 cadenas cinemáticas diferentes:

- Brazos.
- Piernas.
- Tronco.

Esto es debido a que ambas cadenas cinemáticas que representan a los brazos son iguales, lo mismo se produce en las piernas.

Según se ha visto en el capítulo anterior, capítulo 3, el modelo DPM reducido utiliza solo 10 partes, y de estas 10 partes, 8 de ellas coinciden con las variables de articulación de los brazos y de las piernas. Gracias a esta información, se puede utilizar la información 3D de estas partes, gracias a la utilización de la imagen de profundidad, para dar solución a la cinemática inversa utilizando cuaterniones duales, de esta forma se puede inferir en el número de partes total y obtener las 14 partes necesarias para la estimación de la postura del esqueleto del cuerpo humano.

A continuación, se observa qué variables de estado y sistemas de coordenadas se han utilizado para cada uno de estas cadenas cinemáticas.

5.2.2.1 Sistemas de coordenadas y variables de estado

Para cada una de las cadenas cinemáticas, se ha optado en utilizar 6 grados de libertad, con lo que se tiene 6 sistemas de coordenadas. De esta forma se puede dar cobertura a los movimientos gruesos, que cambian la apariencia de forma significativa, que puede realizar el esqueleto del cuerpo humano.

5.2.2.2 Sistemas de coordenadas para los brazos

La figura 5.7 muestra los sistemas de coordenadas que se utilizan para los dos brazos:

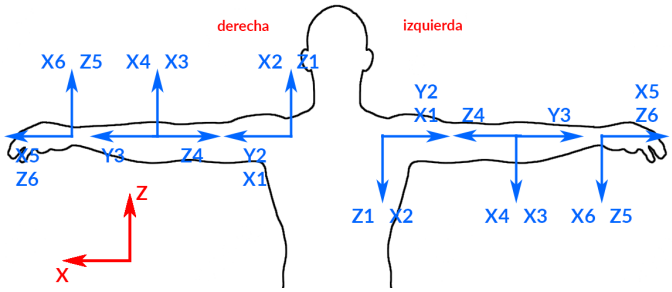


Figura 5.7: Sistemas de Coordenadas de los brazos.

5.2.2.3 Sistemas de coordenadas para las piernas

La figura 5.8 muestra los sistemas de coordenadas que se utilizan para las dos piernas:

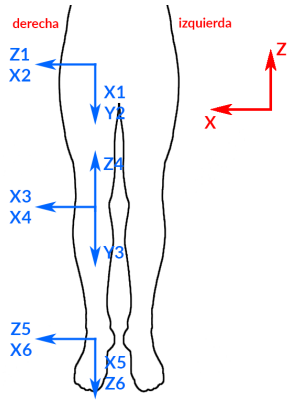


Figura 5.8: Sistemas de Coordenadas de las piernas.

5.2.2.4 Sistemas de coordenadas para el tronco

La figura 5.9 muestra los sistemas de coordenadas que se utiliza para el tronco:

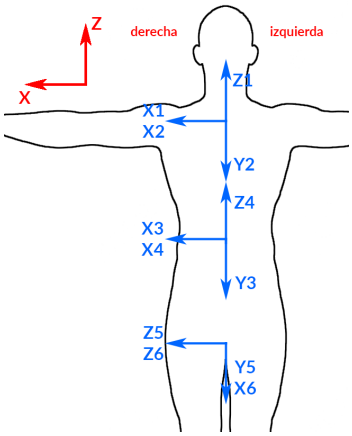


Figura 5.9: Sistemas de Coordenadas del tronco.

En esta parte aparecen dos puntos los cuales no tenemos información suministrada por el modelo DPM utilizado. Estos puntos son los dos extremos. Estos puntos son calculados como los puntos medios que forman las rectas que unen los puntos de los hombros y de las caderas. Aunque disponemos de las coordenadas del tronco, estas ahora no se utilizan el filtro de partículas.

5.3 Aplicación del modelo cinemático utilizando cuaterniones duales

Se va a aplicar los cálculos realizados a una cadena cinemática, en concreto a uno de los brazos mostrado en la figura 5.10, aunque esta imagen la hemos visto anteriormente, la recordamos de nuevo:

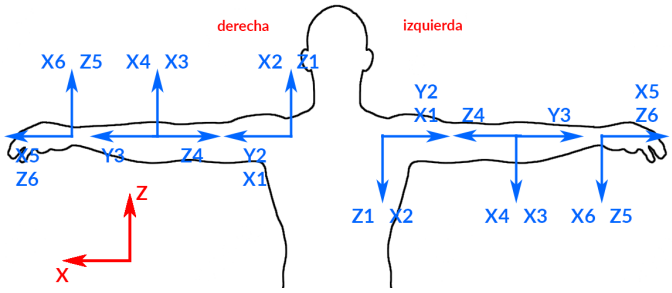


Figura 5.10: Sistemas de Coordenadas de los brazos.

5.3.1 Cinemática directa

Primero, se determinan los ejes de articulación y se calculan los momentos de los vectores de todos los ejes. Para el caso de la cadena cinemática de los brazos, los vectores de los ejes de articulación pueden ser observados en la figura 5.10 anterior.

Se ha definido la posición inicial de la cadena cinemática con los brazos extendidos, con lo cual, las articulaciones están sobre los siguientes ejes:

$$\begin{array}{lll} d_1 = [0,0,1] & d_2 = [0,1,0] & d_3 = [0,1,0] \\ d_4 = [1,0,0] & d_5 = [0,0,1] & d_6 = [1,0,0] \end{array} \tag{5.1}$$

Y un punto de cada uno de los ejes de articulación puede ser escrito como sigue:

$$\begin{array}{lll} p_1 = [a,0,0] & p_2 = [a,0,0] & p_3 = [b,0,0] \\ p_4 = [b,0,0] & p_5 = [c,0,0] & p_6 = [c,0,0] \end{array} \tag{5.2}$$

donde a es la distancia del sistema de coordenadas base a los sistema de coordenadas 1 o 2, b es la distancia de los sistema de coordenadas 3 o 4 al sistema de coordenadas 1 o 2, c es la distancia de los sistemas de coordenadas 5 o 6 a los sistemas de coordenadas 3 o 4.

Ahora se calculan los momentos de los vectores de los ejes de articulación:

$$\begin{array}{lll} m_1 = p_1 \times d_1 & m_2 = p_2 \times d_2 & m_3 = p_3 \times d_3 \\ m_4 = p_4 \times d_4 & m_5 = p_5 \times d_5 & m_6 = p_6 \times d_6 \end{array} \tag{5.3}$$

Posteriormente se definen cada uno de los ejes de articulación en forma de línea utilizando las coordenadas de "Plücker" en forma de cuaterniones duales:

$$\hat{l} = d + \varepsilon m \tag{5.4}$$

Y para cada una de las articulaciones:

$$\begin{array}{lll} \hat{l}_1 = d_1 + \varepsilon m_1 & \hat{l}_2 = d_2 + \varepsilon m_2 & \hat{l}_3 = d_3 + \varepsilon m_3 \\ \hat{l}_4 = d_4 + \varepsilon m_4 & \hat{l}_5 = d_5 + \varepsilon m_5 & \hat{l}_6 = d_6 + \varepsilon m_6 \end{array} \tag{5.5}$$

El operador de transformación que está en forma de cuaternión dual puede ser escrito utilizando los ejes de articulación, los vectores de momento y la ecuación (E.57). Finalmente, la ecuación de la cinemática directa para una cadena cinemática puede ser obtenida por:

$$\begin{array}{l} \hat{l}'_6 = l'_6 + \varepsilon l'^0_6 = \hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16} = \hat{q}_{16} \odot (l_6 + \varepsilon l^0_6) \odot \hat{q}^*_{16} \\ \hat{l}'_5 = l'_5 + \varepsilon l'^0_5 = \hat{q}_{15} \odot \hat{l}_5 \odot \hat{q}^*_{15} = \hat{q}_{15} \odot (l_5 + \varepsilon l^0_5) \odot \hat{q}^*_{15} \end{array} \tag{5.6}$$

donde $\hat{q}_{16} = \hat{q}_1 \odot \hat{q}_2 \odot \hat{q}_3 \odot \hat{q}_4 \odot \hat{q}_5 \odot \hat{q}_6$ y $\hat{q}_{15} = \hat{q}_1 \odot \hat{q}_2 \odot \hat{q}_3 \odot \hat{q}_4 \odot \hat{q}_5$. Con esto se tiene que la orientación del efector final es $\hat{l}'_6$ y la posición del efector final es:

$$\begin{array}{r} p_6 = (V\{R\{\hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16}\}\}) \times (V\{D\{\hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16}\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_{15} \odot \hat{l}_5 \odot \hat{q}^*_{15}\}\}) \times \\ (V\{D\{\hat{q}_{15} \odot \hat{l}_5 \odot \hat{q}^*_{15}\}\}) \end{pmatrix} \cdot (V\{R\{\hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16}\}\}) \right) * \\ *(V\{R\{\hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16}\}\}) \end{array} \tag{5.7}$$

Lo que se pretende es, conocidas todas las variables de articulación se puede conocer la posición de todos los puntos definidos para cada uno de los ejes, de esta forma, la ecuación (5.7) pretende encontrar la intersección entre los ejes 5 y 6 conociendo la orientación de estos ejes a través de todas las variables de articulación de los ejes anteriores, de forma que la intersección de estos ejes es el punto deseado, posición del efector final o punto de muñeca.

5.3.2 Cinemática inversa

En el problema de la cinemática inversa de una cadena cinemática, se dispone de la posición y orientación del efector final como parámetros de entrada:

$$\hat{q}_{in} = \begin{bmatrix} q_{in} \\ q^0_{in} \end{bmatrix} = \begin{bmatrix} \hat{l}'_6 \\ p_6 \end{bmatrix} \tag{5.8}$$

donde $q_{in} = [q_0, q_1, q_2, q_3]$, orientación del efector final, es la parte real del cuaternión dual $\hat{q}_{in}$. Y $q^0_{in} = [q^0_0, q^0_1, q^0_2, q^0_3]$, posición del efector final, parte dual del cuaternión dual $\hat{q}_{in}$.

Con lo cual observando las ecuaciones (5.6) y (5.7):

$$\begin{gathered} \hat{l}'_6 = R\{\hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16}\} = q_{in} \\ (V\{R\{\hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16}\}\}) \times (V\{D\{\hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16}\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_{15} \odot \hat{l}_5 \odot \hat{q}^*_{15}\}\}) \times \\ (V\{D\{\hat{q}_{15} \odot \hat{l}_5 \odot \hat{q}^*_{15}\}\}) \end{pmatrix} \cdot (V\{R\{\hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16}\}\}) \right) * \\ *(V\{R\{\hat{q}_{16} \odot \hat{l}_6 \odot \hat{q}^*_{16}\}\}) = q^0_{inV} \end{gathered} \tag{5.9}$$

El problema general de la cinemática inversa debe ser convertido en los apropiados sub-problemas de "Paden-Kahan" para obtener la solución a la cinemática inversa. Esta solución puede ser obtenida como sigue:

5.3.2.1 Cálculo de θ_3

La primera variable de articulación a calcular es θ_3, esto se debe a que la posición de la muñeca viene dada solamente por las tres primeras variables de articulación debido a que las restantes variables de articulación solamente orientan el efector final.

Primero se ponen dos puntos en la intersección de los ejes. El primer punto es p_w, intersección de los ejes de la muñeca (ejes 5 y 6). El segundo punto es p_b que es la intersección de los ejes de articulación 1 y 2.

De forma que, conociendo que la posición de la muñeca solo se ve afectada por las tres primeras articulaciones, la posición del punto p_w se puede escribir como:

$$\begin{array}{c}(V\{R\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})\times(V\{D\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})+\\ +\left(\begin{pmatrix}(V\{R\{\hat{q}_{13}\odot\hat{l}_5\odot\hat{q}^*_{13}\}\})\times\\(V\{D\{\hat{q}_{13}\odot\hat{l}_5\odot\hat{q}^*_{13}\}\})\end{pmatrix}\cdot(V\{R\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})\right)*\\ *(V\{R\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})=q^0_{inV}\end{array}\tag{5.10}$$

Y de forma similar, para el punto p_b:

$$\begin{array}{c}(V\{R\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})\times(V\{D\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})+\\ +\left(\begin{pmatrix}(V\{R\{\hat{q}_{12}\odot\hat{l}_1\odot\hat{q}^*_{12}\}\})\times\\(V\{D\{\hat{q}_{12}\odot\hat{l}_1\odot\hat{q}^*_{12}\}\})\end{pmatrix}\cdot(V\{R\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})\right)*\\ *(V\{R\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})=p_b\end{array}\tag{5.11}$$

La posición del punto p_b es libre de los ángulos de las dos primeras articulaciones. Si se resta la ecuación (5.10) con la ecuación (5.11), se tiene:

$$\begin{array}{c}\left(\begin{array}{c}(V\{R\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})\times(V\{D\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})+\\ +\left(\begin{pmatrix}(V\{R\{\hat{q}_{13}\odot\hat{l}_5\odot\hat{q}^*_{13}\}\})\times\\(V\{D\{\hat{q}_{13}\odot\hat{l}_5\odot\hat{q}^*_{13}\}\})\end{pmatrix}\cdot(V\{R\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})\right)*\\ *(V\{R\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})\end{array}\right)-\\ \left(\begin{array}{c}(V\{R\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})\times(V\{D\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})+\\ +\left(\begin{pmatrix}(V\{R\{\hat{q}_{12}\odot\hat{l}_1\odot\hat{q}^*_{12}\}\})\times\\(V\{D\{\hat{q}_{12}\odot\hat{l}_1\odot\hat{q}^*_{12}\}\})\end{pmatrix}\cdot(V\{R\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})\right)*\\ *(V\{R\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})\end{array}\right)\end{array}=q^0_{in}-p_b\tag{5.12}$$

Usando la propiedad de la distancia entre dos puntos preservados por los movimientos rígidos, se coge la magnitud de las dos partes de la ecuación 5.12, se tiene:

$$\left\|\begin{array}{c}\left(\begin{array}{c}(V\{R\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})\times(V\{D\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})+\\ +\left(\begin{pmatrix}(V\{R\{\hat{q}_{13}\odot\hat{l}_5\odot\hat{q}^*_{13}\}\})\times\\(V\{D\{\hat{q}_{13}\odot\hat{l}_5\odot\hat{q}^*_{13}\}\})\end{pmatrix}\cdot(V\{R\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})\right)*\\ *(V\{R\{\hat{q}_{13}\odot\hat{l}_6\odot\hat{q}^*_{13}\}\})\end{array}\right)-\\ \left(\begin{array}{c}(V\{R\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})\times(V\{D\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})+\\ +\left(\begin{pmatrix}(V\{R\{\hat{q}_{12}\odot\hat{l}_1\odot\hat{q}^*_{12}\}\})\times\\(V\{D\{\hat{q}_{12}\odot\hat{l}_1\odot\hat{q}^*_{12}\}\})\end{pmatrix}\cdot(V\{R\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})\right)*\\ *(V\{R\{\hat{q}_{12}\odot\hat{l}_2\odot\hat{q}^*_{12}\}\})\end{array}\right)\end{array}\right\|=\parallel q^0_{in}-p_b\parallel\tag{5.13}$$

La ecuación (5.13) es igual que:

$$\left\| \begin{pmatrix} (V\{R\{\hat{q}_{13} \odot \hat{l}_6 \odot \hat{q}_{13}^*\}\}) \times (V\{D\{\hat{q}_{13} \odot \hat{l}_6 \odot \hat{q}_{13}^*\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_{13} \odot \hat{l}_5 \odot \hat{q}_{13}^*\}\}) \times \\ (V\{D\{\hat{q}_{13} \odot \hat{l}_5 \odot \hat{q}_{13}^*\}\}) \end{pmatrix} \cdot (V\{R\{\hat{q}_{13} \odot \hat{l}_6 \odot \hat{q}_{13}^*\}\}) \right) * \\ *(V\{R\{\hat{q}_{13} \odot \hat{l}_6 \odot \hat{q}_{13}^*\}\}) \end{pmatrix} - \begin{pmatrix} (V\{R\{\hat{l}_2\}\}) \times (V\{D\{\hat{l}_2\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{l}_1\}\}) \times \\ (V\{D\{\hat{l}_1\}\}) \end{pmatrix} \cdot (V\{R\{\hat{l}_2\}\}) \right) * \\ *(V\{R\{\hat{l}_2\}\}) \end{pmatrix} \right\| = \| \, q_{in}^0 - p_b \, \| \tag{5.14}$$

Esto se debe a que, tal y como se ha dicho antes, la intersección de los dos primeros ejes de articulación es libre de los ángulos de los ejes de articulación, con lo cual la posición de este punto siempre va a ser la misma.

Esta ecuación se obtiene como resultado el sub-problema 3 de "Paden-Kahan". Los parámetros para este sub-problema son:

$$a = \begin{pmatrix} (V\{R\{\hat{l}_6\}\}) \times (V\{D\{\hat{l}_6\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{l}_5\}\}) \times \\ (V\{D\{\hat{l}_5\}\}) \end{pmatrix} \cdot (V\{R\{\hat{l}_6\}\}) \right) * \\ *(V\{R\{\hat{l}_6\}\}) \end{pmatrix}$$

$$b = \begin{pmatrix} (V\{R\{\hat{l}_2\}\}) \times (V\{D\{\hat{l}_2\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{l}_1\}\}) \times \\ (V\{D\{\hat{l}_1\}\}) \end{pmatrix} \cdot (V\{R\{\hat{l}_2\}\}) \right) * \\ *(V\{R\{\hat{l}_2\}\}) \end{pmatrix} \tag{5.15}$$

donde el parámetro l es el eje de la articulación 3 que es d_3 y el valor $\delta = q_{in}^0 - p_b$. Con la ayuda de estos parámetros y utilizando el sub-problema 3 se puede encontrar la solución a la variable de articulación θ_3.

5.3.2.2 Cálculo de θ_1 y θ_2

Si en la ecuación 5.10 se conoce θ_3, entonces se puede obtener:

$$\begin{pmatrix} (V\{R\{\hat{q}_{12} \odot \hat{l}'_6 \odot \hat{q}_{12}^*\}\}) \times (V\{D\{\hat{q}_{12} \odot \hat{l}'_6 \odot \hat{q}_{12}^*\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_{12} \odot \hat{l}'_5 \odot \hat{q}_{12}^*\}\}) \times \\ (V\{D\{\hat{q}_{12} \odot \hat{l}'_5 \odot \hat{q}_{12}^*\}\}) \end{pmatrix} \cdot (V\{R\{\hat{q}_{12} \odot \hat{l}'_6 \odot \hat{q}_{12}^*\}\}) \right) * \\ *(V\{R\{\hat{q}_{12} \odot \hat{l}'_6 \odot \hat{q}_{12}^*\}\}) \end{pmatrix} = q_{inV}^0 \tag{5.16}$$

donde $\hat{l}'_6 = \hat{q}_3 \odot \hat{l}_6 \odot \hat{q}_3^*$ y $\hat{l}'_5 = \hat{q}_3 \odot \hat{l}_5 \odot \hat{q}_3^*$

Con esta ecuación se obtiene un nuevo sub-problema de "Paden-Kahan", sub-problema 2, donde los parámetros son:

$$a = \begin{pmatrix} (V\{R\{\hat{l}_6'\}\}) \times (V\{D\{\hat{l}_6'\}\})+ \\ + \left(\begin{pmatrix} (V\{R\{\hat{l}_5'\}\})\times \\ (V\{D\{\hat{l}_5'\}\}) \end{pmatrix} \cdot (V\{R\{\hat{l}_6'\}\}) \right) * \\ *(V\{R\{\hat{l}_6'\}\}) \end{pmatrix} \tag{5.17}$$

donde el parámetro l_1 es el eje de la articulación 1 que es d_1, el parámetro l_2 es el eje de la articulación 2 que es d_2 y el valor $b = q_{in}^0$. Con la ayuda de estos parámetros y utilizando el sub-problema 2 se puede encontrar la solución a las variables de articulación θ_1 y θ_2.

5.3.2.3 Cálculo de θ_4 y θ_5

Para encontrar los ángulos de la muñeca, se tiene que considerar un nuevo punto $p_i = p_6 + \lambda d_6$, punto inicial, que esté sobre el eje de articulación d_6. Para encontrar el punto final p_e hacen falta dos ejes imaginarios para que este punto sea la posición del punto p_i después de la rotación de los ángulos θ_4 y θ_5. El punto p_i es la intersección de los dos ejes imaginarios. Con lo que se va a definir dos ejes imaginarios que estén sobre d_6 y que intersecten en el punto p_i dado por:

$$\begin{array}{rr} d_7 = [0,1,0] & d_8 = [0,0,1] \\ p_7 = p_i & p_8 = p_i \\ m_7 = p_7 \times d_7 & m_8 = p_8 \times d_8 \\ \hat{l}_7 = d_7 + \varepsilon m_7 & \hat{l}_8 = d_8 + \varepsilon m_8 \end{array} \tag{5.18}$$

Con lo cual, la posición del punto p_e puede ser encontrada por:

$$\begin{pmatrix} (V\{R\{\hat{q}_{16} \odot \hat{l}_8 \odot \hat{q}_{16}^*\}\}) \times (V\{D\{\hat{q}_{16} \odot \hat{l}_8 \odot \hat{q}_{16}^*\}\})+ \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_{16} \odot \hat{l}_7 \odot \hat{q}_{16}^*\}\})\times \\ (V\{D\{\hat{q}_{16} \odot \hat{l}_7 \odot \hat{q}_{16}^*\}\}) \end{pmatrix} \cdot (V\{R\{\hat{q}_{16} \odot \hat{l}_8 \odot \hat{q}_{16}^*\}\}) \right) * \\ *(V\{R\{\hat{q}_{16} \odot \hat{l}_8 \odot \hat{q}_{16}^*\}\}) \end{pmatrix} = q_{in}^0 + \lambda d_6 \tag{5.19}$$

Desde el punto p_i que esta sobre el eje d_6, la última articulación no afecta a la posición del punto p_i. Con lo que la ecuación (E.50) es igual a:

$$\begin{pmatrix} (V\{R\{\hat{q}_{13} \odot \hat{q}_{45} \odot \hat{l}_8 \odot \hat{q}_{45}^* \odot \hat{q}_{13}^*\}\})\times \\ \times(V\{D\{\hat{q}_{13} \odot \hat{q}_{45} \odot \hat{l}_8 \odot \hat{q}_{45}^* \odot \hat{q}_{13}^*\}\})+ \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_{13} \odot \hat{q}_{45} \odot \hat{l}_7 \odot \hat{q}_{45}^* \odot \hat{q}_{13}^*\}\})\times \\ \times(V\{D\{\hat{q}_{13} \odot \hat{q}_{45} \odot \hat{l}_7 \odot \hat{q}_{45}^* \odot \hat{q}_{13}^*\}\}) \end{pmatrix} \cdot \right) * \\ \cdot(V\{R\{\hat{q}_{13} \odot \hat{q}_{45} \odot \hat{l}_8 \odot \hat{q}_{45}^* \odot \hat{q}_{13}^*\}\}) \\ *(V\{R\{\hat{q}_{13} \odot \hat{q}_{45} \odot \hat{l}_8 \odot \hat{q}_{45}^* \odot \hat{q}_{13}^*\}\}) \end{pmatrix} = q_{in}^0 + \lambda d_6 \tag{5.20}$$

Para los tres primeros ángulos conocidos, la ecuación (5.20) se convierte en:

$$\left(\begin{array}{c} (V\{R\{\hat{q}_{45} \odot \hat{l}_8^{'} \odot \hat{q}_{45}^*\}\}) \times (V\{D\{\hat{q}_{45} \odot \hat{l}_8^{'} \odot \hat{q}_{45}^*\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_{45} \odot \hat{l}_7^{'} \odot \hat{q}_{45}^*\}\}) \times \\ (V\{D\{\hat{q}_{45} \odot \hat{l}_7^{'} \odot \hat{q}_{45}^*\}\}) \end{pmatrix} \cdot (V\{R\{\hat{q}_{45} \odot \hat{l}_8^{'} \odot \hat{q}_{45}^*\}\}) \right) * \\ *(V\{R\{\hat{q}_{45} \odot \hat{l}_8^{'} \odot \hat{q}_{45}^*\}\}) \end{array} \right) = q_{in}^0 + \lambda d_6 \qquad (5.21)$$

donde $\hat{l}_8^{'} = \hat{q}_{13} \odot \hat{l}_8 \odot \hat{q}_{13}^*$ y $\hat{l}_7^{'} = \hat{q}_{13} \odot \hat{l}_7 \odot \hat{q}_{13}^*$. Se tiene que tener en cuenta que el parámetro d_6 cambia en función de la variable q_{in}, es decir se tiene que aplicar al eje d_6 original una rotación de la variable q_{in}. El resultado será el valor correcto para esta variable.

La ecuación anterior proporciona el sub-problema 2 de "Paden-Kahan". Los parámetros de este sub-problema son:

$$a = \left(\begin{array}{c} (V\{R\{\hat{l}_8^{'}\}\}) \times (V\{D\{\hat{l}_8^{'}\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{l}_7^{'}\}\}) \times \\ (V\{D\{\hat{l}_7^{'}\}\}) \end{pmatrix} \cdot (V\{R\{\hat{l}_8^{'}\}\}) \right) * \\ *(V\{R\{\hat{l}_8^{'}\}\}) \end{array} \right) \qquad (5.22)$$

donde el parámetro l_1 es el eje imaginario 7 que es d_7, el parámetro l_2 es el eje imaginario 8 que es d_8 y el valor $b = q_{in}^0 + \lambda d_6$. Con la ayuda de estos parámetros y utilizando el sub-problema 2 se puede encontrar la solución a las variables de articulación θ_4 y θ_5.

5.3.2.4 Cálculo de θ_6

Ahora solo queda una variable de articulación desconocida que es θ_6. Es fácil ahora resolver esta articulación numéricamente porque se trata solamente de un único parámetro desconocido, sin embargo se va a resolver la última variable de articulación utilizando los sub-problemas de "Paden-Kahan". Para encontrar el último parámetro, se necesita un punto que no se encuentre sobre el eje de la última articulación. Se define el punto $p_d = p_5 + \lambda d_5$. Se utilizan dos ejes imaginarios para encontrar el punto $p_d^{'}$ que es la posición del punto p_d después de la rotación del ángulo θ_6. El punto p_d es el punto de la intersección de estos dos ejes imaginarios. Por eso se definen dos ejes imaginarios que estén sobre el eje de articulación 5 y que intersecten en el punto p_d:

$$\begin{array}{rr} d_9 = [0,1,0] & d_{10} = [1,0,0] \\ p_9 = p_d & p_{10} = p_d \\ m_9 = p_9 \times d_9 & m_{10} = p_1 0 \times d_{10} \\ \hat{l}_9 = d_9 + \varepsilon m_9 & \hat{l}_{10} = d_{10} + \varepsilon m_{10} \end{array} \tag{5.23}$$

La posición del punto $p_d^{'}$ puede ser encontrado por:

$$\left(\begin{array}{c} (V\{R\{\hat{q}_{16} \odot \hat{l}_{10} \odot \hat{q}_{16}^*\}\}) \times (V\{D\{\hat{q}_{16} \odot \hat{l}_{10} \odot \hat{q}_{16}^*\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_{16} \odot \hat{l}_9 \odot \hat{q}_{16}^*\}\}) \times \\ (V\{D\{\hat{q}_{16} \odot \hat{l}_9 \odot \hat{q}_{16}^*\}\}) \end{pmatrix} \cdot (V\{R\{\hat{q}_{16} \odot \hat{l}_{10} \odot \hat{q}_{16}^*\}\}) \right) * \\ *(V\{R\{\hat{q}_{16} \odot \hat{l}_{10} \odot \hat{q}_{16}^*\}\}) \end{array} \right) = q_{in}^0 + \lambda d_5 \tag{5.24}$$

Que para los cinco ángulos conocidos, la ecuación (5.24) se convierte en:

$$\left(\begin{array}{c} (V\{R\{\hat{q}_6 \odot \hat{l}_{10}^{'} \odot \hat{q}_6^*\}\}) \times (V\{D\{\hat{q}_6 \odot \hat{l}_{10}^{'} \odot \hat{q}_6^*\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_6 \odot \hat{l}_9^{'} \odot \hat{q}_6^*\}\}) \times \\ (V\{D\{\hat{q}_6 \odot \hat{l}_9^{'} \odot \hat{q}_6^*\}\}) \end{pmatrix} \cdot (V\{R\{\hat{q}_6 \odot \hat{l}_{10}^{'} \odot \hat{q}_6^*\}\}) \right) * \\ *(V\{R\{\hat{q}_{16} \odot \hat{l}_{10}^{'} \odot \hat{q}_{16}^*\}\}) \end{array} \right) = q_{in}^0 + \lambda d_5 \tag{5.25}$$

donde $\hat{l}_{10}^{'} = \hat{q}_{15} \odot \hat{l}_{10} \odot \hat{q}_{15}^*$ y $\hat{l}_9^{'} = \hat{q}_{15} \odot \hat{l}_9 \odot \hat{q}_{15}^*$. Se tiene que tener en cuenta que el parámetro d_5 cambia en función de la variable q_{in}, es decir, se tiene que aplicar al eje d_5 original una rotación de la variable q_{in}. El resultado será el valor correcto para esta variable.

La ecuación (5.25) nos proporciona el sub-problema 1. Los parámetros para este sub-problema son:

$$a = \left(\begin{array}{c} (V\{R\{\hat{l}_{10}^{'}\}\}) \times (V\{D\{\hat{l}_{10}^{'}\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{l}_9^{'}\}\}) \times \\ (V\{D\{\hat{l}_9^{'}\}\}) \end{pmatrix} \cdot (V\{R\{\hat{l}_{10}^{'}\}\}) \right) * \\ *(V\{R\{\hat{l}_{10}^{'}\}\}) \end{array} \right) \tag{5.26}$$

donde el parámetro l es el eje imaginario 6 que es d_6. Con la ayuda de estos parámetros y utilizando el sub-problema 1 se puede encontrar la solución a la variable de articulación θ_6.

Con lo cual ahora ya se dispone de todas las variables de articulación calculadas.

5.3.3 Obtención de todas las posibles configuraciones de una cadena cinemática

Lo que se pretende, además de dar solución tanto a la cinemática directa como la inversa, se pretende poder dar todas las posibles configuraciones para la cinemática inversa, dada una posición y orientación.

Esta es la razón por la que se ha optado por una programación en funciones de cada una de las variables de articulación, para no tener que duplicar código. Se sabe que se puede llegar a tener 8 configuraciones diferentes. Estas 8 configuraciones vienen determinadas por las variables de articulación y los posibles resultados son:

- Codo arriba, con la primera articulación normal, y posición FLIP en la muñeca.
- Codo abajo, con la primera articulación normal, y posición FLIP en la muñeca.
- Codo arriba, con la primera articulación rotada 180 grados, y posición FLIP en la muñeca.
- Codo abajo, con la primera articulación rotada 180 grados, y posición FLIP en la muñeca.
- Codo arriba, con la primera articulación normal, y posición NO FLIP en la muñeca.
- Codo abajo, con la primera articulación normal, y posición NO FLIP en la muñeca.
- Codo arriba, con la primera articulación rotada 180 grados, y posición NO FLIP en la muñeca.
- Codo abajo, con la primera articulación rotada 180 grados, y posición NO FLIP en la muñeca.

Para realizar estas 8 configuraciones los pasos a realizar son:

- Primero se calcula la variable de articulación θ_3, con lo cual realizar codo arriba o codo abajo se trata de calcular esta variable y cambiarle el signo. Con lo que tenemos dos valores diferentes para esta variable.

- Al calcular las variables de θ_1 y θ_2 se tienen dos posibles soluciones, ya que se disponen de dos valores para la variable θ_3. Realmente el valor de θ_1 será el mismo en los dos casos ya que sólo se verá afectada la variable θ_2.

- Con estas variables, además se tiene que rotar la variable θ_1 180 grados y utilizando los mismos valores para θ_2 y θ_3 se obtiene dos nuevas configuraciones.

Estos primeros cálculos nos llevan a 4 configuraciones diferentes, codo arriba o codo abajo, para la variable de articulación 1 rotada o no los 180 grados. Estas configuraciones son las 4 primeras vistas anteriormente.

Ahora se realizan los cálculos para estas mismas configuraciones en posiciones FLIP o NO FLIP.

- Con todas las configuraciones anteriores, ahora se calculan las variables de articulación θ_4 y θ_5. Para ello se utilizan diferentes cuadrantes para calcular estas variables. La manera de cambiar los cuadrantes es a partir del archivo de configuraciones de cada cadena cinemática y esta variable es utilizada como argumento de entrada para cada una de las funciones para los cálculos de θ. Con lo cual con esta variable se puede rotar la variable θ_4 180 grados y cambiar de signo la variable θ_5. Con lo que, se tiene dos soluciones para estas variables.

- Ahora, con todas las configuraciones anteriores, se calcula la variable de articulación θ_6 para cada una de las configuraciones posibles.

Con lo que ahora se tienen las ocho posibles soluciones.

A todos estos cálculos se tiene que añadir que no siempre se puede tener las ocho configuraciones. Esto se debe a que se puede tener configuraciones de la cadena cinemática que no permitan alguna de estas configuraciones como puede ser el caso del robot IRB 140. Por tener un desplazamiento entre las variables de articulación 1 y 2, se impide poder rotar la variable de articulación 1 los 180 grados, ya que las variables de articulación 2 y 3 cambiarían de valores. Con esto, para este caso se tiene solo 4 posibles configuraciones.

También se puede tener cadenas cinemáticas en que los ejes de articulación 4 y 5 sean paralelos, con lo que impide poder realizar las configuraciones FLIP y NO FLIP, pudiendo calcular solamente una de ellas.

Otro caso posible es tener un desplazamiento entre los ejes de articulación 3 y 4, con lo que impide realizar un cambio de signo en la variable de articulación de θ_3 para calcular la configuración de codo arriba o codo abajo.

A continuación se observan unas imágenes donde se puede ver las ocho combinaciones posibles y entender mejor las configuraciones que no se pueden realizar si se tienen los casos dados, Figura 5.11.

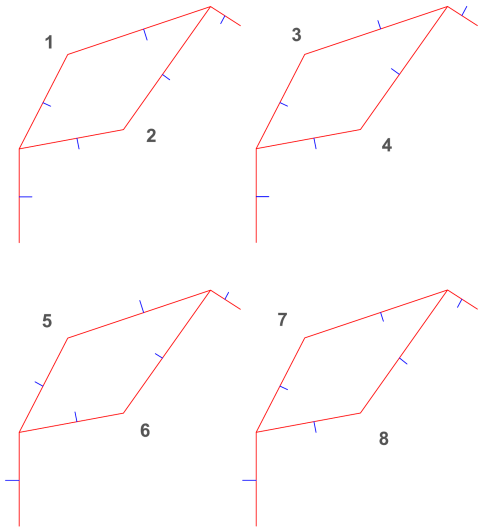


Figura 5.11: Configuraciones posibles de una cadena cinemática.

Capítulo 6

Experimentos y Resultados

Es este capitulo se describe el trabajo de desarrollo y experimentación implementado realizado, analizando los resultados obtenidos.

6.1 Introducción

En este capítulo se van a analizar todos los resultados obtenidos en los experimentos realizados.

Para cada uno de los experimentos realizados, los cuales se visualizan posteriormente, se ha entrenado un modelo distinto para cada uno de estos experimentos. Para cada uno de los experimentas realizados se utiliza una base de datos distinta, y para cada uno de los modelos se utiliza un conjunto de imágenes aleatorio. Las imágenes de cada base de datos utilizados para entrenar un modelo, son diferentes a las imágenes utilizadas para testear el modelo en esa misma base de datos. De esta forma nos aseguramos que las imágenes de entrenamiento y testeo no son los mismos, y ademas tras entrenar diferentes modelos en una misma base de datos, nos aseguramos también que las imágenes no sean las mismas.

La sección 6.2 describe la herramienta de programación utilizada. A continuación, en la sección 6.3 se observan los métodos de evaluación que se utilizan para comparar los resultados. En la sección 6.4 se detallan las bases de datos utilizadas para para comprobar el correcto funcionamiento del modelo 4D-DPM y poder comparar este con otros modelos similares. En la sección 6.5 se visualizan los resultados a los experimentos según tablas de comparación, de esta forma se pueden

comparar los resultados con otros métodos. Tras observar los resultados y obtener todos los datos necesarios, se pueden visualizar los resultados en la sección 6.5.7 solapando los resultados con las imágenes. Además, en esta última sección, en el último apartado se realiza un análisis computacional del sistema.

6.2 Herramienta de desarrollo utilizada

Los experimentos visualizados en las secciones siguientes, han sido implementadas en la herramienta "MATLAB" utilizando "Windows" como sistema operativo, sabiendo que no es una de las mejores combinaciones para realizar simulaciones en tiempo real.

No obstante, todos los algoritmos expuestos en la tesis han sido desarrollados para poder obtener los resultados deseados.

El modelo 4D-DPM propuesto a sido desarrollado a partir del modelo original DPM, desarrollando la ampliación del código para la utilización de las imágenes RGB y de profundidad a la vez.

Ambos filtros, el filtro de Kalman y el filtro de partículas, han sido implementados en esta herramienta para poder comparar ambos filtros y realizar la elección del filtro más adecuado.

Para realizar los cálculos de la cinemática, se han implementado los algoritmos de Denavit-Hartenberg (DH) y cuaterniones duales (DQ). Ambos algoritmos han sido implementados de forma genérica en forma de "toolbox", donde cambiando un solo fichero de configuración, son capaces de realizar los cálculos oportunos para cualquier cadena cinemática. Ambos algoritmos han sido utilizados para corroborar la elección del algoritmo utilizado, en este caso, los cuaterniones duales.

Además de los algoritmos citados anteriormente, se destaca la implementación de los algoritmos de detección de colisiones, la visualización de los resultados utilizando poli-esferas y las proyecciones 2D de estas sobre el plano 2D entre otros.

6.3 Modo de evaluación

Se describen los criterios de evaluación utilizados para evaluar la estimación de la postura del esqueleto del cuerpo humano.

PCP: Ferrari, Marin-Jimenez y Zisserman 2008 describen un protocolo de evaluación ampliamente adoptado basado en la probabilidad de una postura correcta (PCP), que mide el porcentaje de partes del cuerpo correctamente localizadas. Un candidato de la parte del esqueleto del cuerpo humano se etiquetará como correcto si los extremos de dicho segmento están dentro del 50 % de la longitud de los puntos extremos anotados en el la base de verdad (ground-truth). Hay tres dificultades asociadas con su uso en la práctica. En primer lugar, la base de datos BUFFY Ramanan 2007 publicado con Ferrari, Marin-Jimenez y Zisserman 2008 utiliza una definición relajada que marca el promedio de los puntos extremos de la extremidad prevista, y no los extremos de la extremidad en sí. En segundo lugar, el PCP es sensible a la cantidad de "escorzo" de una extremidad, por lo que puede ser una medida demasiado débil en algunos casos y una medida demasiado estricta en otros. Por último, el PCP exige que las posturas candidatas y las posturas del "ground truth" sean colocadas en correspondencia, pero no especifica cómo obtener esta correspondencia. Las soluciones comunes incluyen la evaluación del candidato de puntuación más alto dada: (a) una imagen con una sola persona anotada, (b) una ventana devuelta por un detector de persona. La opción (a) no es satisfactoria porque el candidato puede disparar contra una persona no anotada en el fondo, mientras que la opción (b) no es satisfactoria porque esto predispone los datos de la prueba a ser respuestas de un detector de personas, según Tran y Forsyth 2010. La base de datos BUFFY Ramanan 2007 en su lugar coincide con múltiples candidatos de las posturas del "ground-truth". Las posturas que no coinciden con las del "ground-truth" (detecciones perdidas / falsos negativos) son penalizadas como localizaciones incorrectas, pero notablemente, los candidatos no coincidentes (falsos positivos) no son penalizados. Esto da una ventaja injusta a las aproximaciones que predicen un gran número de candidatos.

PCK: Se proponen dos medidas para la estimación de la posición que abordan estas cuestiones. La primera evaluación explica explícitamente la detección al requerir que las imágenes de testeo sean anotadas con un cuadro delimitador bien acotado para cada persona. Fundamentalmente, no se limita a evaluar un subconjunto de cuadros delimitadores verificados encontrados por un detector, debido a que esto predispone las ventanas de prueba a ser posturas rígidas, como se observa en Tran y Forsyth 2010. El enfoque es similar al protocolo utilizado en el "PASCAL Person Layout Challenge" Everingham y col. 2010. Dado el cuadro delimitador, un algoritmo de la estimación de la postura debe divulgar las loca-

lizaciones de los puntos de interés de las variables de articulación del esqueleto del cuerpo humano. El "Person Layout Challenge" mide la superposición entre los cuadros delimitadores clave (keypoint bounding boxes), que pueden sufrir de artefactos de cuantificación para cuadros delimitadores pequeños. Definimos un punto clave (keypoint) candidato para que sea correcto si cae dentro de $\alpha \cdot \text{máx}(h, \omega)$ píxeles del punto clave del "ground-truth", donde h y ω son la altura y el ancho del cuadro delimitador respectivamente, y α controla el umbral relativo para la corrección considerada. Es decir, el punto correspondiente del "ground-truth" se enmarca dentro de un cuadro delimitador, y si la solución dada cae dentro de este cuadro, el punto es correcto, si cae fuera, es falso.

APK: En un sistema real, sin embargo, uno no tendrá acceso a cuadros delimitadores anotados en el tiempo de testeo, y por lo tanto también se debe abordar el problema de detección. Se puede combinar de forma limpia los dos problemas pensando en las partes del cuerpo (o más bien las articulaciones) como objetos a ser detectados, y evaluar la precisión de la detección de objetos con una curva de recuperación de precisión Everingham y col. 2010. Como arriba, se considera que un candidato es correcto (verdadero positivo) si se encuentra dentro de $\alpha \cdot \text{máx}(h, \omega)$ del "ground-truth". Es decir, se mide la distancia del punto correspondiente del "ground-truth" a la solución dada, si la distancia es menor a un umbral, el punto es correcto, si el umbral es mayor, es falso. A esto se le llama la precisión media de los puntos clave (APK). Esta evaluación penaliza correctamente tanto las detecciones perdidas como las falsas. Obsérvese que la correspondencia entre los candidatos y las posturas del "ground-truth" se establecen separadamente para cada punto clave, por lo que esto sólo proporciona una visión "marginal" de la precisión de detección del punto clave. Pero estas estadísticas marginales son útiles para comprender qué partes son más difíciles que otras. Finalmente, APK requiere que todas las personas sean etiquetadas en una imagen de prueba, a diferencia de PCP y PCK.

PCP vs PCK vs APK. Debido a que APK es la evaluación más realista y más estricta, se considera el "estándar de oro". Al ajustar la estrategia de supresión no máxima (NMS) para que el detector devuelva más posturas candidatas, se realiza peor en APK pero artificialmente se mejora en PCP. Este comportamiento tiene sentido dado que los falsos positivos no son penalizados por el PCP, pero penalizados por el APK. Se puede realizar una curva similar que compara APK y PCK bajo diferentes estrategias de NMS, pero hay que tener en cuenta que PCK no se ve afectada por NMS porque se dan ventanas de verdad. Más bien, se selecciona una dimensión arbitraria del modelo para evaluar (como el número de mezclas), y mostrar una correlación positiva de PCK con APK. Dado que PCK

es más fácil de interpretar y más rápido de evaluar que APK, se utiliza PCK para realizar experimentos de diagnóstico que exploran diferentes aspectos del modelo.

Error: el error utilizado en las tablas se centra únicamente en el la media del error de cada uno de los puntos. Calculando la distancia entre los puntos etiquetados en la base de verdad y los puntos, sumando las diferencia obtenidas en cada instante de tiempo, y obtener la media para cada uno de los puntos en el instante final.

6.4 Datasets utilizados

Para realizar la evaluación del modelo 4D-DPM se han utilizado diferentes bases de datos. Estas bases de datos son los siguientes.

6.4.1 Base de datos "PARSE"

Esta base de datos es pública y se puede encontrar en el siguiente enlace: "http:// www.ics.uci.edu/ dramanan/papers/parse/"

6.4.2 Base de datos "CAD60"

Esta base de datos también es pública y se puede encontrar en el siguiente enlace: "http://pr.cs.cornell.edu/humanactivities/data.php".

Las características de esta base de datos son: contiene 60 videos con imágenes RGB y de profundidad donde intervienen 4 personas. Están grabados en 5 ambientes diferentes: oficina, cocina, dormitorio, baño y en el comedor. Las personas dentro de la imagen recogen un total de 12 actividades diferentes. En cada una de las imágenes interviene solo 1 persona. Además se proporciona para cada imagen las etiquetas de cada uno de los puntos de interés, entre ellos las variables de articulación del esqueleto del cuerpo humano.

6.4.3 Base de datos "CAD60 AMPLIADA"

Esta base de datos se ha elaborado a partir de la base de datos anterior. Se ha ampliado la base de datos "CAD60" con más videos de una nueva persona en 2 ambientes nuevos de oficinas. Esta persona realiza ciertos movimientos deseados para poder comparar casos más concretos en las soluciones de los modelos, como por ejemplo cruzar los brazos por delante del cuerpo.

Para la realización de estos videos se ha utilizado una cámara de bajo coste, como es el caso de la cámara Kinect. Para la utilización de esta cámara es necesario realizar la calibración de los parámetros intrínsecos y extrínsecos del sensor monocular y del sensor IR de la cámara Kinect.

6.5 Resultados

Una vez vistos los modos de evaluación y las bases de datos utilizadas, a continuación se presentan las tablas obtenidas tras realizar los experimentos y en donde se pueden observar los resultados.

6.5.1 Utilización de MSER

La siguiente table, tabla 6.1, muestra los resultados obtenidos para decidir el correcto funcionamiento de la eliminación del fondo utilizando MSER. Para ello se a entrenado el modelo 4D-DPM utilizando un conjunto de imágenes de la base de datos "CAD60 ampliada" para validar la utilización del MSER.

Tabla 6.1: Métricas APK, PCK y Error para la eliminación del fondo en las imágenes. Las métricas APK y PCK están expresadas en %, mientras que el error está expresado en píxeles.

Modelo	MSER entrenando	MSER testeando	Evaluación	Cabeza	Hombros	Muñeca	Caderas	Tobillo	Media
1	no	no	APK	100	100	89,64	100	100	97,92
			PCK	100	100	92,42	100	100	98,48
			error	4,22	3,66	7,63	5,96	4,43	5,18
2	no	si	APK	100	100	83,79	100	100	96,75
			PCK	100	100	89,39	100	100	97,87
			error	4,54	**3.61**	7,70	**3.35**	3,77	4,59
3	si	no	APK	100	100	82,40	100	100	96,49
			PCK	100	100	87,37	100	100	97,47
			error	3,03	4,49	9,65	3,38	3,09	4,72
4	**si**	**si**	**APK**	100	100	**95.63**	100	100	**99.12**
			PCK	100	100	**96.46**	100	100	**99.29**
			error	**2.55**	4,70	**5.62**	3,41	**2.64**	**3.78**

"MSER entrenando" representa la utilización del método MSER para el entrenamiento del modelo 4D-DPM y "MSER testeando" representa la utilización del método MSER de manera on-line.

Incluso utilizando un pequeño subconjunto de imágenes, e incluso obteniendo en algunos casos una precisión del 100 %, en los casos que la precisión es menor, se ve como esta aumenta según se utiliza el método MSER, llegando a la conclusión que la utilización de este en ambos casos, para entrenar el modelo como de manera on-line, se obtienen mejores resultados.

6.5.2 DPM vs 4D-DPM sin MSER y sin el filtro de partículas

El modelo original DPM esta entrenado con la base de datos "PARSE", que solo contiene imágenes en RGB, de tal forma que no se puede comparar dicho modelo con el propuesto 4D-DPM. Para ello lo que hacemos es re-entrenar el modelo DPM original con la base de datos de "CAD60", y entrenar el modelo propuesto 4D-DPM. Para ello se utilizan las mismas imágenes para entrenar ambos modelos y se utilizan las mismas imágenes para testear los modelos obtenidos, pero las imágenes de entrenamiento y las imágenes de testeo no son las mismas. La tabla resultante es la tabla 6.2. Para una correcta comparación de ambos métodos en el modelo 4D-DPM no se ha utilizado el filtro de partículas descrito ni el método MSER para la eliminación del fondo. La tabla visualiza las 10 partes utilizadas en el modelo 4D-DPM propuesto, comparándolas con las mismas 10 partes del modelo original DPM, aunque el modelo original contenga 14 partes. Estas 10 partes son: la cabeza, los hombros, las manos, las caderas y los pies. No se comparan los codos y las rodillas porque serán comparados posteriormente cuanto se utilicen los cuaterniones duales para inferir en el número de puntos utilizados. A estos puntos se le añade la posición del tronco, pero este punto no se utiliza en las tablas.

Tabla 6.2: Métricas APK, PCK y Error utilizando la base de datos "CAD60". Las métricas APK y PCK están expresadas en %, mientras que el error está expresado en píxeles.

Modelo	Evaluación	Cabeza	Hombros	Muñeca	Caderas	Tobillos	Media
DPM	APK	47,42	66,69	22,95	45,98	47,10	46,02
	PCK	62,00	70,50	39,00	60,00	57,50	57,80
	error	17,35	14,10	35,89	7,06	19,57	18,79
DPM-t	APK	73,02	73,53	32,26	66,33	42,38	57,50
	PCK	78,50	78,50	44,50	70,50	49,50	64,30
	error	15,21	12,30	31,02	6,64	16,31	16,29
4D-DPM	**APK**	91,23	87,06	51,63	86,21	82,01	79,63
	PCK	92,80	90,00	66,00	89,00	90,00	85,56
	error	8,81	7,53	19,25	6,05	9,25	10,17

Donde "DPM" representa el modelo DPM original entrenado en la base de datos "PARSE" pero testeado dicho modelo con la base de datos "CAD60". "DPM-t" representa el modelo original DPM entrenado y testeado en la base de datos "CAD60". "4D-DPM" representa el modelo propuesto entrenado y testeado en la base de datos "CAD60". Tras observar la tabla 6.2 se observa que el modelo 4D-DPM mejora en la precisión al modelo DPM original alrededor de un 20 % y se reduce en 10 puntos el error.

6.5.3 DPM vs 4D-DPM sin y con el filtro de Kalman

La primera idea utilizada para mejorar los resultados del modelo 4D-DPM fue introducir el filtro de Kalman para comprobar que los resultados mejoraban y poder estudiar posteriormente otras opciones para realizar el seguimiento de los puntos de interés dentro de una imagen. La tabla 6.3 muestra los resultados obtenidos.

Tabla 6.3: Comparación entre el modelo original DPM entrenado y testeado en la base de datos "CAD60" y el modelo 4D-DPM con y sin el filtro de Kalman. Las métricas APK y PCK están expresadas en %, mientras que el error está expresado en píxeles.

Modelo	Evaluación	Cabeza	Hombros	Muñeca	Caderas	Tobillos	Media
DPM-t	APK	91,20	92,30	82,70	86,60	83,50	87,26
	PCK	91,50	89,00	85,80	89,90	83,80	88,00
	Error	8,17	8,81	10,87	9,37	11,59	9,76
4D-DPM sin el KF	APK	94,20	95,10	88,30	89,70	90,30	91,52
	PCK	93,80	92,50	88,90	90,30	91,00	91,30
	Error	6,48	6,02	8,73	8,01	7,66	7,38
4D-DPM con el KF	APK	97,50	98,30	92,20	94,70	94,00	95,34
	PCK	96,40	95,20	93,70	96,50	94,20	95,20
	Error	5,82	5,71	7,43	6,37	6,61	6,38

donde "DPM-t" representa el modelo original DPM entrenado y testeado en la base de datos "CAD60". "4D-DPM" representa el modelo propuesto entrenado y testeado en la base de datos "CAD60" además de las dos soluciones aportadas, sin utilizar y utilizando el filtro de Kalman. Para obtener estos resultados en ambos modelos se han utilizado las mismas imágenes para realizar el entrenamiento y testeo del modelo, pero las imágenes de entrenamiento y testeo son diferentes entre si. Observando la tabla 6.3 volvemos a llegar a la misma conclusión que antes, que con el modelo 4D-DPM se obtiene una precisión mas alta utilizando y sin utilizar el filtro de Kalman, además que la precisión del modelo 4D-DPM utilizando el filtro de Kalman proporciona un 3,5 % más de precisión comparado con el mismo modelo sin utilizar el filtro de Kalman.

6.5.4 4D-DPM sin y con el filtro de partículas

Tras observar la mejora al introducir el filtro de Kalman, se propone introducir un filtro diferente, en este caso el filtro de partículas, ya que este tipo de filtros obtienen mejores respuestas en sistemas no lineales como es nuestro caso. A continuación, tabla 6.4, se observan los resultados del modelo 4D-DPM sin y con la utilización del filtro de partículas.

Tabla 6.4: Comparación entre el modelo 4D-DPM entrenado y testeado en la base de datos "CAD60 ampliada" con y sin la utilización del filtro de partículas. Las métricas APK y PCK están expresadas en %, mientras que el error está expresado en píxeles.

Modelo	Evaluación	Cabeza	Hombros	Muñeca	Caderas	Tobillos	Media
4D-DPM sin el PF	APK	92,60	93,20	87,10	88,20	88,50	89,92
	PCK	93,10	91,70	86,70	89,40	90,40	90,26
	Error	6,95	7,15	9,58	8,43	7,89	8,00
4D-DPM con el PF	APK	94,10	95,40	90,40	91,20	91,60	92,54
	PCK	96,40	92,90	91,10	92,00	92,10	92,10
	Error	6,33	6,97	7,48	7,25	7,35	7,07

Para obtener estos resultados en ambos modelos se han utilizado las mismas imágenes para realizar el entrenamiento y testeo del modelo, pero las imágenes de entrenamiento y testeo son diferentes entre si. Observando la tabla 6.4 se observa que la utilización del filtro de partículas mejora la precisión de los resultados alrededor de un 2,5 %. Con lo cual ahora sabemos que el filtro de partículas también mejora los resultados obtenidos.

6.5.5 4D-DPM con el filtro de Kalman vs 4D-DPM con el filtro de partículas

Tras comprobar que ambos filtros, filtro de Kalman y el filtro de partículas, mejoran los resultados del modelo DPM original, haciendo que el modelo 4D-DPM obtenga mejor precisión en los resultados, se realiza un estudio para comprobar qué filtro realiza una mejora más sustancial en los resultados. Para ello se visualiza la siguiente tabla, tabla 6.5:

Tabla 6.5: Comparación entre el modelo 4D-DPM entrenado y testeado en la base de datos "CAD60 ampliada" utilizando el filtro de Kalman en un caso y el filtro de partículas en el otro.

Modelo	Evaluación	Cabeza	Hombros	Muñeca	Caderas	Tobillos	Media
4D-DPM con el KF	APK	89,60	88,70	84,20	87,50	85,90	87,18
	PCK	90,10	88,90	85,20	88,00	87,10	87,86
	Error	8,16	8,67	9,94	9,25	8,97	8,99
4D-DPM con el PF	APK	92,30	91,10	87,60	89,50	88,20	89,74
	PCK	92,80	90,30	88,10	90,70	89,90	90,36
	Error	7,15	8,02	8,36	8,44	8,69	8,13

Observando la tabla 6.5 se puede ver que tras comparar ambos filtros, los resultados obtenidos por el filtro de partículas mejoran la precisión con respecto a los resultados con el filtro de Kalman alrededor de un 3 %. El filtro de partículas introducido será el filtro que se utilice en el modelo 4D-DPM.

6.5.6 Comparación del modelo 4D-DPM, utilizando MSER, el filtro de partículas y la visualización utilizando los cuaterniones duales, con otros modelos de seguimiento

A continuación se compara el modelo 4D-DPM, realizando la sustracción del fondo utilizando MSER y utilizando el filtro de partículas para aumentar la precisión del sensor, con el modelo original DPM, Felzenszwalb y col. 2010; Felzenszwalb, McAllester y Ramanan 2008, donde solo se utilizan imágenes RGB y con el algoritmo de Shotton y col. 2013c; Shotton y col. 2013a (Kinect), donde solo se utilizan imágenes de profundidad.

Tabla 6.6: Comparación entre el modelo 4D-DPM, modelo DPM original y el algoritmo "Kinect".

Modelo	Evaluación	Cabeza	Hombros	Muñeca	Caderas	Tobillos	Media
DPM-t Felzenszwalb y col. 2010	APK	47,30	66,70	22,40	45,50	47,10	45,80
	PCK	62,50	70,40	39,00	60,50	57,90	58,06
	Error	15,53	12,23	22,34	16,29	18,50	16,97
Kinect Shotton y col. 2013c	APK	68,30	90,70	76,40	9,50	77,10	64,40
	PCK	79,50	94,40	85,00	23,50	85,90	73,66
	Error	13,17	6,85	9,64	18,42	11,28	11,87
4D-DPM	APK	75,40	93,00	83,70	85,50	84,20	84,36
	PCK	84,10	96,30	90,20	88,90	89,90	89,88
	Error	10,59	5,98	8,13	9,82	9,08	8,72

donde "DPM-t" es el modelo original DPM entrenado y testeado con la base de datos "CAD60 ampliada" utilizando solo las imágenes RGB. "Kinect" utiliza el algoritmo de Shotton y col. 2013c donde también se utiliza la cámara Kinect pero solo utilizando las imágenes de profundidad. "4D-DPM" es el modelo propuesto utilizando las imágenes RGBD para entrenar y testear el modelo utilizando el la base de datos "CAD60 ampliada".

En el algoritmo de Kinect, por ser un código no publico, no hay posibilidad de entrenar con la misma base de datos que en los otros casos, pero lo que si que se ha realizado es testear este modelo con las mismas imágenes que se han utilizado en los otros. El algoritmo utilizado por "Microsoft" es un algoritmo en tiempo

real que utiliza solo imágenes de profundidad y donde se ha utilizado una base de datos de imágenes mucho mayor al que se ha utilizado, pero aún así se va a comparar este algoritmo con el modelo 4D-DPM propuesto.

Observando la tabla 6.6 se observa que los resultados son positivos. Comparando los 3 modelos se llega a la conclusión de que se obtiene una mejor precisión en el modelo 4D-DPM propuesto, obteniendo alrededor de un 5 % más de precisión. Es verdad que en el caso de las caderas, en la Kinect, se obtienen unos valores inusuales y esto puede ser debido a que el etiquetado de las caderas en la base de datos utilizada para entrenar este modelo, las caderas no están etiquetadas en la misma posición que en la base de datos "CAD60" o la base de datos "CAD60 ampliada". Para el resto de los casos, se puede afirmar que la precisión del modelo 4D-DPM utilizando la sustracción del fondo utilizando MSER, el filtro de partículas y los cuaterniones duales, se obtienen mejores resultados que en el modelo original DPM, Felzenszwalb y col. 2010, o el modelo de Shotton y col. 2013c.

6.5.7 Visualización de los resultados

Una vez vistos los experimentos realizados, se visualizan las imágenes obtenidas en diferentes casos de experimentos.

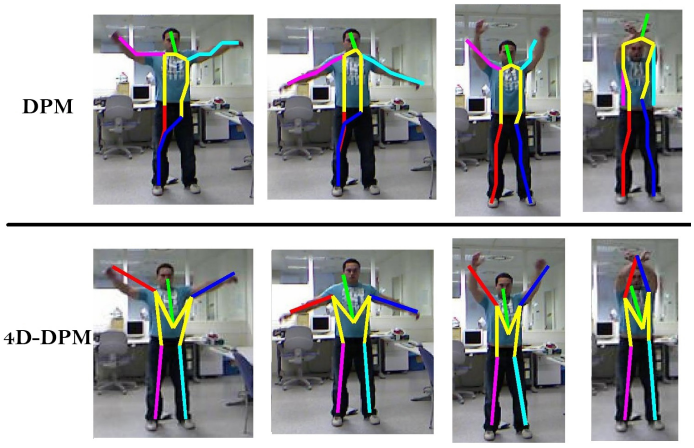


Figura 6.1: Comparación cualitativa entre el modelo original DPM, entrenado con el dataset PARSE y testeado en el dataset CAD60 ampliado, y el modelo 4D-DPM, entrenado y testeado en el dataset CAD60 ampliado.

La figura 6.1 muestra los resultados obtenidos tras testear el modelo original de DPM, entrenado en la base de datos “PARSE” y testeado en la base de datos de “CAD60 ampliada”, fila DPM en la imagen, con el modelo 4D-DPM propuesto utilizando la sustracción del fondo con MSER entrenado y testeado con la base de datos “CAD60 ampliada”.

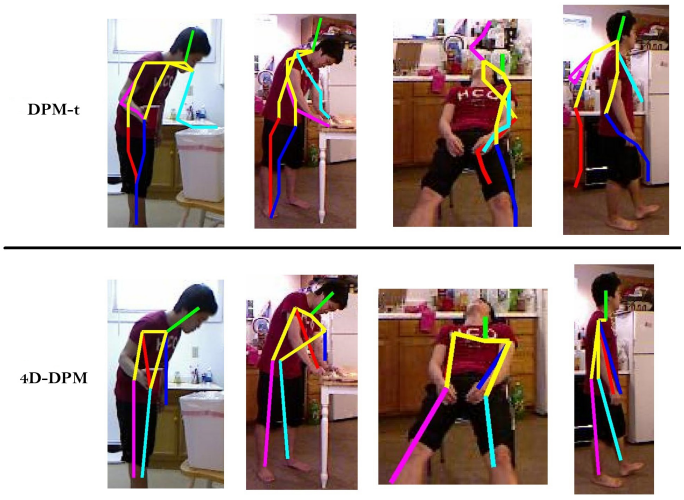


Figura 6.2: Comparación cualitativa entre el modelo original DPM y el modelo 4D-DPM propuesto, entrenados y testeados en la base de datos “CAD60 ampliada”.

La figura 6.2 muestra los resultados obtenidos tras entrenar y testear el modelo original de DPM, fila DPM en la imagen, con el modelo 4D-DPM propuesto utilizando la sustracción del fondo con MSER y el filtro de partículas, sobre la base de datos “CAD60 ampliada”.

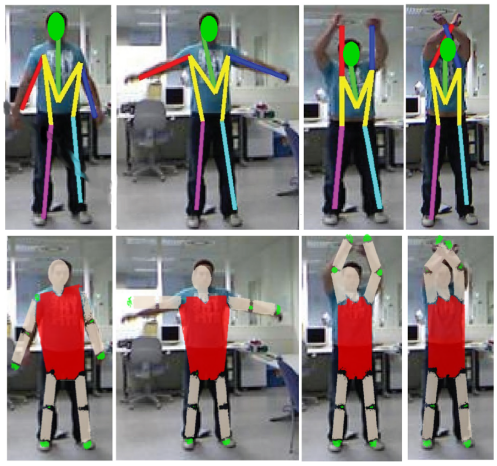


Figura 6.3: Modelo 4D-DPM, entrenado y testeado en la base de datos "CAD60 ampliada". La primera fila muestra los resultados del modelo reducido con 10 partes. La segunda fila muestra el modelo inferido para estimar los codos y las rodillas utilizando la cinemática inversa.

La figura 6.3 muestra los resultados obtenidos tras entrenar el modelo 4D-DPM propuesto sobre la base de datos "CAD60" utilizando la sustracción del fondo con MSER, el filtro de partículas y aplicando los cuaterniones duales para obtener y visualizar el modelo completo. En la fila superior se observa los puntos obtenidos por el modelo 4D-DPM, y en la fila inferior se muestran todos los puntos utilizados para la visualización utilizando poli-esferas solapando esta representación con la imagen RGB.

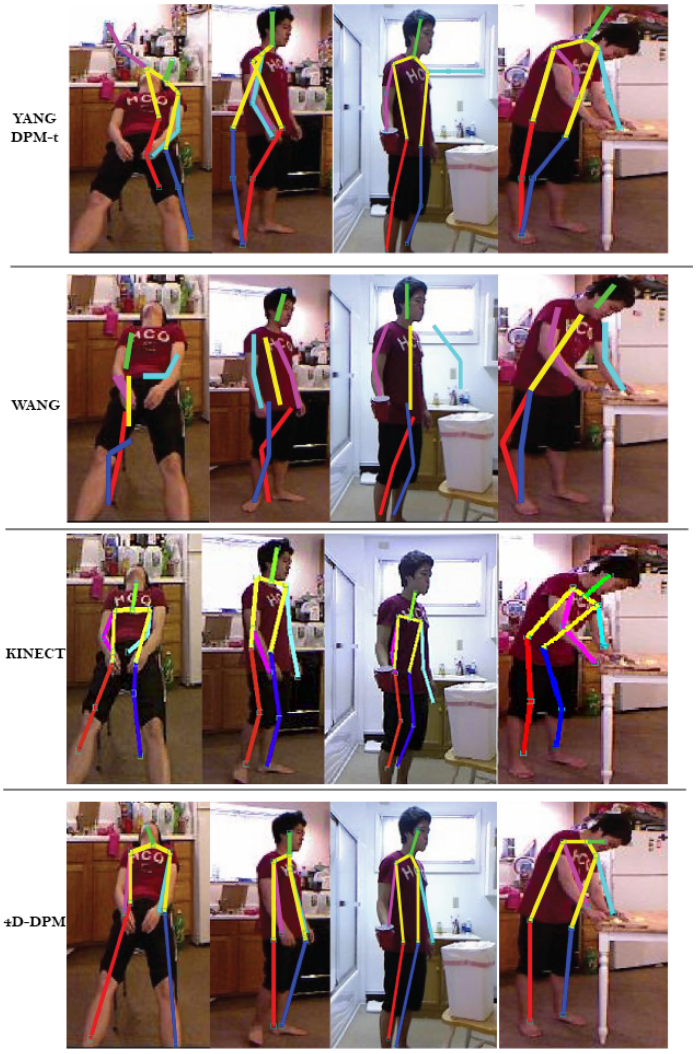


Figura 6.4: Comparación cualitativa entre 4 modelos diferentes para la estimación de la postura en 4 secuencias de la base de datos CAD60.

La figura 6.4 muestra los resultados obtenidos de diferentes modelos donde "DPM-t" es el modelo original DPM, "WANG" es el modelo publicado por Fang y Yi 2013, "Kinect" es el modelo de Shotton y col. 2013c y "4D-DPM" es el modelo propuesto utilizando MSER y el filtro de partículas.

6.5.8 Análisis del coste computacional

Para el análisis del coste computacional, primero se observan las comparativas entre la utilización de DH y DQ, el cual nos lleva al porqué de la utilización de los DQ, y las comparativas entre la utilización del filtro de partículas y el filtro de Kalman. Por último se describe el sistema utilizado para realizar los experimentos y el coste computacional utilizado tanto en el entrenamiento como en el testeo del modelo 4D-DPM propuesto con respecto al modelo DPM original.

6.5.8.1 Comparación: Denavit-Hartenberg (DH) vs cuaterniones duales (DQ)

Ahora, se presenta un estudio comparativo entre los cuaterniones duales y Denavit-Hartenberg. Las soluciones a la cinemática directa e inversa utilizando DH se puede encontrar en Murray, Li y S.S. 1994; Xie y col. 2007. El método de DH utiliza matrices de transformación homogéneas. Estas requieren de 16 posiciones de memoria para la definición del movimiento del cuerpo rígido, mientras que los cuaterniones duales requieren 8 posiciones de memoria. Esto afecta al tiempo de cómputo debido a que el costo de ir buscando un operando en la memoria excede el costo de realizar una operación aritmética básica, Aspragathos y Dimitros 1998, y es muy importante para la aplicación en tiempo real. La comparación de los resultados entre DH y DQ se puede ver en las siguientes tablas, Tabla 6.7 y Tabla 6.8.

Tabla 6.7: Comparación de rendimiento de las operaciones de rotación.

Método	Memoria	Multiplicaciones	Sumas / Restas	Total
Matriz Rotación	9	27	18	45
Cuaternión	4	16	12	28

Tabla 6.8: Comparación de rendimiento de las operaciones de transformación rígida.

Método	Memoria	Multiplicaciones	Sumas / Restas	Total
Matrices homogéneas	16	64	48	112
Cuaterniones duales	8	48	40	88

Con el fin de obtener las operaciones de la cadena de transformación del cuerpo rígido para los robots manipuladores de n grados de libertad:

- Se han de realizar $64(n-1)$ multiplicaciones y $48(n-1)$ sumas si el operador son las matrices de transformación homogénea.
- Se han de realizar $48(n-1)$ multiplicaciones y $40(n-1)$ sumas si el operador de transformación son los cuaterniones duales.

Si se coge que se tiene una cadena cinemática de 6 articulaciones, se necesitan 320 multiplicaciones y 240 sumas para el método de DH, y 240 multiplicaciones y 200 sumas para el método de cuaterniones duales. La siguiente figura, figura 6.5, muestra como al aumentar los grados de libertad, el método de cuaterniones duales, como transformación del cuerpo rígido, tiene más ventajas que el método de DH, Figura 6.5.

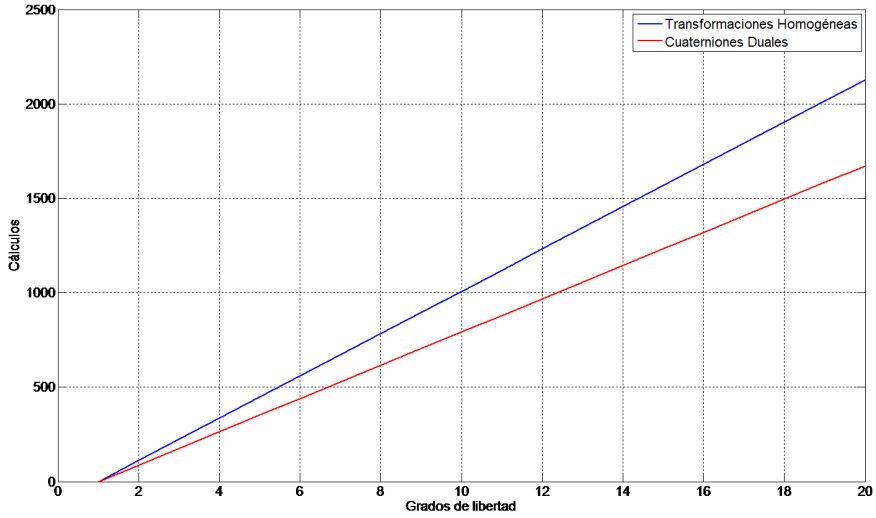


Figura 6.5: Comparación número de cálculos necesarios entre DH y Cuaterniones Duales.

Las figuras, Figura 6.6 y Figura 6.7, comparan el tiempo de simulación para las soluciones de la cinemática directa e inversa utilizando cuaterniones duales o el método de DH.

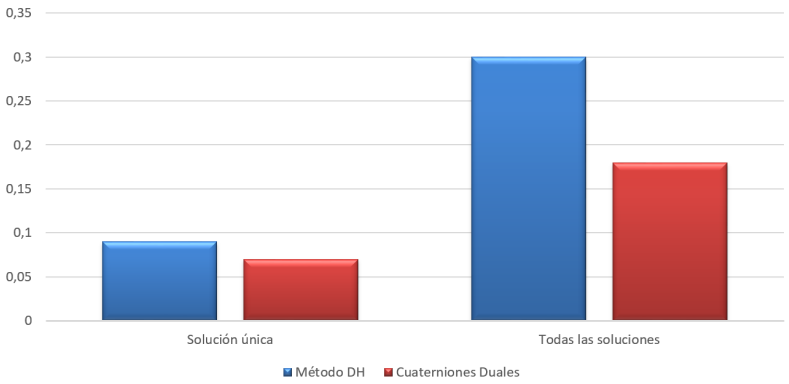


Figura 6.6: Tiempos empleados (en segundos) para obtener la solución de la cinemática directa.

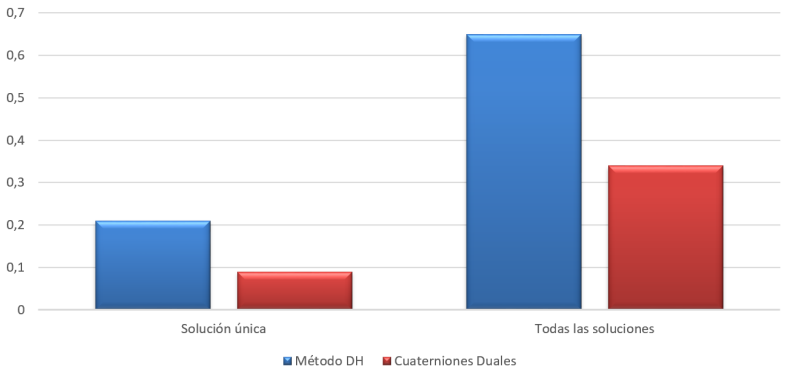


Figura 6.7: Tiempos empleados (en segundos) para obtener la solución de la cinemática inversa.

Como se puede ver en las figuras anteriores, el método que utiliza los cuaterniones duales como operadores del movimiento de "screw" es más eficiente computacionalmente que el método de DH.

Para realizar estos cálculos se ha utilizado el sistema de la tabla 6.9.

Tabla 6.9: Sistema utilizado.

CPU	Memoria	Systema operativo	Software
Intel Quad Core	4 Gb	Windows 7	Matlab 2011

6.5.8.2 Comparación: Filtro de partículas (PF) vs filtro de Kalman (KF)

El filtro de Kalman (KF) y el filtro de partículas (PF) son algoritmos que actualizan recursivamente una estimación del estado y encuentran las innovaciones que conducen un proceso estocástico dada una secuencia de observaciones. El filtro de Kalman cumple este objetivo mediante proyecciones lineales, mientras que el filtro de partículas lo hace mediante un método secuencial de Monte Carlo.

El sistema que se utiliza para realizar el seguimiento de la postura del esqueleto del cuerpo humano es un sistema no lineal. Para sistemas no lineales el filtro de partículas genera mejores resultados en comparación con el filtro de Kalman.

El coste computacional del filtro de partículas, en nuestro caso, es menor al coste computacional del filtro de Kalman. En el filtro de partículas el coste computacional es función del número de partículas utilizadas. En nuestro caso se utilizan 200 partículas asegurándonos que la solución aportada por el filtro de partículas es mejor a la aportada por el filtro de Kalman, al mismo tiempo que el coste computacional del filtro de partículas es menor al del filtro de Kalman.

Los resultados obtenidos tras la comparación de ambos filtros puede observarse en las siguientes secciones.

6.5.8.3 Sistema utilizado y coste computacional

Para la realización de los experimentos anteriores se ha utilizado un sistema con "Windows 7" de 64 bits con 4Gb de memoria RAM. El procesador utilizado es un "Intel Core Quad 2,33 GHz". Para cada imagen se ha calculado el tiempo medio utilizado en el modelo 4D-DPM propuesto para procesar cada imagen. Las imágenes utilizadas tienen una resolución de 320x240 píxeles.

En la parte de entrenamiento, el modelo 4D-DPM propuesto utiliza un total de 8,10 minutos por cada imagen, mientras que el modelo original de DPM, Felzenszwalb y col. 2010, utiliza 8,54 minutos por cada imagen, esto es aproximadamente un 5 % de reducción del coste computacional utilizando el modelo propuesto.

En la parte de testeo del modelo, nuestro modelo utiliza 7,20 segundos para nuestro modelo, mientras que el modelo original de DPM, Felzenszwalb y col. 2010, utiliza 9,21 segundos para cada imagen, esto es reducir el coste computacional alrededor del 20 % utilizando el modelo 4D-DPM propuesto. Estos resultados se han obtenido utilizando la herramienta "MATLAB". Como se ha visto anteriormente, el código utilizado se puede optimizar realizando la programación en "C++" y paralelizando el procesamiento de imágenes en sistemas de varios núcleos o utilizando la GPU.

Sin embargo hay que tener en cuenta que el modelo 4D-DPM propuesto, donde se utiliza la sustracción del fondo con MSER, el filtro de partículas y los cuaterniones duales, es mucho más lento que el modelo Kinect, Shotton y col. 2013c, que es un modelo de tiempo real. Pero según los resultados anteriormente vistos, se puede decir que utilizando nuestro modelo, donde se tiene información tanto de la imagen de RGB como de la de profundidad, y entrenando el modelo con muchas menos imágenes, se puede llegar a obtener mejores resultados.

Capítulo 7

Conclusiones y Trabajos futuros

Es este capitulo se describen las conclusiones a las cuales se ha llegado tras observar los resultados anteriores.

7.1 Conclusiones

El objetivo general de la tesis, tal y como hemos visto en el capítulo 1 pretende desarrollar mejoras para realizar la detección del esqueleto del cuerpo humano dentro de una imagen, desarrollando filtros de seguimiento de los puntos de interés introduciéndoles restricciones gracias al diseño de modelos geométricos. Los modelos geométricos servirán además para la representación de la solución aportada desarrollando el modelo cinemático del cuerpo humano.

Con el modelo expuesto en los capítulos anteriores se ha conseguido mejorar el funcionamiento del modelo DPM, que utiliza solo imágenes monoculares RGB y 14 partes para representar el esqueleto del cuerpo humano, incrementando en una dimensión más el modelo utilizando la imagen de profundidad. El añadir una dimensión más al sistema, influye directamente en el aumento del coste computacional del mismo y para contrarrestar tal efecto, se ha optado por inferir en el número de partes utilizadas en el modelo DPM original, donde se utilizan 14 partes, y utilizar solo 10 partes.

Una variación del número de partes, significa una variación en la precisión del modelo. En artículos previos, como por ejemplo Yang y Ramanan 2013, se reduce el modelo DPM utilizando 26 partes a solo 14 partes sin perjudicar a la precisión del modelo debido a las mejoras del modelo introducidas. Pero si en este mismo modelo DPM variamos el número de partes, variamos también la precisión, aumentando la precisión en los casos donde se aumente el número de partes y reduciéndola en los casos en que se reduce el número de partes. Nuestro caso reduce el número de partes a 10 sobre el modelo original, llegando a pensar que así reducimos la precisión del modelo, pero la influencia de la imagen de profundidad en el modelo 4D-DPM propuesto, resuelve este problema tal y como hemos visto en la sección de resultados, sección 6.

El reducir el número de partes del modelo puede afectar directamente en la precisión del mismo. En artículos previos, como por ejemplo Yang y Ramanan 2013, se reduce el modelo de 26 a 14 partes sin perjudicar a la precisión del modelo, para ello se han introducido mejoras en el modelo DPM y poder realizar esta disminución del número de partes. Hay que tener en cuenta que cuando más partes se utilicen en el modelo, la precisión del mismo será mejor, pero esto también influye en un incremento del coste computacional, una desventaja a tener en cuenta si se quiere reducir el tiempo computacional al máximo sin reducir la precisión del modelo. Del mismo modo, utilizando el mismo modelo original DPM y reducir el número de partes a 10, como es nuestro caso, perjudica a la

Para mejorar la precisión del modelo DPM se ha añadido un filtro de partículas capaz de realizar el seguimiento de los puntos de interés dentro de la imagen a lo largo del tiempo y corregir aquellas soluciones donde se ha detectado que el sensor, la cámara, ha fallado.

Para estimar la postura del esqueleto del cuerpo humano es necesario inferir en las partes de este modelo DPM propuesto para volver a obtener las 14 partes necesarias. Para ello se ha realizado el modelado geométrico del cuerpo humano como si de un conjunto de cadenas cinemáticas se tratara. Utilizando el modelado en poli-esferas para representar cada una de estas cadenas cinemáticas. La utilización de las poli-esferas nos da como ventaja el poder utilizar dicho modelo para la detección de colisiones entre diferentes partes, y añadir esta restricción al filtro de partículas. Además, para cada una de las cadenas cinemáticas en las que se ha dividido el esqueleto del cuerpo humano se han definido unos límites de variables de articulación, el cual también nos sirven como restricción en el filtro de partículas.

Tratar al cuerpo humano como si de un conjunto de cadenas cinemáticas se tratara nos da como ventaja poder utilizar el modelo cinemático para inferir en el número

de partes del modelo. Para ello se ha optado por la utilización de los cuaterniones duales para dar solución a la cinemática inversa que nos ayuda a poder obtener las 14 partes necesarias para representar, utilizando las poli-esferas, la postura del esqueleto del cuerpo humano.

La siguiente figura, figura 7.1, muestra el esquema final obtenido con cada una de las partes utilizadas para llegar a la solución final.

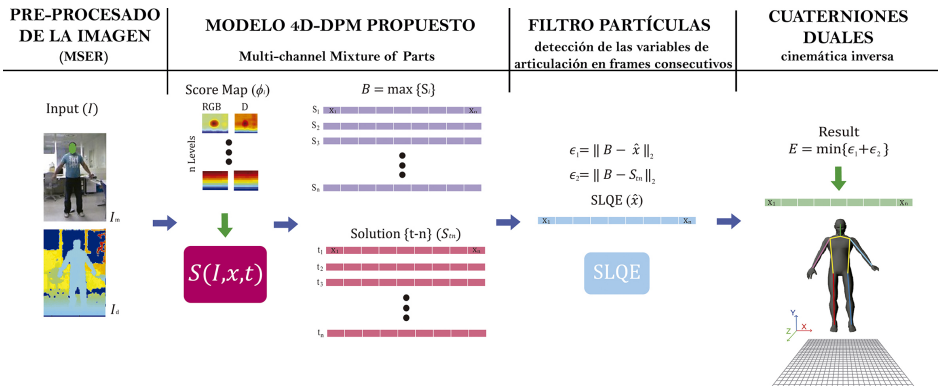


Figura 7.1: Estructura final.

Con lo que como aportaciones de la presente tesis tenemos: la introducción de la imagen de profundidad en el modelo DPM original nombrando el nuevo modelo con el nombre 4D-DPM, la reducción del número de partes a detectar por el algoritmo 4D-DPM, la introducción de las 3 restricciones vistas en el filtro de partículas, la representación del modelo geométrico utilizando poli-esferas y la utilización de los cuaterniones duales para dar solución a la cinemática inversa e inferir en el número de variables de articulación detectadas. Estos aportes han sido publicados en los artículos mencionados a continuación, donde la publicación de alguna de estas partes están en fase de revisión en diferentes revistas.

7.2 Artículos publicados

Al principio de la tesis se colaboró en varios artículos como Ricolfe, Sanchez y Martinez 2011; Ricolfe, Sanchez y Martinez 2012 para la calibración de cámaras con grandes angulares, que nos sirvió de base para calibrar el sensor utilizado, la cámara Kinect, y calibrar los parámetros tanto intrínsecos como extrínsecos de la misma y poder solapar las imágenes de RGB y profundidad para utilizar los datos que ambos nos suministran.

Posteriormente desarrollamos un algoritmo para calcular la cinemática en robots paralelos, Martinez, Benimeli y Sanchez 2012, y que aumento los conocimientos para el cálculo de la cinemática utilizando varios métodos de cálculo, que como hemos visto son Denavit-Hartenberg, cuaterniones duales, métodos geométricos, entre otros.

Una vez capaces de realizar el control de cadenas cinemáticas, utilizamos esta información para desarrollar Martinez, Benimeli y Sanchez 2012 donde se realiza la interacción hombre-máquina a través del seguimiento del efector final de una cadena cinemática con la ayuda del filtro de Kalman. Tras la calibración de la cámara y con la ayuda del seguimiento realizado, podemos conocer la configuración del robot en cada instante y utilizar esta información para evitar una posible colisión entre el hombre y la máquina. En Martinez y Sanchez 2011 se utilizó esta información para desarrollar una aplicación de "pick-and-place" donde interviene la interacción entre el hombre y la máquina y además en Martinez, Hernandez y Sanchez 2011 se utiliza un filtro de partículas en vez del filtro de Kalman.

Para realizar el seguimiento de la postura del esqueleto humano, utilizando el modelo 4D-DPM, desarrollamos Martinez y col. 2016 donde se utiliza un modelo de partes deformable (DPM) con imágenes RGBD y con la ayuda de un filtro de Kalman, para corroborar la información dada por el sensor Kinect, y con la ayuda del cálculo de la cinemática utilizando Denavit-Hartenberg, se mejoran los resultados del modelo DPM. En Martinez y col. 2017 se utiliza esta misma información pero además se realiza el modelo geométrico del mismo utilizando poli-esferas el cual nos ayudará a la detección de colisiones como mejora al modelo y ademas nos permite la visualización en 3D del resultado final de la postura del esqueleto del cuerpo humano obtenida.

Al mismo tiempo que se escriben estas líneas, se ha enviado a revisión un nuevo artículo en la revista "International Journal of Advanced Robotic Systems" donde tras una primera resolución favorable, se ha realizado las correcciones sugeridas por los correctores. Además de un nuevo artículo, todavía en fase de desarrollo, para la revista "Sensors".

7.3 Trabajos futuros

Como trabajos futuros, como se ha mencionado en la sección 4.4, se pretende terminar de introducir la restricción 3 del filtro de partículas para seguir mejorando la precisión del modelo 4D-DPM.

Como se ha visto, la herramienta utilizada para el desarrollo de los experimentos es "MATLAB", pero esta herramienta no es la mejor para realizar un algoritmo en tiempo real. Por ello se quiere estudiar la posibilidad de trasladar el código a "C++", además de realizar una programación paralela para el procesamiento de las imágenes en sistemas de varios núcleos o con la utilización de la GPU.

Además se quiere ampliar la base de datos CAD60 para introducir nuevos movimientos específicos con diferentes sujetos y ambientes.

Se estudiará la inclusión de la cadena cinemática del tronco para estudiar los beneficios que esta puede llegar proporcionar.

Se esta diseñando una página web para difundir los resultados de la presente tesis, la dirección web es: `http://4d-dpm.ai2.upv.es/`.

Apéndice A

Anexo: Estado del arte anterior

En este capítulo se realiza un repaso sobre el estado del arte del problema planteado. Inicialmente se realiza una introducción del tema. Posteriormente se describen las posibles aplicaciones, y se compara las distintas taxonomías. Finalmente se revisan los trabajos más destacados clasificándolos según la taxonomía que se considera más apropiada.

A.1 Estado del arte anterior.

A.1.1 Introducción

El primer trabajo sobre captura del movimiento humano fue desarrollado por Muybridge 1955, además Muybridge está considerado el padre de los dibujos animados por su trabajo en las primeras películas y animaciones. El estudio incluía la toma de fotografías a intervalos de tiempo discretos de un sujeto realizando un movimiento. En 1973 el psicólogo Johansson realizo su famoso trabajo "Moving Light Display" (MLD) que consistía en colocar marcas reflectivas en las articulaciones de la persona a estudiar y filmar el movimiento. Posteriormente propuso a dichas personas que tratasen de identificar los movimientos solo visualizando las marcas reflectantes que les había colocado. Estos experimentos fueron los primeros pasos de una línea de investigación en alza.

La captura del movimiento es el análisis de una escena obteniendo como resultado datos numéricos del movimiento realizado por la persona sujeta a estudio, o como enuncio Menache: "La captura del movimiento es el proceso de registrar un movimiento y convertirlo en términos matemáticos útiles, siguiendo en el tiempo un número de puntos clave en el espacio, y combinando estos puntos obtener una representación 3D de la acción realizada" Menache 1999.

Hoy en día hay un gran interés por el tópico de captura del movimiento humano, y el número de artículos publicados en esta área crece exponencialmente. El creciente interés se debe a varios factores.

Avances en la tecnología hardware, y el continuo descenso de los precios de los dispositivos de videocaptura y procesamiento han permitido a la sociedad acceder a estos dispositivos. Igualmente, el propósito de la investigación potencia la atención y el interés de los investigadores para obtener resultados en este tópico. Segmentar objetos no rígidos los cuales pueden provocar auto-oclusiones es un problema muy complicado y difícil de resolver. Agravado por el hecho de que la estimación del movimiento debe ser precisa ya que el ojo humano percibe los pequeños errores que pueda haber en los datos capturados.

El creciente número de posibles aplicaciones es otro factor influyente en el interés por la comunidad científica.

A.1.2 Aplicaciones

En los siguientes apartados se describen algunas aplicaciones de las tecnologías de la captura del movimiento. A continuación no se presenta un detallado estudio sino una visión general de las distintas alternativas en las que se aplica las tecnologías de captura del movimiento humano. La siguiente tabla, A.1, muestra un esquema general.

Tabla A.1: Posibles aplicaciones en el campo de captura del movimiento.

Interfaces de usuarios avanzadas	Interfaces sociales
	Interpretación de lenguajes de signos
	Aplicaciones interpretan gestos
Codificación basado en modelo (MPG-4)	Compresión de video
	Control de animación
Análisis del movimiento	Estudios clínicos
	Coreografía de danza
	Estudio deportivo
	Indexación de contenido de TV
Video vigilancia	Interiores y exteriores
	Reconocimiento de andares
Realidad virtual	Interacción en mundos virtuales
	Animación de personajes
	Teleconferencia
	Producción de películas
	Videojuegos

A.1.2.1 Interfaces de usuario avanzadas

El principal objetivo de estas interfaces es proporcionar métodos más naturales de comunicación entre el hombre y la máquina. Por ello, también son conocidas como aplicaciones de "interacción humano-máquina" (IHM). Se han realizado muchas investigaciones con el objetivo de crear máquinas capaces de comprender los métodos de comunicación de los seres humanos. Uno de estos, el habla, ha recibido gran atención por parte de la comunidad científica, y ha habido muchos avances, los cuales ya se han introducido en la sociedad. Sin embargo, el problema no se ha resuelto completamente. Un interfaz basado en la visión por ordenador capaz de reconocer movimientos o gestos podría ser utilizado de forma aislada, o con otros interfaces tales como el habla.

En este sentido el sistema de visión podría ayudar al sistema de reconocimiento del habla para discernir sonidos leyendo los labios. Otra posibilidad es utilizar un interfaz basado en visión para reconocer lenguajes de signos, o bien para interactuar con una aplicación haciendo uso de un conjunto de gestos predefinidos.

A.1.2.2 Codificación MPEG-4

El grupo MPEG aspira a codificar video de forma eficiente, esto permite utilizar menos ancho de banda durante la transmisión. El grupo MPEG-4 se centra en el problema de compresión de medias basados en un modelo. Esencialmente el grupo busca formas de comprimir la información necesaria para mostrar una escena que está compuesta por imágenes reales y generadas por ordenador. Un ejemplo sencillo de esto es codificar la posición de una cara en una escena y transmitir solamente los cambios que suceden dentro de la región de la cara. Un ejemplo más complejo es codificar la malla de la cara de una persona de forma que es generada a partir de un conjunto pequeño de parámetros clave.

Actualmente el método de transmisión en secuencias de personajes de animación consiste en enviar la malla completa para cada cambio sucedido.

A.1.2.3 Análisis de movimiento

Se comercializan sistemas para la captura del movimiento humano. Estos sistemas se usan para realizar análisis deportivos o clínicos. En estudios clínicos se necesita una gran precisión para obtener una solución en pacientes con dificultades motrices. En estudios deportivos se analiza el sujeto que se quiere mejorar comparándolo con un sujeto referencia, normalmente el número uno de la especialidad. Estos sistemas de captura usan marcadores para obtener la precisión deseada, llegando a calcular la traslación que sufre el fémur respecto la tibia en un corredor. También se han propuesto sistemas que permiten crear una base de datos de vídeos indexado por el tipo de movimiento que se realiza.

A.1.2.4 Videovigilancia

Sistemas de captura del movimiento utilizan el tipo de andar como una característica biométrica para reconocer individuos. También los sistemas de visión se utilizan para detectar comportamientos anormales.

Un sistema podría aprender comportamientos normales y disparar una alarma cuando se produce un comportamiento anormal. Estos sistemas deberían ser robustos para operar en entornos reales y sin restricciones.

A.1.2.5 Realidad virtual

De largo, la aplicación más común de los sistemas de captura del movimiento es la animación de personajes virtuales por ordenador. Los animadores graban el movimiento de un actor y posteriormente aplican este movimiento a un personaje virtual. Esto les proporciona un movimiento muy real a los personajes virtuales. El uso de estos personajes virtuales va desde efectos especiales de TV a video juegos.

Como se ha visto existe una gran variedad de posibles aplicaciones, y en general la solución propuesta difiere para cada tipo de aplicación. Estas posibles aplicaciones han alimentado el interés del mercado mundial y de la comunidad científica.

A.1.3 Taxonomía

El número de artículos publicados recientemente es indicativo del número de soluciones propuestas. Las revisiones agrupan los artículos en una serie de categorías. Es importante seleccionar una esquema de clasificación relevante y bien estructurado para no sobre-clasificar o infra-clasificar los distintos enfoques en función de las similitudes o diferencias que puedan existir. Además, es importante que dicho esquema sea jerárquico y homogéneo. Para realizar dicho esquema se puede tener en cuenta una serie de distinciones.

- Enfoque 2D vs. enfoque 3D.
- Basado en modelo vs. no basado en modelo.
- Tipo de modelo (figura de palos, modelo estadístico,...).
- Cinético vs. Cinemático.
- Tipo de sensores (luz visible, luz infrarroja, ultrasonido,...).
- Número de sensores.
- Sensores móviles vs. sensores estáticos.
- Seguimiento vs. Reconocimiento.
- Estimación de la postura vs. Seguimiento.
- Estimación de la postura vs. Reconocimiento.
- Aplicaciones.

- Una persona vs. varias personas.
- Distribuido vs. Centralizado.
- Suposiciones sobre el tipo de movimiento (rígido, no rígido, elástico,...).

Sin embargo, ninguna de estas clasificaciones es suficientemente detallada y amplia.

Por lo tanto, muchos autores no podrían introducirse dentro de este marco de distinciones. Varios trabajos de clasificación han sido publicados, cada uno dando un punto de vista particular de la clasificación. Cedras y Shah 1995 da una visión de diferentes métodos dentro de la extracción del movimiento, los cuales todos pueden ser clasificados como de flujo óptico o puesta en correspondencia. Posteriormente, una taxonomía para el problema de la captura del movimiento humano se propone como: reconocimiento de acciones, reconocimiento de las distintas partes del cuerpo, y estimación de la configuración del cuerpo, Aggarwal y Cai 1995 da una revisión basada en el tipo de movimiento, articulado o elástico. A continuación, procede a clasificar tanto los tipos de movimientos como el uso o no de un modelo. En la última revisión de Aggarwal y Cai 1999 usa la misma taxonomía que ha usado Cedras aunque usan diferentes nombres para las tres clases. Las tres clases principales son divididas en subclases dando un esquema de clasificación algo complejo. Más recientemente, una revisión realizada por Gavrila 1999 aporta una introducción más general con un enfoque especial en las diferentes aplicaciones. Esta taxonomía consiste en enfoques 2D sin usar modelo, enfoques 2D con un modelo explícito y enfoques 3D. Sobre estas tres clases describe las soluciones que se ocupan de realizar reconocimiento. Las revisiones de Ozer y Wolf 2002, Aggarwal y Cai 1999 y Gavrila 1999 proporcionan una clasificación robusta del área de investigación. Sin embargo, Moeslund 2001 proporciona la taxonomía más lógica y flexible. Moeslund categoriza los trabajos no por los enfoques realizados o técnicas utilizadas, sino por la etapa de la captura del movimiento que trata de solventar.

Posteriormente divide estas etapas en diferentes enfoques. Este punto de vista es similar a Gavrila 1999 quien permite que una solución pueda ser clasificada en varios enfoques, un artículo puede ser clasificado tanto en reconocimiento como en cualquiera de las otras clases. Esto reduce el problema de clasificar soluciones que están ubicadas entre clases o en más de una clase. Moeslund postula que la captura del movimiento y sus soluciones pueden ser clasificadas de una forma sistemática. El autor repasa como los investigadores abordan el sujeto en un sentido temporal. Para ilustrarlo, primero el sistema es inicializado, esto incluye construir el modelo apropiado y realizar tareas tales como estudiar el fondo o calibrar las cámaras,

o por ejemplo, encontrar una persona en una escena estática. Posteriormente, la persona es seguida cuadro a cuadro. A continuación, se da una estimación de la postura en cada instante de tiempo capturado. Finalmente, se lleva a cabo el reconocimiento de la acción, y para ello se utiliza un periodo de tiempo.

La siguiente figura, figure A.1, muestra una visión general del esquema de clasificación desde un punto de vista de requisitos del sistema. Cada etapa del análisis del movimiento humano está representada en el lado izquierdo. Un enfoque no tiene por qué tratar de solventar todas las etapas, tampoco es necesario solventar una etapa anterior para solventar las sucesivas etapas. De hecho, muchos autores intercambian información entre las etapas para hallar la solución.

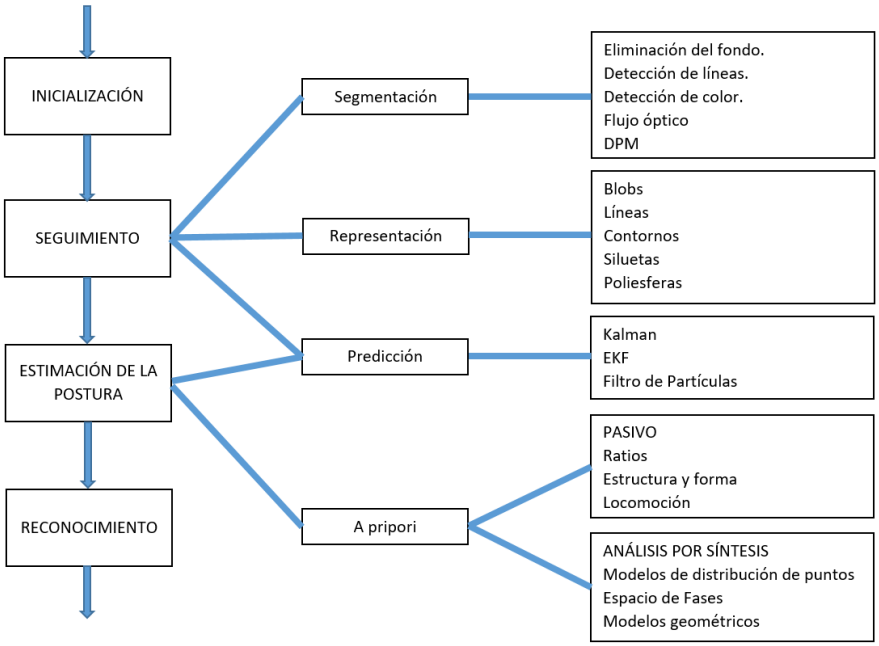


Figura A.1: Esquema de clasificación. Esquema de clasificación de trabajos en el área de análisis del movimiento humano. Las etapas están representadas en el lado izquierdo. El centro son las técnicas. El lado derecho es una descripción más detallada de las técnicas

Se ha discutido las taxonomías utilizadas por varios autores en sus revisiones. El enfoque dado por Moeslund 2001 se está aceptando como la taxonomía más acertada debido a que abarca todos los trabajos y es flexible, a continuación se revisan las cuatro etapas que se han mencionado.

A.1.4 Inicialización

La inicialización consiste en configurar el sistema para que se lleve a cabo las tareas de procesamiento de datos, como su nombre indica la configuración se realiza al inicio, es decir, antes de que se realice el procesamiento, aunque la inicialización puede comprender un periodo.

Todas las tareas que se llevan a cabo durante el procesamiento se descartan como parte de la inicialización. Una buena inicialización puede ser la clave para obtener buenos resultados en las etapas posteriores. Las líneas de investigación principales en la etapa de inicialización abordan la estructura cinemática, la forma 3D, la apariencia de color o textura y la postura.

Trabajos como Barron y Kakadiaris 2003; Parameswaran y Chellappa 2004 tratan de obtener las medidas y la postura identificando las articulaciones manualmente en una sola imagen no calibrada. Otros tratan de obtener las medidas de forma automática a partir de una secuencia de imágenes desde una sola cámara Song, Goncalves y Perona 2003; Krahnstoever y Sharma 2004.

La inicialización de la estructura cinemática se limita a estimar las distancias entre las articulaciones. Los sistemas comerciales basados en marcadores requieren un conjunto de secuencias fijas de movimientos, cada una en cada grado de libertad por separado. Otros métodos Cheung, Baker y Kanade 2003; Brostow y col. 2004 tratan de obtener la estructura cinemática a partir de la reconstrucción 3D obtenida a partir de varias cámaras. Además de obtener el modelo antropométrico (la estructura y sus dimensiones) es muy útil la información relativa a las limitaciones de movimiento de las articulaciones, Moeslund y Granum 2001b; Moeslund y col. 2002; Mulligan 2005 enfocan su investigación a usar dichas restricciones. Demirdjian, Ko y Darrell 2003b restringe la postura del tronco superior basándose en múltiples movimientos aprendidos. Otras investigaciones modelan los límites de las articulaciones a partir de medidas tomadas en secuencias con marcadores Herda, Urtasun y Fua 2005.

Para aproximar el modelo al sujeto se usa un modelo humano genérico (también llamado humanoide). Para representar las distintas partes del cuerpo se usan primitivas simples (cilindros, conos o elipsoides) o superficies, estas primitivas complementan el modelo cinemático. En trabajos previos Hilton y col. 1999 refina un modelo genérico basándose en las vistas frontal y lateral tomadas desde una sola cámara.

Posteriormente se le aplica la textura obtenida para aportar mayor realismo. En los trabajos de Carranza y col. 2003; Plankers y Fua 2003; Starck e Hilton 2003 se captura de forma simultánea desde varias cámaras calibradas. El uso

de escáneres 3D permite obtener un modelo más preciso del cuerpo. En Allen, Curless y Popovic 2002b se usa múltiples escaneados de una persona en diferentes posturas para generar el modelo. Investigaciones como Thalmann y Seo 2004 usan bases de datos de escaneados 3D para hallar relaciones entre las medidas de una persona.

La gran variabilidad de apariencia entre personas y en una misma persona es muy grande debido a la vestimenta. Generalmente se usa un modelo estadístico de color para realizar el seguimiento. En algunos casos se obtiene una textura a partir de varias imágenes Carranza y col. 2003; Starck e Hilton 2003.

Sidenbladh y Black 2003 obtiene el modelo de color a partir de un conjunto de ejemplos para cada parte del cuerpo. Roberts, McKenna y Ricketts 2002 modela la apariencia de cada parte del cuerpo con histogramas multimodales. En las publicaciones de Ronfard, Schmid y Triggs 2002b; Ramanan, Forsyth y Zisserman 2005; Micilotta, Ong y Bowden 2005 la tendencia es usar detectores de partes del cuerpo para localizar personas. Ramanan, Forsyth y Zisserman 2005 detecta las posturas clave en una secuencia de andar, e inicializa un modelo local de apariencia para detectar las partes del cuerpo en los cuadros intermedios.

A.1.5 Seguimiento

En los últimos años los algoritmos de seguimiento se han centrado principalmente en las aplicaciones de videovigilancia, realizando seguimientos con oclusiones, y detectando personas en una sola imagen.

El concepto de seguimiento en análisis del movimiento humano se usa de forma diferente en la literatura. Se puede tomar como correspondencia temporal: cuando se realiza una correspondencia de las personas detectadas en cada cuadro con las personas que había en el cuadro anterior; o segmentación de la figura: consiste en separar en la imagen la silueta de la persona del fondo. Según el tipo de medida utilizado para realizar esta segmentación podemos categorizarlos en base a: movimiento, apariencia, forma o profundidad. Previo a describir las distintas técnicas veremos los avances en las técnicas de eliminación de fondo, a menudo usado como primer paso en la segmentación.

A.1.5.1 Eliminación del fondo

En 1998 Stauffer y Grimson 1998 mejoraron las técnicas existentes de eliminación de fondo, permitiendo su uso en exteriores. Para ello cada píxel se modela mediante una "mixture of Gaussians" (MoG) que se actualiza en cada cuadro, permitiendo pequeños cambios del fondo, pero no grandes cambios como la aparición de nubes. A partir de entonces se han producido avances en la técnica, quedando dividido en representación del fondo, clasificación, actualización del fondo e inicialización del fondo.

La representación MoG se puede realizar en el espacio de color RGB o en otros espacios de color. Se han usado diferentes representaciones basadas en distintos conceptos. Elgammal, Harwood y Davis 2000 usan los últimos N cuadros para construir una aproximación basada en núcleo. Haritaoglu, Harwood y Davis 2000 representa el valor mínimo, el máximo y el mayor cambio posible entre dos cuadros consecutivos. Heikkila y Pietikainen 2006 representan cada píxel como una secuencia de bits, de esta forma construyen un modelo invariante a cambios de iluminación monotónicos. La elección de una u otra representación no solo influye en la precisión sino también en la velocidad de procesamiento. Cucchiara y col. 2003 usan un único valor para representar cada píxel pero obtiene resultados buenos y rápidos debido a un buen sistema de clasificación y de actualización.

Después de la eliminación del fondo aparecen falsos positivos y negativos, debido por ejemplo a sombras. El uso de técnicas de filtrado puede mejorar los resultados Elgammal, Harwood y Davis 2000; Cucchiara y col. 2003; Zhao y Nevatia 2004a; Guha, Mukerjee y Venkatesh 2005. Métodos recientes tratan de identificar los píxeles incorrectos, y clasificarlos en: fondo invariante, cambios debidos al autoiris, sombras, brillos, objetos en movimiento, sombras causadas por los objetos en movimiento, objetos fantasma (falsos positivos), etc. Horprasert, Harwood y Davis 1999; Cucchiara y col. 2003; Bradski y Davis 2002. Dichos clasificadores se basan en el color, gradiente, información del flujo, y umbrales por histéresis.

En exteriores el valor de un píxel del fondo varía en el tiempo por lo que se hace necesario algún método para actualizar dicho valor. Pequeños cambios pueden solventarse actualizando recursivamente el modelo Stauffer y Grimson 1998; Elgammal, Harwood y Davis 2000; McKenna y col. 2000; Cucchiara y col. 2003. Con el fin de tener en cuenta cambios súbitos Kim y col. 2005 usa un conjunto de colores clave, que denomina "codebook".

El modelo correspondiente al fondo debe construirse en una fase de inicialización. Los primeros trabajos asumen que no hay ningún objeto en movimiento en el periodo de inicialización. Esta suposición no es válida y trabajos más recientes

se centran en lograr una inicialización con objetos en movimiento. Una posible solución es detectar los píxeles que pertenecen a objetos usando un filtro temporal de la mediana Gloyer 1995; Haritaoglu, Harwood y Davis 2000. Chu, Jenkins y Mataric 2003 lo combina con un detector del color característico de la piel para eliminar las personas. El método de "codewords" usa un filtro temporal posterior a la fase de inicialización para eliminar cualquier palabra que no se ha repetido en un periodo de tiempo Kim y col. 2005.

A.1.5.2 Segmentación basada en movimiento

La segmentación basada en movimiento se basa en hallar la diferencia entre dos cuadros consecutivos. El movimiento se calcula bien por flujo o bien por diferencia entre imágenes. Eng y col. 2003 hace uso de flujo óptico. El flujo óptico es sensible al ruido por lo que se usa flujo de imagen. Gonzalez y col. 2003 sigue unas características KLT para obtener los vectores de flujo. Bradski y Davis 2002 halla los vectores de flujo a partir de los gradientes en un conjunto de imágenes de historia del movimiento (MHI) Davis y Bobick 1997.

La diferencia entre imágenes se adapta rápidamente a los cambios en la escena, pero no se detecta los píxeles de una persona que no se ha movido o si los vecinos son similares. Una versión mejorada usa tres imágenes consecutivas Haritaoglu, Harwood y Davis 2000; Kale y col. 2002; Collins y col. 2000. Viola, Jones y Snow 2005 usa otra técnica de diferenciación, donde se combina un conjunto de características sencillas. Otros trabajos recientes donde se utiliza la segmentación basada en movimiento es Bera y col. 2016; Marin y col. 2013; Thanh Nguyen y col. 2010.

A.1.5.3 Segmentación basada en apariencia

La segmentación basada en la apariencia de la persona se basa en que la apariencia de la persona y el fondo es diferente y en que la apariencia entre personas es diferente. Las soluciones trabajan construyendo un modelo de apariencia de cada persona, posteriormente se compara la predicción de este modelo en las sucesivas imágenes. Algunos de estos métodos no tienen un contexto temporal, en contraposición a los métodos donde el modelo de apariencia de la persona se aprende o actualiza en base a las imágenes previas.

A.1.5.3.1 Sin contexto temporal

Los métodos sin contexto temporal se usan para detectar personas en una imagen aislada Mohan, Papageorgiou y Poggio 2001, detectar personas entrando en la escena Oliver, Rosario y Pentland 2000, o para indexar imágenes en una base de datos Ozer y Wolf 2002. En Utsumi y Tetsutani 2002 la imagen es dividida en bloques, para cada bloque se calcula la media y la matriz de covarianza de las intensidades. A partir de aquí, se construye una matriz distancia donde cada elemento representa la distancia de Mahalanobis entre dos bloques. La detección se basa en el hecho de que en imágenes sin personas la distancia entre bloques próximos será mayor que en las imágenes que contienen personas.

A.1.5.3.2 Con contexto temporal

Son aquellos métodos donde el modelo es aprendido o actualizado en imágenes previas. Se utiliza bien para detectar los objetos o para clasificar los píxeles de los objetos. Estos métodos pueden operar a nivel de píxel o a nivel de región. En muchos sistemas el color se representa como un histograma McKenna y col. 2000; Comaniciu, Ramesh y Meer 2003; Xu y Puig 2005; Hu, Hu y Tan 2004 o como una MoG Khan y Shah 2000; Yang, Duraiswami y Davis 2005. Los histogramas normalmente se comparan usando la distancia de "Bhattacharyya", mientras que las representaciones MoG se comparan usando la distancia de "Mahalanobis".

A.1.5.4 Segmentación basada en forma

La forma o silueta de una persona es a menudo muy diferente a la forma de otros objetos en una escena. Por ello, la detección basándose en la forma puede ser una vía de abordar el problema muy productiva. Los avances en las técnicas de eliminación de fondo permiten obtener siluetas en entornos no controlados. Como se ha hecho en los métodos basados en apariencia se distinguirá los que son libre de contexto temporal y los que no.

A.1.5.4.1 Sin contexto temporal

Zhao y Thorpe 2000 usa información de profundidad extraída de la silueta en una imagen. Una red neuronal entrenada se encarga de determinar si las siluetas extraídas corresponden o no a personas. Leibe, Seemann y Schiele 2005 aprende los contornos de personas caminando y los almacena como plantillas. Cada plantilla se compara a diferente escala y usando la puesta en correspondencia de "Chamfer", con una imagen contorno extraída de la imagen de entrada. Dalal y Triggs 2005a usa una SVM para detectar personas en una ventana de píxeles.

A.1.5.4.2 Con contexto temporal

Cuando se tiene en cuenta el contexto temporal, los métodos basados en forma pueden utilizarse para seguir personas en el tiempo. Haritaoglu, Harwood y Davis 2000 lleva a cabo una correlación de contornos binarios entre el contorno de la silueta en el cuadro anterior y los alrededores del cuadro actual. Davis, Philomin y Duraiswami 2000 usa un modelo de "distribución de puntos" (PDM) para representar el contorno de la persona. La configuración de contorno del cuadro anterior más parecida se usa para predecir la localización en el cuadro actual usando filtro de partículas. Las predicciones se evalúan comparando los bordes del modelo con los bordes de la imagen. En situaciones de oclusión parcial los métodos basados en forma suelen fallar debido a la falta de información global.

Las mejoras tratan de incluir la detección de las personas basándose en la detección de unas partes de la persona. En el trabajo de Wu y Nevatia 2005 se detectan cuatro partes del cuerpo: todo el cuerpo, cabeza-hombros, torso y piernas.

A.1.5.5 Segmentación basada en profundidad

La segmentación de la persona usando información de profundidad se basa en la idea de que la persona destaca en un entorno 3D. Los métodos se basan directamente en información 3D extraída de la escena Ivanov, Bobick y Liu 2000; Haritaoglu, Flickner y Beymer 2002; Hayashi y col. 2004 o indirectamente combinando características extraídas de diferentes vistas Mittal y Davis 2005; Yang, Banos y Guibas 2003; Iwase y Saito 2004.

Las técnicas de eliminación del fondo son sensibles a los cambios de iluminación. Sin embargo, una solución basada en la profundidad no presenta dicho inconveniente. El fondo se modela como un mapa de profundidad y cada nuevo cuadro es comparado con el modelo para obtener los objetos segmentados. Sin embargo,

un algoritmo estéreo en tiempo real no es viable si no se usa un hardware especial Lim y col. 2005. Una solución para evitar este problema fue propuesta por Ivanov, Bobick y Liu 2000 donde no es necesario un mapa de profundidad, en su lugar, aprende la correspondencia entre píxeles de dos cámaras.

Haritaoglu también ha producido avances en la detección de personas basándose en la profundidad Haritaoglu, Flickner y Beymer 2002 donde la información de profundidad se proyecta al plano del suelo. Las personas son localizadas realizando una búsqueda del perfil 3D hombro-cabeza-hombro.

Mittal y Davis 2005 detecta las personas en cada cámara usando un método basado en apariencia. El centro de cada persona detectada se combina con los hallados en las otras cámaras usando restricciones epipolares. Los puntos 3D resultantes se proyectan al plano del suelo y se representan probabilísticamente usando modelos Gaussianos y probabilidades de oclusión. Yang, Banos y Guibas 2003 combina las siluetas obtenidas de diferentes cámaras. Las interpretaciones incorrectas son eliminadas usando criterios de tamaño así como un histórico temporal.

A.1.5.6 Correspondencia temporal

Una de las tareas principales de un algoritmo de seguimiento es encontrar las correspondencias temporales. Esto es, dado el estado de N personas en el cuadro anterior y los datos de entrada del cuadro actual, cuales son los estados de las mismas personas en el cuadro actual. Aquí interpretamos el estado como la posición de la persona en la imagen, pero pueden ser otros atributos como: la posición 3D, forma y color.

Inicialmente se consideraba el problema en entornos controlados y con pocas personas en la escena. Actualmente, los algoritmos se encaminan a entornos no controlados y exteriores, donde se presentan muchas personas y oclusiones. Además del problema de segmentación presentado anteriormente, existe otro problema igualmente importante que es como tratar múltiples personas que pueden oclusionarse unos a otros. A continuación se revisan los avances relacionados con el antes y el después de la oclusión y los avances durante la oclusión.

A.1.5.6.1 Antes y después de la oclusión

Previo al seguimiento debe construirse un modelo de cada objeto a seguir. Métodos recientes pretenden realizarlo de forma automática. Una forma de hacerlo es buscar la aparición de objetos grandes posiblemente cerca del borde de la imagen Haritaoglu, Harwood y Davis 2000; McKenna y col. 2000; Roth, Doubek y Gool 2005.

Khan y Shah 2000 hacen corresponder gaussianas 1D a los píxeles objetos proyectados en el eje X. Si el número de coincidencias es mayor que el número de personas previsto entonces se considera que ha entrado una nueva persona en la escena.

Una vez comenzado el seguimiento el problema consiste en hacer corresponder las predicciones con las mediciones u observaciones.

Recientemente se ha solventado usando una matriz de correspondencia, dicha matriz tiene los objetos a seguir en una dirección y las mediciones de los objetos en la otra dirección. Cada elemento de la matriz se calcula como la distancia entre la predicción y la medida del objeto. Analizando las filas y columnas se pueden producir los siguientes casos: un nuevo objeto, un objeto perdido, una correspondencia, una división en dos y una unión de dos objetos. En el caso de división o unión de objetos la matriz no puede ser resuelta directamente y se necesitan métodos "ad hoc". Por ejemplo analizando los vectores de movimiento y el área de cambio de cada objeto Guha, Mukerjee y Venkatesh 2005; McKenna y col. 2000; Xu y Puig 2005; Cucchiara y col. 2004.

Otra alternativa es realizar optimizaciones globales. En Zhao y Nevatia 2004b; Smith, Perez y Odobez 2005 usan un filtro de partículas, donde cada estado es una configuración multiobjeto. Los objetos pueden entrar y salir de la escena lo que significa que el número de elementos del vector estado puede variar. En el trabajo de Li, Hilton e Ilingworth 2002 se usa una optimización global basada en árbol para hacer corresponder múltiples objetos en múltiples vistas. Esta solución se usa para realizar el seguimiento de manos, cabeza y pies para estimar la posición del cuerpo entero.

A.1.5.6.2 Durante la oclusión

En los trabajos anteriores el seguimiento durante la oclusión no se ha contemplado, en su lugar se utiliza el grupo para actualizar el estado de cada objeto en particular. Sin embargo, esto provoca que resulte imposible actualizar los modelos de cada objeto en particular, lo que puede probar un seguimiento poco fiable después de que se separe el grupo. Además, es difícil analizar las interacciones entre personas durante las oclusiones, por el hecho de que se representen como un objeto. Por lo tanto, se está investigando el problema de hallar correspondencias durante las oclusiones.

En estos sistemas la primera tarea es detectar que está sucediendo una oclusión, esto se puede realizar usando la matriz de correspondencia que se ha mencionado anteriormente como en Khan y Shah 2000; Capellades y col. 2003; Roth, Doubek y Gool 2005. Khan y Shah 2000 detectan una no-oclusión como una situación donde los objetos detectados están lejos uno del otro. Capellades y col. 2003 define una unión como una situación donde el número total de objetos ha decrementado y dos o más objetos del cuadro anterior superponen con otro objeto en el cuadro actual. En el trabajo de Roth, Doubek y Gool 2005 una unión se detecta como uno de los ocho posibles tipos de oclusiones basándose en el orden de profundidad y la disposición de las fronteras de los objetos.

Se han presentado diferentes soluciones en publicaciones recientes para asignar píxeles a objetos en concreto durante las oclusiones. Una aproximación local es asignar cada píxel al modelo predicho más parecido usando un modelo probabilístico Khan y Shah 2000; Park y Aggarwal 2002. Las aproximaciones globales intentan clasificar los píxeles basándose por ejemplo en la asunción que la gente en un grupo permanece junta respecto a la cámara Xu y Puig 2005; Haritaoglu, Harwood y Davis 1998b. En el trabajo de McKenna y col. 2000 se halla el orden de profundidad de forma explícita. Durante las oclusiones la similitud de cada píxel con el objeto (persona) se calcula mediante la regla de Bayes. En Roth, Doubek y Gool 2005 el orden de profundidad se obtiene asumiendo un suelo plano. Esto conlleva que los objetos más cercanos a la cámara tienen una coordenada vertical menor. Xu y Puig 2005 generaliza esta idea usando geometría proyectiva para encontrar la línea en la imagen que corresponde al horizonte en la escena 3D. Los objetos más alejados a la cámara son aquellos que están más cerca al horizonte.

A.1.6 Estimación de la postura

La estimación de la postura se refiere al proceso de estimar la configuración cinemática subyacente o la estructura del esqueleto de la persona. Este proceso puede ser una parte integrada del proceso de seguimiento como en las soluciones de análisis por síntesis basándose en un modelo o puede llevarse a cabo directamente a partir de las observaciones cuadro a cuadro. Las revisiones separan los algoritmos de estimación de la postura en tres categorías basándose en el uso de un modelo a priori.

- Sin Modelo: Los métodos donde no hay un modelo a priori. Estos métodos optan por una solución de abajo a arriba, para seguir y etiquetar las partes del cuerpo en 2D Wren y col. 1997 o mapeando directamente a partir de observaciones de la imagen en secuencias a la postura 3D.
- Uso indirecto del modelo: Estos métodos usan un modelo a priori en la estimación de la postura como una referencia para guiar la interpretación de las mediciones. Se etiquetan partes del cuerpo usando relaciones de aspecto entre los miembros Cai, Mitiche y Agarwal 1995 o reconociendo la postura Haritaoglu, Harwood y Davis 1998a; *Hierarchical Models for Visual Recognition and Learning of Objects, Scenes, and Activities, Studies in Systems, Decision and Control, Cap. 6* 2015.
- Uso directo del modelo: Hacen uso de una representación geométrica 3D explícita de la forma de la persona y la estructura cinemática para reconstruir la postura. La mayoría de las soluciones usan métodos de análisis por síntesis para optimizar las similitudes entre el modelo proyectado y las observaciones en la imagen Hogg 1983; Wachter y Nagel 1997.

En Gong y col. 2016 se detallan ciertos trabajos para poder estimar la postura a partir de diferentes métodos.

A continuación se revisan las contribuciones y avances en cada categoría de los algoritmos de estimación de la postura. Una serie de tendencias pueden identificarse a partir de la literatura. Tres líneas de investigación las cuales han recibido considerable atención: la introducción de soluciones probabilísticas para detectar las partes del cuerpo y fusionar dicha información en la categoría de sin modelo; la incorporación de modelos de movimiento aprendidos para restringir el movimiento 3D recuperado; y el uso de técnicas de muestreo estocástico en el análisis por síntesis basado en modelo para mejorar la robustez de la estimación de la postura 3D.

Dos importantes puntos están en relación directa con el problema de estimación de la postura: una cámara vs. varias cámaras; y estimación 2D de la postura en el plano imagen vs. reconstrucción de la postura 3D.

El problema más dificultoso consiste en recuperar la postura 3D desde una sola imagen, donde en la literatura se puede encontrar resultados. También podemos encontrar resultados en la estimación 2D a partir de una cámara y la estimación 3D a partir de varias cámaras.

A.1.6.1 Sin modelo

Una tendencia para superar las limitaciones de realizar el seguimiento en una secuencia larga ha sido la investigación de la estimación directa de la postura a partir de una imagen. Dos aproximaciones caen dentro de esta categoría: Fusión probabilística de las partes donde primero se detectan las partes simples del cuerpo y posteriormente se fusionan para estimar la postura 2D; y métodos basados en ejemplos donde se aprende directamente el mapeo del espacio 2D de la imagen al espacio 3D del modelo.

A.1.6.1.1 Fusión probabilística de las partes

La ventaja de estos métodos frente al seguimiento es que no asumen pequeños cambios entre cuadros, por lo que es robusto a movimientos rápidos. Puede introducirse información temporal para estimar configuraciones coherentes en una secuencia. Forsythe y Fleck 1997 introdujeron el concepto de planos del cuerpo para representar personas o animales como una fusión estructurada de partes aprendida a partir de imágenes. En la misma dirección Iofee y Forsyth 1999 hace uso de estructuras pictóricas para estimar la configuración de partes del cuerpo 2D a partir de una secuencia de imágenes. Se han combinado detectores de partes del cuerpo con el fin de localizar múltiples personas en escenarios confusos y con oclusiones Mohan, Papageorgiou y Poggio 2001; Wu y Nevatia 2005. Las partes del cuerpo se detectan usando la forma 2D Roberts, McKenna y Ricketts 2004, clasificadores SVM Ronfard, Schmid y Triggs 2002b, "AdaBoost" Haritaoglu, Harwood y Davis 2000 y modelos de apariencia inicializados localmente Ramanan, Forsyth y Zisserman 2005.

Algunos trabajos recientes introducen soluciones para estimar la postura 2D a partir de una imagen. Ren, Berg y Malik 2005 usa restricciones entre las partes del cuerpo para fusionar las detecciones de las partes del cuerpo en una postura

2D. Las restricciones incluyen relación de aspecto, escala, apariencia, orientación y conectividad.

Una contribución importante de las soluciones basadas en la fusión probabilística de las partes es la estimación de la postura 2D en escenas naturales y confusas a partir de una imagen. Esto supera las limitaciones de métodos previos de estimación de la postura, los cuales requieren escenas estructuradas, modelos a priori precisos o múltiples cámaras.

A.1.6.1.2 Basados en el ejemplo

Los métodos basados en el ejemplo comparan la imagen observada con una base de datos de ejemplos. Brand 1999 usa un "modelo de Markov" (HMM) para representar el mapeo de una secuencia de siluetas 2D en el espacio imagen a un esqueleto en movimiento en el espacio 3D. En este trabajo el mapeo para secuencias de movimiento específicas se aprendió usando imágenes de siluetas de un modelo renderizadas.

HMM se utiliza para estimar la secuencia de posturas 3D más probable a partir de una secuencia de siluetas 2D observadas desde un punto de vista específico. Existen otras soluciones: usando regresiones lineales Agarwal y Triggs 2006c, aprendiendo un modelo probabilístico para localizar las articulaciones Grauman, Shakhnarovich y Darrell 2003, o realizando una búsqueda de siluetas usando la distancia de Chamfer para seleccionar el candidato junto una cadena de Markov para propagar la estimación 3D de la postura de caminar o bailar Howe 2004.

Las soluciones basadas en ejemplos representan el mapeo entre el espacio imagen y el espacio de la postura, proporcionando un potente mecanismo para estimar la postura 3D directamente. Una limitación es la restricción a las posturas o movimientos usados en la fase de entrenamiento. Un gran vocabulario de movimientos puede introducir ambigüedades en el mapeo.

A.1.6.2 Uso indirecto del modelo

Mikic y col. 2002 presenta un sistema integrado para la reconstrucción automática tanto del modelo de la persona como del movimiento a partir de varias cámaras. La construcción del modelo se basa en un punto de vista basado en reglas jerárquicas para localizar las partes del cuerpo y etiquetarlas. Conocimiento a priori de la forma de las partes del cuerpo, relación de tamaños y configuración se utilizan para segmentar el espacio visual. Entonces se aplica un filtro de Kalman extendido

para reconstruir el movimiento entre cuadros. Un proceso de etiquetaje de píxeles permite grandes desplazamientos entre cuadros.

Cheng, Christmas y Kittler 2002 primero reconstruye un modelo de la estructura cinemática, la forma y la apariencia de la persona que usa posteriormente para estimar el movimiento 3D. El seguimiento lo realiza mediante una puesta en correspondencia jerárquica del modelo.

Starck e Hilton 2005 presenta una solución alternativa basada en una puesta en correspondencia de superficies no rígidas de 3D a 3D. Los resultados demuestran la estimación de la postura 3D en movimientos rápidos de la persona llevando ropa ajustada.

Estas soluciones explotan la reconstrucción de la escena desde múltiples cámaras para recuperar tanto la forma como el movimiento. Esta solución es indicada para estudios con múltiples cámaras permitiendo estimar los datos en movimientos complejos.

A.1.6.3 Uso directo del modelo

El uso de un modelo explícito de la cinemática, forma y apariencia de la persona en un contexto de análisis por síntesis es la solución más estudiada en la estimación de la postura a partir de imágenes. Las principales líneas de investigación son: la introducción de técnicas de muestreo estocástico basado en secuencias de "MonteCarlo"; y la introducción de restricciones en el modelo. A continuación se revisan las principales contribuciones a estos avances haciendo uso de múltiples cámaras o una cámara.

A.1.6.3.1 Estimación de la postura desde múltiples cámaras

Hasta el 2000 la mayoría de trabajos de estimación de la postura usaban técnicas determinísticas de gradiente para estimar iterativamente los cambios en la postura Plankers y Fua 2003; Delamarre y Faugeras 2001. El filtro de Kalman extendido se usaba ampliamente para seguir personas con dinámicas de bajo orden para predecir cambios en la postura Wachter y Nagel 1999. Trabajos posteriores han mejorado las soluciones basadas en gradiente usando análisis por síntesis basado en modelos permitiendo movimientos más complejos. Plankers y Fua 2003 demuestran el seguimiento del brazo con auto-oclusiones usando estéreo e información de la silueta. Una limitación de las soluciones basadas en gradiente es el uso de un único estado de la postura que se actualiza en cada cuadro. En Moon

y col. 2016 se utiliza un filtro de Kalman con varios sensores Kinect para para dar solución a las oclusiones. En la práctica si hay un movimiento rápido o ambigüedades visuales la estimación de la postura puede fallar catastróficamente. Para lograr seguimientos más robustos, se usan técnicas de búsqueda determinísticas o estocásticas en el espacio de la postura.

Técnicas de seguimiento estocástico, tales como el filtro de partículas, se introdujeron para el seguimiento robusto de objetos donde suceden cambios repentinos de movimiento o escenarios no controlados provocan fallos. La principal dificultad de su aplicación en la estimación de la postura de la persona es la dimensión del espacio de búsqueda. El número de muestras o partículas necesarias crece exponencialmente con la dimensión. Generalmente un modelo completo de la persona ronda de los 20 a los 30 grados de libertad provocando que el uso directo del filtro de partículas sea computacionalmente inviable. MacCormick e Isard 2000 proponen un muestro particionado del espacio de búsqueda para lograr una estimación eficiente de la postura 2D de objetos articulados como la mano. Sin embargo, esta solución no se puede extender directamente a la estimación del cuerpo entero. Deutscher, Blake y Reid 2000 introdujo el "filtro de partículas templado" (annealed particle filter) que combina una solución templar con un muestreo estocástico para reducir el número de muestras necesario. En cada paso de tiempo el conjunto de partículas se refina mediante una serie de ciclos de templado (de templar el metal), decrementando la temperatura (el ruido) para aproximar la función de correspondencia al mínimo local. Los resultados muestran una reconstrucción de un movimiento complejo tales como una voltereta.

Mitchelson e Hilton 2005 presenta un esquema de muestreo jerárquico estocástico para estimar de forma eficiente la postura 3D de un movimiento complejo o con múltiples personas. Esta aproximación estima inicialmente la postura del torso para cada persona y propaga las partículas con gran precisión para estimar la postura de las partes del cuerpo adyacentes. Trabajos recientes combinan búsquedas estocásticas o determinísticas con técnicas de gradiente para refinar la postura. Carranza y col. 2003 demuestra la estimación del movimiento de cuerpo entero de una persona desde múltiples cámaras combinando una búsqueda determinística con técnicas de gradiente. La estimación de la postura se realiza jerárquicamente empezando con el torso. Para cada parte del cuerpo se realiza una búsqueda de rejilla, primero encuentra un conjunto de postura válidas para el cual la proyección de la posición de la articulación cae dentro de la silueta observada. A continuación se evalúa una función de correspondencia para todas las posturas válidas y determinar la mejor solución. Finalmente se realiza una optimización de gradiente para refinar la solución obtenida anteriormente. Este proceso de búsqueda resulta más eficiente si se aprovecha el hardware gráfico para

evaluar la función de correspondencia que superpone el modelo proyectado sobre la silueta obtenida de cada cámara. En un trabajo relacionado Kehl, Bray y Van-Gool 2005 propone una solución para estimar el cuerpo entero con 24 grados de libertad desde múltiples cámaras. Combina un muestreo estocástico del conjunto de puntos del modelo usados en cada iteración del algoritmo de gradiente. Esto introduce un elemento de búsqueda estocástica a la optimización con lo que evita converger a mínimos locales. El uso de un número de muestras bajo (5) para cada parte del cuerpo junto a un tamaño del paso adaptativo permite un rendimiento óptimo. Los resultados de este trabajo demuestran la reconstrucción de un movimiento complejo como patear o bailar.

En resumen, la introducción de búsqueda y muestreo estocástico han logrado estimar la postura del cuerpo entero en movimientos complejos desde múltiples cámaras. Un alto porcentaje de los trabajos publicados se limitan a estimar la postura del torso, brazos y piernas, y no estiman detalles tales como la orientación de las manos, o la rotación del brazo. Múltiples hipótesis de muestreo logran seguimientos robustos pero no permiten una estimación consistente en el tiempo, el cual debe ser suavizado para obtener resultados visualmente correctos. Todavía existe un abismo entre la precisión de los sistemas comerciales con marcadores y la reconstrucción basada en video sin marcadores.

A.1.6.3.2 Estimación de la postura desde una cámara

Reconstruir la postura de una persona desde una cámara es mucho más difícil que estimar la postura 2D o estimar la postura 3D desde múltiples cámaras. Para resolver la ambigüedad en la reconstrucción normalmente se utilizan restricciones cinemáticas o de movimiento Wachter y Nagel 1999; Bregler, Malik y Pullen 2004. En Liu y col. 2016; Janabi-Sharifi y Marey 2010 se hace uso del filtro de Kalman y además en Wachter y Nagel 1999 se añade al filtro de Kalman restricciones en las articulaciones para estimar el movimiento 3D de una persona. Como se ha mencionado anteriormente forzar a un movimiento en concreto provoca que no sea aplicable a movimientos más complejos. Loy, Eriksson y Sullivan 2004 aporta una solución a partir de cuadros clave para estimar la postura 3D de un movimiento complejo en el ámbito deportivo.

Sminchisescu y Triggs 2003 han investigado la aplicación de muestreo estocástico para la estimación de la postura 3D desde una sola cámara. Observaron que las posturas alternativas que daban buena correspondencia con las observaciones ocurrían en la dirección de incertidumbre. Esto les motivo a introducir un muestreo escalado de covarianza, una extensión del filtro de partículas que incrementa la

covarianza en la dirección de incertidumbre. Posteriormente se busca el mínimo local mediante técnicas de gradiente. Los resultados muestran el seguimiento y reconstrucción 3D desde una sola cámara con complejidad moderada tales como andar con cambios de dirección.

También se han investigado soluciones probabilísticas que fusionan las partes con un nivel superior de conocimiento sobre la cinemática y la forma. Lee y Cohen 2004b combinan un mapa probabilístico que representa la similitud de las partes en diferentes posiciones 3D con un modelo explicito 3D, como resultado se recupera la postura 3D desde una sola cámara. Moeslund, Madsen y Granum 2005 aplican un enfoque de guiado secuencial de MonteCarlo para estimar un brazo. Un detector de partes provee la localización de la mano en la imagen. Esta información se utiliza para corregir la predicción, reduciendo el número de partículas necesarias.

Navaratnam y col. 2005 combina un modelo cinemático jerárquico con una detección de partes para recuperar la postura 3D del tronco superior. La detección de las partes permite localizar de forma independiente cada parte del cuerpo en cada cuadro. Las restricciones cinemáticas entre las partes del cuerpo se representan de forma jerárquica. A diferencia de los anteriores métodos de fusión de partes probabilística sin modelo, este permite recuperar la postura en cada cuadro. También se integra información temporal usando una HMM para reconstruir secuencias de movimiento temporalmente coherentes. Algunos trabajos que utilizan una cámara para la estimación de la pose son Shotton y col. 2011; Gall y col. 2010; Pishchulin, Jain y Andriluka 2012. La reconstrucción desde una cámara de movimientos complejos permanece como un problema abierto. Muchos trabajos explotan el uso de modelos de movimientos aprendidos para proporcionar fuertes restricciones y acotar el espacio de búsqueda.

A.1.6.3.3 Modelos de movimiento aprendidos

Cada vez hay más interés en utilizar modelos aprendidos de posturas y movimientos para restringir la reconstrucción de movimientos de personas usando una o múltiples cámaras. La posibilidad de capturar movimientos humanos usando sistemas comerciales basados en marcadores ha permitido hacer uso de modelos aprendidos de movimientos humanos tanto para la generación de gráficos por ordenador como para la visión por ordenador.

Los modelos aprendidos se han desarrollado en la animación por ordenador para permitir generar movimientos naturales con restricciones especificadas por el usuario a partir de una base de datos de capturas de movimientos. Este uso está

relacionado con el problema de visión por ordenador ya que se desarrollan métodos para predecir y restringir la postura y el movimiento a estimar. La aportación de la cinemática inversa ayuda a estos modelos aprendidos a generar los gráficos. Ong e Hilton 2005 aplica un modelo aprendido de configuraciones de todo el cuerpo para restringir la postura dada la posición de un conjunto de nodos terminales para una secuencia de movimiento.

Sidenbladh, Black y Sigal 2002 combina un muestreo estocástico con un potente aprendizaje del caminar. Agarwal y Triggs 2004a usan un modelo aprendido de dinámica de segundo orden para realizar un seguimiento 2D de movimientos más generales de caminar y correr con transiciones y giros en una secuencia de una cámara.

Los trabajos expuestos a continuación usan los modelos aprendidos para resolver las ambigüedades dadas al usar una cámara. En Howe, Leventon y Freeman 2000 modelos aprendidos de una serie de secuencias de movimientos sirven para deducir la postura 3D a partir del seguimiento de unos puntos de interés en un movimiento sencillo. Sigal y col. 2004 combina un detector de partes del cuerpo con un modelo aprendido de movimientos para deducir la postura 3D de una secuencia de caminar con inicialización automática. Urtasun y Fua 2004b introduce el uso de modelos aprendidos temporales a partir de información capturada para reconstruir el movimiento humano usando optimización basada en la pendiente del gradiente. Lleva a cabo un "análisis de componentes principales" (ACP) en los ángulos de las articulaciones en varias capturas de caminar y correr para proporcionar parámetros de baja dimensión. El modelo aprendido parametrizado se utiliza para restringir el movimiento de un modelo 3D en movimientos de andar y caminar con velocidad variable desde un par estéreo y movimientos de golf desde una cámara. Urtasun y Fua 2004b presentan una solución alternativa para representar el movimiento humano, "Modelo Variable Latente Proceso Gaussiano Escalado" (SGPLVM), para aprender un espacio de estados de posturas de baja dimensión para movimientos específicos. Utiliza SGPLVM para reconstruir tanto movimientos de caminar y de golf en secuencias captadas desde una cámara. En Agarwal y Triggs 2006c se aborda el uso de modelos de movimiento aprendidos sin dependencia del punto de vista.

Las investigaciones demuestran que el uso de modelos aprendidos facilita la reconstrucción del movimiento obtenidos de una sola cámara. Estas soluciones están limitadas a movimientos específicos con pocas variaciones y transiciones fijadas. El reto de estas investigaciones está en construir modelos más generales o métodos que permitan la transición entre modelos, para así permitir reconstruir movimientos sin restricciones.

A.1.7 Reconocimiento

El campo de representar las acciones y actividades, y reconocerlas es relativamente antiguo, aun así es un campo actualmente de podemos encontrar más trabajos desarrollados. Esta área es sujeto de muchas investigaciones que se ve reflejado en el gran número de trabajos publicados. Por otra parte, las soluciones son fuertemente dependientes de los fines que persigue. Su aplicación final abarca la vigilancia, estudios médicos y de rehabilitación, robótica, indexación de video y animaciones para películas y videojuegos. Por ejemplo, para interpretar un entorno el conocimiento se suele representar de forma estadística y se cataloga como actividad normal o irregular.

La representación debería ser independiente del objeto que realiza la actividad, de esta forma no debería distinguirse explícitamente. Por otra parte, algunas aplicaciones de videovigilancia se centran explícitamente en actividades de personas y la interacción entre ellas. En este caso, se busca soluciones holísticas que tienen en cuenta la persona sin tener en cuenta sus partes y soluciones locales. La mayoría de las soluciones holísticas tratan de identificar información como el género, identidad o acciones simples como caminar o correr. Las soluciones locales se interesan en acciones más sutiles o intentan modelar acciones buscando primitivas de acción con las que se construyen acciones más complejas. Los trabajos de reconocimiento los catalogamos en función del nivel de abstracción que alcanzan: interpretación de la escena, reconocimiento holístico, reconocimiento de partes, primitivas de acción y gramáticas. en Samitha, Mehrtash y Fatih 2017 se realiza un estudio de la taxonomia para el reconocimiento de acciones y actividades.

A.1.7.1 Interpretación de la escena

Muchos trabajos tratan de aprender y reconocer actividades simplemente observando el movimiento de los objetos sin conocer necesariamente su identidad. Esto es un planteamiento razonable si se consideran los objetos como un punto en el espacio.

En Stauffer y Grimson 2000 se presenta un sistema que interpreta una escena al completo, que permite detectar situaciones anormales. En Chu, Jenkins y Mataric 2003 se presenta un sistema de videovigilancia de una piscina. De cada objeto se extrae información como la velocidad, postura, índice de inmersión, un índice de actividad y un índice de salpicones. Estas características alimentan una red multivariable para detectar eventos de crisis.

A.1.7.2 Reconocimiento holístico

En muchas publicaciones se discute el reconocimiento de la identidad de una persona, basándose en su estructura y dinámica global del cuerpo. En concreto resulta de interés los andares de las personas. Otras soluciones usando información global de la estructura y su dinámica se centran en el reconocimiento a partir de acciones simples como andar o correr. Casi todos estos métodos se basan en el contorno. A continuación se revisan soluciones donde se trata de obtener la identidad de las personas.

A.1.7.2.1 Reconocimiento Holístico de la identidad

Wang y col. 2003b compara trayectorias de puntos del contorno primero aplicando un suavizado temporal para minimizar la distancia entre las observaciones y la galería de trayectorias. En Kale y col. 2002 se define un modelo de Markov para modelar la dinámica del caminar de cada persona. Yam, Nixon y Carter 2002 investiga la relación entre caminar y correr, para construir un único algoritmo que reconoce las personas tanto caminando como corriendo.

A.1.7.2.2 Reconocimiento Holístico de la acción

Mientras que un número de trabajos reconocen las personas basándose en su dinámica, también se puede usar la dinámica para saber qué acción está realizando. Un trabajo pionero ha sido Efros y col. 2003 donde tratan de reconocer acciones simples de personas en las cuales solo son 30 píxeles de alto y la calidad del video es baja. Para ello usan un conjunto de características que se basan en el flujo óptico difuminado. En Robertson y Reid 2005 tratan de entender acciones construyendo un sistema jerárquico que se basa en razonar con redes neuronales y modelos de Markov en el nivel superior. Y en el nivel inferior con información como posición y velocidad como descriptores de la acción.

Otro gran número de publicaciones trabajan en el volumen espacio-tiempo. Una de las principales aportaciones Rittscher, Blake y Roberts 2002 usa paletas espacio temporal XT a partir del volumen XYT donde los movimientos de las articulaciones pueden asociarse con patrones típicos de trayectoria.

Bobick 1997 introdujeron el concepto de plantillas temporales. Proponen una representación que se basa en “imágenes de energía del movimiento” (MEI) e imágenes de historia del movimiento (MHI).

Yi, Rajan y Chia 2004 presenta la idea de un mapa de "relación de cambio de píxeles" (PCRM) que conceptualmente es similar al MHI. Sin embargo, posteriores procesamientos se basan en histogramas de movimiento que se calculan a partir del PCRM.

Ozer y Wolf 2002 abordan el seguimiento, la estimación de la postura y el reconocimiento de una forma integrada. Aplicando un número de técnicas bien conocidas. Otra solución es conocida como "esbozos de acciones" (Action Sketches) o "formas espacio-temporales" (Space-Time Shapes) Gorelick y col. 2003; Yilmaz y Shah 2005; Blank y col. 2005.

Otros autores en lugar de usar volúmenes espacio-temporales optan por abordarlo de una forma clásica considerando una secuencia de siluetas Yu y col. 2005. En Sato y Aggarwal 2001 detecta la interacción entre dos personas.

Cheng, Christmas y Kittler 2002 distingue entre caminar y correr basándose en información de videos deportivos. Gao, Hauptmann y Wactlar 2004 presenta una aplicación para una habitación inteligente. Se lleva a cabo un análisis de un comedor combinando segmentación con seguimiento.

Otros trabajos se fundamentan en modelos de Markov (HMM) Elgammal y col. 2003; Luo, Wu y Hwang 2003; Leo y col. 2004.

A.1.7.3 Reconocimiento basado en partes

Muchos autores tratan el reconocimiento de acciones basándose en la dinámica y configuración de las partes individuales del cuerpo. Wang y col. 2003a basándose en Wang y col. 2003b presenta un trabajo donde se extraen los contornos y mediante un filtro de partículas sigue la postura. Para discernir usa un clasificador de vecino más próximo. En Davis y Taylor 2002 se presenta una solución para distinguir entre caminar y no caminar, basándose en el sistema W4 Haritaoglu, Harwood y Davis 2000.

En Parameswaran y Chellappa 2003, Parameswaran y Chellappa considera el problema de reconocimiento de acciones invariante al punto de vista basándose en visualizar desde el punto de luz.

Fanti, Zelnik-Manor y Perona 2005 recoge la estructura de una persona como un modelo de conocimiento. Para encontrar la correspondencia del modelo más semejante con los datos de entrada explotan la información de la apariencia que permanece más o menos invariante dentro de una misma configuración. Se usa una maximización de las expectativas para el aprendizaje, sin supervisión de los

parámetros y la estructura del modelo para una acción en particular y datos de entrada sin etiquetar. Finalmente, la acción se reconoce como estimación de máxima similitud. Davis y Gao 2003 reconoce el género de las personas basándose en los andares.

A.1.7.4 Primitivas de acción y gramáticas

Existe una evidencia neurofisiológica que las acciones y actividades humanas están directamente conectadas al control del cuerpo humano. Al observar otras personas realizando acciones, el sistema visual humano parece ser que relaciona la entrada visual con una secuencia de primitivas. En el aprendizaje por imitación, el objetivo es desarrollar un sistema robótico que es capaz de relacionar las acciones percibidas con su propio sistema de movimiento para aprender y posteriormente reconocer y realizar las mismas acciones. La investigación actual se centra en la representación de las acciones percibidas visualmente y el control del sistema motor por imitación. Esto lleva a la idea de interpretar y reconocer actividades en una escena a través de una jerarquía de primitivas, acciones simples y actividades.

En Jenkins y Mataric 2002 se propone aplicar una técnica de reducción de la dimensión espacio temporal de forma no lineal sobre una secuencia de movimiento segmentada manualmente. En Calinon, Guenter y Billard 2005 se presenta una solución basada en HMM para aprender características de movimientos repetitivos. Sugieren el uso de HMM para sintetizar las trayectorias de las articulaciones de un robot, para cada articulación se usa un HMM.

Otros trabajos tratan de descomponer acciones en primitivas de acción e interpretar acciones como una composición de un alfabeto de estas primitivas de acción. En Vecchio, Murray y Perona 2003 usan técnicas de un sistema dinámico para logar la segmentación y clasificación. En Lu y Ferrier 2004 también abordan el problema desde un punto de vista teórico. Su objetivo es segmentar y representar movimientos repetitivos.

Rao, Yilmaz y Shah 2002 proponen una representación invariante de las acciones basándose en intervalos e instantes dinámicos.

En Gonzalez y col. 2002 se usa un modelo de distribución de puntos para modelar la variabilidad de la configuración de los ángulos de las articulaciones de un modelo humano sencillo. Un “espacio de acciones” (aSpace) se entrena con un conjunto de dichas configuraciones de diferentes individuos realizando la misma acción. aSpaces se usan posteriormente para sintetizar y reconocer acciones conocidas.

Yu y Yang 2005 utilizan redes neuronales para encontrar primitivas. Aplican "mapas auto-organizados" (SOM) que agrupan las imágenes de aprendizaje basándose en información de la forma. Después del aprendizaje SOM genera una etiqueta para cada imagen de entrada que convierte una secuencia de imágenes en una secuencia de etiquetas.

Apéndice B

Anexo: Modelo DPM

En este anexo se describe un "modelo de partes deformables" (DPM), multiescalado y entrenado discriminatoriamente para la detección de objetos.

B.1 Introducción

El modelo original DPM utilizando imágenes en RGB esta basado en Felzenszwalb, McAllester y Ramanan 2008; Felzenszwalb y col. 2010; Yang y Ramanan 2013.

Se considera el problema de detectar y localizar objetos de una categoría genérica, como personas, en imágenes estáticas. Se ha desarrollado un nuevo modelo de partes deformables de varias escalas para resolver este problema. Los modelos son entrenados usando un procedimiento discriminativo que sólo requiere etiquetas de cajas delimitadoras para los ejemplos positivos. Usando estos modelos, se implementa un sistema de detección que es altamente eficiente y preciso, procesando una imagen en pocos segundos y logrando tasas de reconocimiento altas.

El modelo de partes deformables utiliza un modelo de árbol tal y como muestra la figura B.1 donde cada una de las partes están relacionadas entre sí. Como por ejemplo la colocación de las muñecas depende de la colocación de los hombros, y estos a su vez de la colocación del tronco dentro de una imagen.

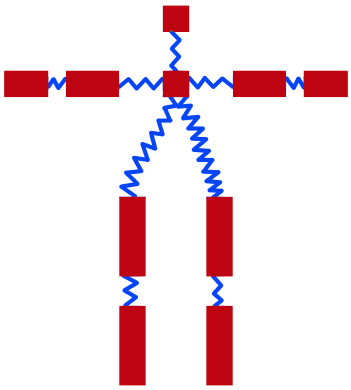


Figura B.1: Relación entra cada una de las partes.

Este sistema logrará una mejora de dos veces en la precisión media sobre el sistema ganador de Dalal y Triggs 2005c, en el desafío del 2006 de PASCAL de detección de personas. El sistema también superará los mejores resultados en el reto de 2007 en diez de las veinte categorías de objetos. La Figura B.2 muestra un ejemplo de detección obtenido con el modelo de persona propuesto.

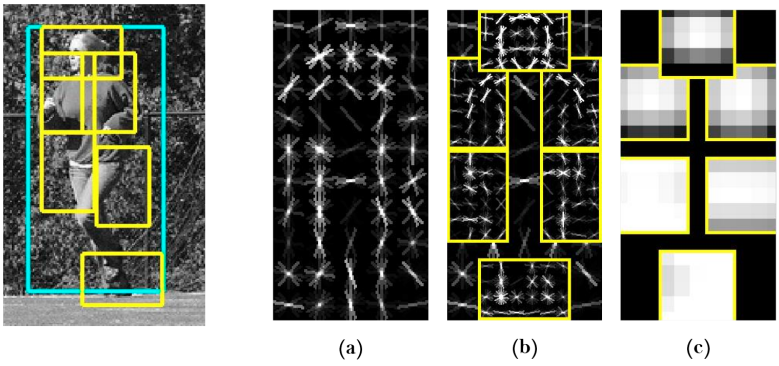


Figura B.2: Ejemplo de detección obtenido con el modelo de persona. El modelo se define mediante una plantilla gruesa (a), varias plantillas de partes de mayor resolución (b) y un modelo espacial para la ubicación de cada parte (c).

La noción de que los objetos pueden ser modelados por partes en una configuración deformable proporciona un marco elegante para representar las categorías

de objetos Amit y Trouve 2007; Burl, Weber y Perona 1998; Crandall, Felzenszwalb y Huttenlocher 2005; Epshtein y Ullman 2007b; Felzenszwalb y Huttenlocher 2005b; Fergus, Perona y Zisserman 2003; Fischler y Elschlager 1973; Ioffe y Forsyth 2001; Jin y Geman 2006; Schneiderman y Kanade 2004. Si bien estos modelos son atractivos desde un punto de vista conceptual, fue difícil establecer su valor en la práctica. En los conjuntos de datos difíciles, los modelos deformables fueron a menudo superados por modelos "más débiles" conceptualmente como plantillas rígidas, Dalal y Triggs 2005c, o bolsa de características (bag-of-features), Zhang y col. 2007.

Los modelos incluyen una plantilla global gruesa que cubre un objeto entero y plantillas de partes de mayor resolución. Las plantillas representan el histograma de las características de gradiente, Dalal y Triggs 2005c. Como en Holub y Perona 2005; Quattoni y col. 2007; Ramanan y Sminchisescu 2006, se entrenan a los modelos discriminativamente. Sin embargo, el sistema propuesto es semi-supervisado, entrenado con un marco "maxmargin", y no se basa en la detección de características. También se describe una estrategia simple y eficaz para aprender partes de datos débilmente etiquetados.

Para la formación discriminativa, se generalizan SVMs para manejar variables latentes tales como posiciones de parte, y se utiliza un método para la extracción de datos de ejemplos negativos durante el entrenamiento. La posición de cada parte del objeto es tratada como una variable latente. También, la ubicación exacta del objeto es tratada como una variable latente, requiriendo sólo que el clasificador seleccione una ventana que tenga una superposición grande con el recuadro delimitado previamente etiquetado.

Un SVM latente, como un CRF oculto Quattoni y col. 2007, conduce a un problema de entrenamiento no convexo. Sin embargo, a diferencia de un CRF oculto, un SVM latente es semi-convexo y el problema de entrenamiento se vuelve convexo una vez que se especifica información latente para los ejemplos de entrenamiento positivos. Esto conduce a un algoritmo general de coordenadas descendentes (Stochastic Gradient Descent - SGD) para SVMs latentes.

B.2 Modelo

Los bloques de construcción subyacentes para los modelos son el histograma de características de gradientes orientados (HOG) Dalal y Triggs 2005c. Se representan las características HOG en dos escalas diferentes. Las características gruesas son capturadas por una plantilla rígida que cubre toda una ventana de detección. Las características de escala más fina se capturan mediante plantillas de partes

que se pueden mover con respecto a la ventana de detección. El modelo espacial para las ubicaciones de las partes es equivalente a un grafo en estrella Crandall, Felzenszwalb y Huttenlocher 2005 donde la plantilla gruesa sirve como posición de referencia.

EL sistema utiliza un enfoque de ventana de exploración. Un modelo para un objeto consiste en un filtro raíz global, figura B.2 imagen (a), y varios modelos de partes. Cada modelo de partes especifica un modelo espacial y un filtro de partes. El modelo espacial define un conjunto de ubicaciones permitidas para una parte relativa a una ventana de detección, figura B.2 imagen (b), y un coste de deformación para cada ubicación, figura B.2 imagen (c).

B.2.1 Representación de HOG

La construcción en Dalal y Triggs 2005c se sigue para definir una representación densa de una imagen en una resolución en particular. La imagen se divide primero en regiones de $8x8$ píxeles que no se superponen. Para cada región se acumula un histograma $1D$ de las orientaciones de los gradientes sobre los píxeles de esa región. Estos histogramas capturan propiedades de forma locales pero también son invariantes a pequeñas deformaciones.

El gradiente en cada píxel es discretizado en una de las nueve celdas de orientación, y cada píxel vota para la orientación de su gradiente, con una fuerza que depende de la magnitud del gradiente. Para las imágenes en color, se calcula el gradiente de cada canal de color y se selecciona el canal con la magnitud de gradiente más alta en cada píxel. Finalmente, el histograma de cada región se normaliza con respecto a la energía del gradiente en un vecindario alrededor de ella. Se observan los cuatro bloques $2x2$ que contienen una región en particular y se normaliza el histograma de la región dada con respecto a la energía total en cada uno de estos bloques. Esto conduce a un vector de longitud $9x4$ que representa la información de gradiente local dentro de una celda.

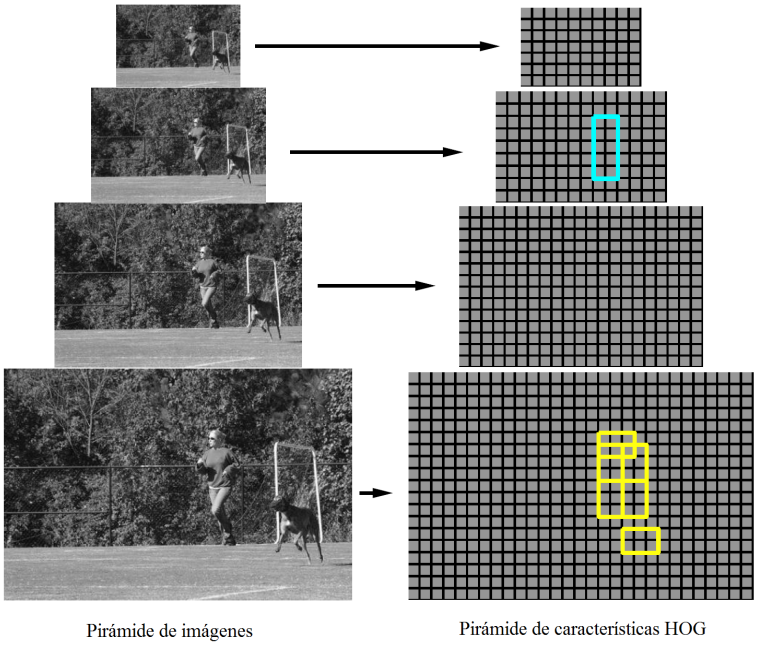


Figura B.3: La pirámide multiescala de características HOG y una hipótesis de objeto definida en términos de una colocación del filtro raíz (cerca de la parte superior de la pirámide) y los filtros de parte (dos niveles por debajo).

Se define una pirámide de características HOG mediante el cálculo de las características HOG de cada nivel de una pirámide de imágenes estándar (ver Figura B.3). Las características en la parte superior de esta pirámide captan gradientes gruesos histografiados sobre áreas bastante grandes de la imagen de entrada, mientras que las características en la parte inferior de la pirámide capturan gradientes más finos histografiados sobre áreas pequeñas.

B.2.2 Filtros

Los filtros son plantillas rectangulares que especifican pesos para las subventanas de una pirámide HOG.

Un filtro es una plantilla rectangular definida por una matriz de vectores d-dimensionales de pesos. La respuesta, o coste, de un filtro F en una posición (x, y) en un mapa de características G es el "producto escalar" del filtro y una subventana del mapa de características con la esquina superior izquierda en (x, y):

$$\sum_{x',y'} F[x',y'] \cdot G[x+x',y+y'] \tag{B.1}$$

Un filtro F de dimensiones w por h es un vector con pesos $w \times h \times 9 \times 4$. El coste de un filtro se define tomando el producto escalar del vector de pesos y las características en una subventana $w \times h$ de una pirámide HOG.

El sistema de Dalal y Triggs 2005c utiliza un solo filtro para definir un modelo de objetos. Ese sistema detecta objetos de una clase particular puntuando cada $w \times h$ subventana de una pirámide HOG y limando los costes.

Sea H una pirámide HOG y $p = (x, y, l)$ sea una región en el nivel l-ésimo de la pirámide. (H, p, w, h) denotan el vector obtenido concatenando las características HOG en la subventana $w \times h$ de H con la esquina superior izquierda en p. El coste de F en esta ventana de detección es $F \cdot \phi(H, p, w, h)$. Si organizamos el vector concatenando las características HOG en la subventana $w \times h$ de H con la esquina superior izquierda en p según el orden de la fila, el coste de F en esta ventana de detección la denotaremos como $F' \cdot \phi(H, p, w, h)$, donde F' es el vector obtenido concatenando los vectores de peso en F en según el orden de la fila.

A continuación, se utiliza $\phi(H, p)$ para denotar (H, p, w, h) cuando las dimensiones se pueden extraer del contexto.

La figura B.4 muestra de manera esquemática los pasos utilizados para calcular la función de coste combinada con las localizaciones raíz.

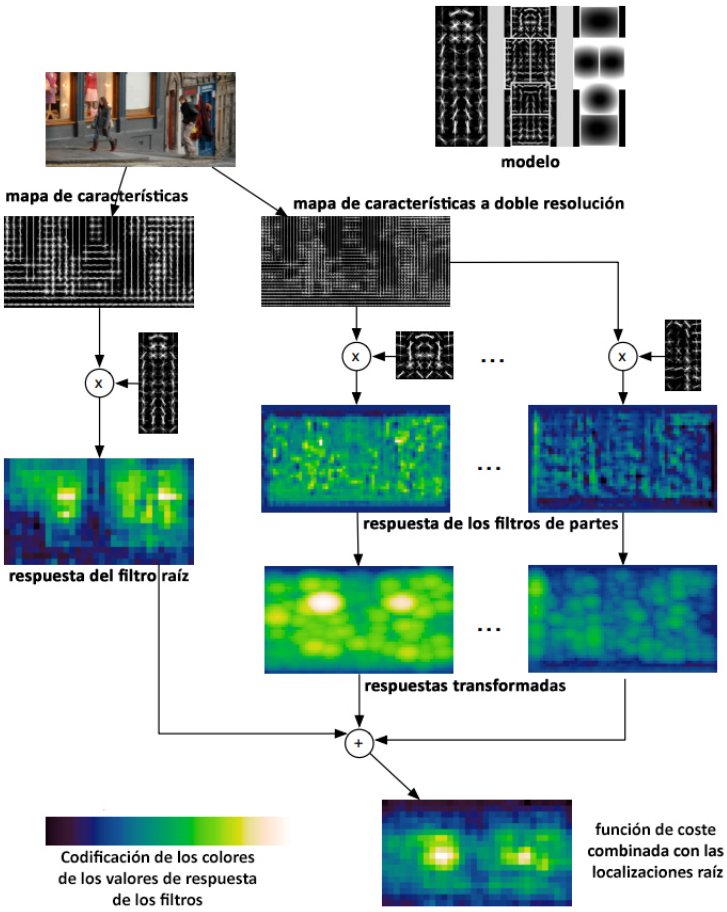


Figura B.4: Representación gráfica para el cálculo de la función de coste.

B.2.3 Partes deformables

Aquí se consideran los modelos definidos por un filtro raíz grueso que cubre todo el objeto y filtros de partes de mayor resolución que cubren partes más pequeñas del objeto. La figura B.3 ilustra una colocación de este modelo en una pirámide HOG. La ubicación del filtro raíz define la ventana de detección (los píxeles dentro de la región cubierta por el filtro). Los filtros de partes se colocan dos niveles por arriba en la pirámide, por lo que las regiones HOG a ese nivel tienen el doble del tamaño de las regiones en el nivel de filtro raíz.

El uso de características de mayor resolución para definir filtros de partes es esencial para obtener un alto rendimiento de reconocimiento. Con este enfoque, los filtros de partes representan bordes de resolución más finos que se localizan con mayor precisión cuando se comparan con los bordes representados en el filtro raíz. Por ejemplo, considere la posibilidad de construir un modelo para una cara. El filtro raíz podría capturar bordes de resolución gruesos, como el límite de la cara, mientras que los filtros de partes podrían capturar detalles tales como ojos, nariz y boca.

El modelo de un objeto con n partes es formalmente definido por un filtro raíz F_0 y un conjunto de modelos de partes $(P_1, \ldots, P_n)$ donde $P_i = (F_i, v_i, s_i, a_i, b_i)$. Aquí F_i es un filtro para la i-ésima parte, v_i es un vector bidimensional que especifica el centro de una caja de posibles posiciones para la parte i en relación a la posición raíz, s_i da el tamaño de esta caja, mientras que a_i y b_i son vectores bidimensionales que especifican coeficientes de una función cuadrática midiendo una puntuación para cada colocación posible de la i-ésima parte. La figura B.2 ilustra un modelo de persona.

La colocación de un modelo en una pirámide HOG viene dada por $z = (p_0, \ldots, p_n)$, donde $p_i = (x_i, y_i, l_i)$ es la localización del filtro raíz cuando $i = 0$ y la localización de la $i - th$ parte cuando $i > 0$. Se supone que el nivel de cada parte es tal que una región HOG en ese nivel tiene la mitad del tamaño de una región HOG en el nivel de raíz. El coste de una ventana de detección es el coste del filtro raíz en la ventana más la suma de las partes, de los máximos de las ubicaciones de cada una de las partes, del coste del filtro de partes en la subventana resultante menos el coste de deformación. Esto es similar a los modelos clásicos basados en partes Felzenszwalb y Huttenlocher 2005b; Fischler y Elschlager 1973. Descrito de otra forma, el coste de una ubicación viene dado por el coste de cada filtro (el término de apariencia) más un coste de la ubicación de cada parte relativa a la raíz (el término espacial),

$$\sum_{i=0}^{n} F_i \cdot \phi(H, p_i) + \sum_{i=1}^{n} a_i \cdot (\tilde{x}_i, \tilde{y}_i) + b_i \cdot (\tilde{x}_i^2, \tilde{y}_i^2) \tag{B.2}$$

donde $(\tilde{x}_i, \tilde{y}_i) = ((x_i, y_i) - 2(x, y) + v_i)/s_i$ da la ubicación de la i-ésima parte relativa a la ubicación de la raíz. Ambos $\tilde{x}_i$ y $\tilde{y}_i$ están acotados entre -1 y 1.

Posteriormente a este estudio, en Felzenszwalb y col. 2010, la ecuación anterior, ecuación B.2, es re-escrita para aumentar la precisión. Para ello un modelo para un objeto con n partes es formalmente definido por una vector de dimensión $(n+2)$, $(F_0, P_1, \ldots, P_n, b)$, donde F_0 es un filtro raíz, P_i es un modelo para la i-ésima

parte, b es un término de sesgo de valor real. Cada modelo de pieza se define por un vector de dimensión 3, (F_i, v_i, d_i), donde F_i es un filtro para la i-ésima parte, v_i es un vector bidimensional que especifica una posición de anclaje para la parte i relativa a la posición raíz y d_i es un vector de dimensión 4 que especifica coeficientes de una función cuadrática que define una deformación de coste para cada colocación posible de la parte con relación a la posición de anclaje. Con lo que la ahora la función de coste de una hipótesis viene dada por el coste de cada filtro en sus respectivas ubicaciones (el término de apariencia) menos un coste de deformación que depende de la posición relativa de cada parte con respecto a la raíz (el término espacial), más el sesgo, la ecuación B.2, se re-escribe como:

$$score(p_0, \ldots, p_n) = \sum_{i=0}^{n} F_i^{'} \cdot \phi(H, p_i) - \sum_{i=1}^{n} d_i \cdot \phi_d(dx_i, dy_i) + b \tag{B.3}$$

donde $(dx_i, dy_i) = (x_i, y_i) - (2(x_0, y_0) + v_i)$ da el desplazamiento de la i-ésima parte con respecto a su posición de anclaje y $\phi_d(dx_i, dy_i) = (dx, dy, dx^2, dy^2)$ son características de deformación.

Existe un gran número (exponencial) de ubicaciones para un modelo en una pirámide HOG. Se utiliza la programación dinámica y las técnicas de transformación de la distancia Felzenszwalb y Huttenlocher 2004; Felzenszwalb y Huttenlocher 2005b para calcular la mejor ubicación para las partes de un modelo en función de la ubicación de la raíz. Esto tiene un coste temporal de $O(nk)$, donde n es el número de partes en el modelo y k es el número de regiones en la pirámide HOG. Para detectar objetos en una imagen puntuamos las ubicaciones de raíz según la mejor ubicación posible de las partes y el umbral de esta puntuación.

Los filtros de raíz y los filtros de parte se calculan realizando el producto escalar entre un conjunto de pesos y un histograma de características de gradiente (HOG) dentro de una ventana. El filtro raíz es equivalente al modelo de Dalal-Triggs Dalal y Triggs 2005c. Las características de los filtros de partes se calculan al doble de la resolución espacial del filtro raíz. El modelo se define a una escala fija, y se detectan los objetos buscando sobre una pirámide multiescala de la imagen

El coste de una ubicación z puede expresarse en términos del producto escalar, $\beta \cdot \psi(H, z)$, entre un vector de parámetros de modelo β y un vector $\psi(H, z)$,

$$\beta = (F_0, \ldots, F_n, a_1, b_1, \ldots, a_n, b_n) \tag{B.4}$$

$$\psi(H,z) = (\phi(H,p_0), \phi(H,p_1), \ldots, \phi(H,p_n), \\ \tilde{x}_1, \tilde{y}_1, \tilde{x}_1^2, \tilde{y}_1^2, \ldots, \tilde{x}_n, \tilde{y}_n, \tilde{x}_n^2, \tilde{y}_n^2) \tag{B.5}$$

donde si utilizamos el filtro ordenado F ordenado anteriormente definido como $F^{'}$, las ecuaciones B.4 y B.6 son reescritas como:

$$\beta = (F_0^{'}, \ldots, F_n^{'}, d_1, \ldots, d_n, b) \tag{B.6}$$

$$\psi(H,z) = (\phi(H,p_0), \phi(H,p_1), \ldots, \phi(H,p_n), \\ -\phi_d(dx_1, dy_1), \ldots, -\phi_d(dx_n, dy_n), 1) \tag{B.7}$$

Se utiliza esta representación para aprender los parámetros del modelo β con el algoritmo SVM latente, ya que así se relaciona los modelos deformables con los clasificadores lineales.

Un aspecto interesante de los modelos espaciales aquí definidos es que se permite que los coeficientes (a_i, b_i) sean negativos. Esto es más general que el coste cuadrático de "muelle" que se ha utilizado en trabajos anteriores.

B.3 Entrenamiento

En el entrenamiento se dispone de un conjunto de imágenes anotadas con cajas delimitadoras alrededor de cada instancia de un objeto. El problema de detección se reduce a un problema de clasificación binaria. Cada ejemplo x está marcado por una función de la forma, $f_\beta(x) = max_z \beta \cdot \Phi(x,z)$. Aquí β es un vector de parámetros del modelo y z son los valores latentes (por ejemplo, las ubicaciones de parte). Con estos datos se reduce el problema de aprender un modelo de partes deformable a un problema de clasificación binaria. Sea $D = (\langle x_1, y_1 \rangle, ..., \langle x_n, y_n \rangle)$ un conjunto de ejemplos etiquetados donde $y_i \in \{-1, 1\}$ y x_i especifica una pirámide HOG, $H(x_i)$, junto con un rango, $Z(x_i)$, de ubicaciones válidas para los filtros raíz y los filtros de partes. Se construye un ejemplo positivo de cada caja delimitadora en el conjunto de entrenamiento. Para estos ejemplos se define $Z(x_i)$ para que el filtro raíz se coloque para superponer el cuadro delimitador en al menos el 50 %. Los ejemplos negativos provienen de imágenes que no contienen el objeto de interés. Cada colocación del filtro raíz en una imagen de este tipo da como resultado un ejemplo de entrenamiento negativo.

Obsérvese que para los ejemplos positivos se tratan tanto las ubicaciones de las partes como la ubicación exacta del filtro raíz como variables latentes. Se ha encontrado que permitir la incertidumbre en la localización de la raíz durante el entrenamiento mejora perceptiblemente el funcionamiento del sistema.

B.3.1 SVMs latente

Para aprender un modelo se define una generalización de SVMs que se llama "latent variable SVM" (LSVM). Una propiedad importante de LSVMs es que el problema de entrenamiento se vuelve convexo si se fija los valores latentes para los ejemplos positivos. Esto se puede usar en un algoritmo de descenso de coordenadas.

Una SVM latente se define como sigue. Se asume que cada ejemplo x es anotado por una función de la forma,

$$f_\beta(x) = \max_{z \in Z(x)} \beta \cdot \Phi(x, z) \tag{B.8}$$

Donde β es un vector de parámetros del modelo y z es un conjunto de valores latentes. Para los modelos deformables, se define $\Phi(x, z) = \psi(H(x), z)$ de manera que $\beta \cdot \psi(x, z)$ es la puntuación de colocar el modelo de acuerdo con z.

En la práctica se aplica iterativamente el entrenamiento SVM clásico a triplas $(\langle x_1, z_1, y_1\rangle, \ldots, \langle x_n, z_n, y_n\rangle)$ donde z_i se selecciona para ser el mejor marcador latente de x_i bajo el modelo aprendido en la iteración anterior. Se genera un filtro raíz inicial a partir de los cuadros delimitadores en el "dataset". Las partes se inicializan desde este filtro raíz.

En analogía a las SVM clásicas, se quiere entrenar a partir de los ejemplos marcados $D = (\langle x_1, y_1\rangle, ..., \langle x_n, y_n\rangle)$ optimizando la siguiente función objetivo,

$$\beta^*(D) = argmin_\beta \lambda ||\beta||^2 + \sum_{i=1}^{n} \max(0, 1 - y_i f_\beta(x_i)) \tag{B.9}$$

Al restringir los dominios latentes $Z(x_i)$ a una sola opción, f_β se convierte en lineal en β, y se obtienen SVM lineales como un caso especial de SVMs latentes. SVMs latentes son ejemplos de la clase general de los métodos basados en energía (energy-based methods) LeCun y col. 2006.

B.3.2 Semi-convexidad

Obsérvese que $f_\beta(x)$, como se define en la ecuación B.8, es un máximo de funciones lineales en β. Por tanto, $f_\beta(x)$ es convexa en β. Esto implica que la pérdida de bisagra (hinge loss) máx$(0, 1 - y_i f_\beta(x_i))$ es convexa en β cuando $y_i = -1$. Es decir, la función de pérdida es convexa para ejemplos negativos. A esta propiedad de la función de pérdida se le llama semi-convexidad.

Considere una LSVM donde los dominios latentes $Z(x_i)$ para los ejemplos positivos se limitan a una sola elección. La pérdida debida a cada ejemplo positivo es ahora convexa. Combinada con la propiedad de semi-convexidad, la ecuación B.9 se convierte en convexa en β.

Si las etiquetas de los ejemplos positivos no son fijas se puede calcular un óptimo local de la ecuación B.9 usando un algoritmo de descenso de coordenadas:

1. Siendo β fijo, optimizar los valores latentes para los ejemplos positivos $z_i = argmax_{z \in Z(x_i)} \beta \cdot \Phi(x, z)$.
2. Siendo $\{z_i\}$ fijo para los ejemplos positivos, optimizar β mediante la solución del problema convexo definido anteriormente.

Se puede demostrar que ambos pasos siempre mejoran o mantienen el valor de la función objetivo en la ecuación B.9. Si ambos pasos mantienen el valor se tiene un óptimo local fuerte de la ecuación B.9, en el sentido de que el Paso 1 busca sobre un espacio exponencialmente grande de etiquetas latentes para ejemplos positivos mientras que el Paso 2 busca simultáneamente sobre vectores de peso y un espacio exponencialmente grande de etiquetas latentes para ejemplos negativos.

B.3.3 Extracción de datos negativos fuertes (Data Mining Hard Negatives)

En la detección de objetos la gran mayoría de los ejemplos de entrenamiento son negativos. Esto hace que sea imposible considerar todos los ejemplos negativos a la vez. En cambio, es común construir datos de entrenamiento que consisten en instancias positivas y casos "negativos fuertes", donde los negativos fuertes son datos extraídos del conjunto muy grande de posibles ejemplos negativos.

Se describe un método general para los ejemplos de minería de datos para SVMs y SVMs latentes. El método resuelve iterativamente sub-problemas usando sólo instancias fuertes.

Los resultados descritos aquí se aplican tanto a las SVM clásicas como al problema definido en el paso 2 del algoritmo de coordenadas descendentes para las SVMs latentes.

Se definen las instancias fuertes de D en relación con β como:

$$M(\beta, D) = \{\langle x, y\rangle \in D | y f_\beta(x) \leq 1\} \tag{B.10}$$

Es decir, $M(\beta, D)$ son ejemplos de entrenamiento que están clasificados incorrectamente o cerca del margen del clasificador definido por β. Se puede demostrar que $\beta^*(D)$ sólo depende de instancias fuertes.

Apéndice C

Anexo: Filtro de partículas

En este anexo vamos a introducir el filtro de partículas en el que se ha basado la presente tesis. Este anexo es complementario al capítulo 4.

C.1 Introducción

El "filtro de partículas" (FP) es un método empleado para estimar el estado de un sistema que cambia a lo largo del tiempo. Más concretamente, es un método de "Montecarlo" (secuencial) usado comúnmente en visión artificial para el seguimiento de objetos en secuencias de imágenes.

Fue propuesto en 1993 por N. Gordon, D. Salmond y A. Smith como filtro "bootstrap" para implementar filtros bayesianos recursivos. Básicamente, el filtro de partículas se compone de un conjunto de muestras (las partículas) y unos valores, o pesos, asociados a cada una de esas muestras. Las partículas son estados posibles del proceso, que se pueden representar como puntos en el espacio de estados de dicho proceso.

Posee cuatro etapas principales:

- Inicialización.
- Actualización.
- Estimación

- Predicción.

Para realizar el seguimiento de un objeto sobre una secuencia de imágenes, el filtro de partículas lanza al azar un conjunto de puntos sobre la imagen (etapa de inicialización, se crea un conjunto de partículas con un estado aleatorio), se le asignará un valor, o valores, a cada uno de esos puntos realizando unos cálculos (etapa de actualización). A partir de estos valores, se creará un nuevo conjunto de puntos que reemplazará al anterior. Esta elección también será al azar, pero los valores que se han adjudicado a cada uno de los puntos provocarán que sea más probable de elegir aquellos puntos que hayan capturado al objeto sobre el que quiere realizar el seguimiento (etapa de estimación). Una vez que se crea el nuevo conjunto de puntos, se realiza una leve modificación al estado (posición) de cada uno de ellos, con el fin de estimar el estado del objeto en el instante siguiente (etapa de predicción).

Al terminar la etapa de predicción, se obtiene un nuevo conjunto de puntos al que se le vuelve a aplicar la etapa de actualización, repitiéndose este bucle hasta que termine la secuencia o desaparezca el objeto, caso en el cual se volvería a la etapa de inicialización.

A menudo son una alternativa al "filtro de Kalman extendido" (EKF) con la ventaja de que, con muestras suficientes, se acercan más a la estimación del bayesiano óptimo, por lo que la información puede ser más precisa que la aportada por EKF. Los enfoques pueden ser combinados utilizando una versión del "filtro de Kalman" (FK) como una distribución propuesta para el filtro de partículas.

C.1.1 Objetivo

El objetivo del filtro de partículas es estimar la secuencia de parámetros ocultos x_k para $k = 0, 1, 2, 3, \ldots$ basándose únicamente en los valores observados y_k para $k = 0, 1, 2, 3, \ldots$. Todas las estimaciones bayesianas de x_k se derivan de la distribución posterior $p(x_k|y_0, y_1, \ldots, y_n)$.

C.1.2 Modelo

Los métodos del filtro de partículas asumen que los estados x_k y las observaciones y_k pueden ser modeladas en la forma:

- $x_0, x_1, \ldots$ es un sistema de primer orden de "Markov" de tal forma que $x_k|x_{k-1} \sim p_{x_k|x_{k-1}}(x|x_{k-1})$ y con una distribución inicial de $p(x_0)$.

- Los observadores $y_0, y_1, \ldots$ son condicionalmente independientes, siempre que $x_0, x_1, \ldots$ sean conocidas. En otras palabras, cada y_k solo depende de $x_k : y_k|x_k \sim p_{y|x}(y|x_k)$.

Un ejemplo de lo mencionado es:

$$\begin{aligned} x_k &= f(x_{k-1}) + w_k \\ y_k &= h(x_k) + v_k \end{aligned} \tag{C.1}$$

donde tanto w_k como v_k son mutuamente independientes e idénticamente distribuidos con secuencias conocidas de funciones de densidad de probabilidad, y $f(\bullet)$ y $h(\bullet)$ son funciones conocidas. Estas dos ecuaciones se pueden ver como ecuaciones del espacio de estados y de un aspecto similar las ecuaciones del espacio de estados para el filtro de Kalman. Si las funciones $f(\bullet)$ y $h(\bullet)$ son lineales, y tanto w_k como v_k son gaussianos, el filtro de Kalman encuentra la distribución exacta del filtro Bayesiano. Si no, los métodos basados en el filtro de Kalman son una aproximación de primer orden (EKF) o una aproximación de segundo orden (UKF en general, pero si la distribución de probabilidad es gaussiana, es posible una aproximación de tercer orden). Los filtros de partículas son también una aproximación, pero con partículas que pueden ser mucho más precisas.

C.1.3 Aproximación de MonteCarlo

Los métodos de partículas, como todos los basados en métodos de muestreo (por ejemplo MCMC), generan un conjunto de muestras que se aproximan a la distribución de filtrado $p(x_k|y_0, \ldots, y_k)$. Así, con muestras de P, las expectativas con respecto a la distribución de filtrado se aproximan por:

$$\int f(d_x)p(x_k|y_0, \ldots, y_k)dx \approx \frac{1}{P}\sum_{L=1}^{P} f(x_k^{(L)}) \tag{C.2}$$

y $f(\bullet)$, de la forma habitual de "MonteCarlo", puede dar todos los momentos de la distribución, hasta un cierto grado de aproximación.

C.1.4 Sequencial importance resampling (SIR)

El algoritmo original del filtro de partículas Gordon, Salmond y Smith 1993, es uno de los filtros de partículas comúnmente utilizado, que se aproxima a la distribución de filtrado $p(x_k|y_0,\ldots,y_k)$ por un sistema ponderado de partículas P.

$$\{(w_k^{(L)}, x_k^{(L)}) : L \in \{1,\ldots,P\}\} \tag{C.3}$$

La importancia de pesos $w_k^{(L)}$ son aproximaciones a las probabilidades posteriores relativas (o de densidad) de las partículas de tal manera que:

$$\sum_{L=1}^{P} w_k^{(L)} = 1 \tag{C.4}$$

El método SIR es una versión secuencial (es decir, recursivo) del "Importance Sampling" (SIS). Como en "Importance Sampling", la expectativa de una función $f(\bullet)$ se puede aproximar como un promedio ponderado:

$$\int f(d_x)p(x_k|y_0,\ldots,y_k)dx \approx \frac{1}{P}\sum_{L=1}^{P} w^{(L)} f(x_k^{(L)}) \tag{C.5}$$

Para un conjunto finito de partículas, el rendimiento de los algoritmos depende de la elección de la distribución propuesta:

$$\pi(x_k|x_{0_k-1}, y_{0:k}) \tag{C.6}$$

La distribución óptima propuesta se da como la distribución de destino:

$$\pi(x_k|x_{0_k-1}, y_{0:k}) = p(x_k|x_{k-1}, y_k) \tag{C.7}$$

Los filtros SIR con prioridad de transición como función de importancia son comúnmente conocidas como "bootstrap filter" y "condensation algorithm".

El re-muestreo (Resampling) se utiliza para evitar el problema de la degeneración del algoritmo, es decir, evitar la situación en que todos los pesos de importancia menos uno se aproximan a cero. El desempeño del algoritmo puede ser afectado también por la adecuada elección de un método de re-muestreo. El "stratfield sampling" propuesto por Kitagawa en 1996 es óptimo en términos de la varianza.

Unos pasos sencillos del método SIR son los siguientes:

1. Para $L = 1, \ldots, P$ tomar muestras de la distribución propuesta:

$$x_k^{(L)} \sim \pi(x_k | x_{0_k-1}^{(L)}, y_{0:k}) = p(x_k | x_{k-1}) \tag{C.8}$$

2. Para $L = 1, \ldots, P$ actualización de las ponderaciones de importancia normalizadas:

$$\hat{w}_k^{(L)} = \hat{w}_{k-1}^{(L)} \frac{p(y_k | x_k^{(L)}) p(x_k^{(L)} | x_{k-1}^{(L)})}{\pi(x_k^{(L)} | x_{0:k-1}^{(L)}, y_{0:k})} \tag{C.9}$$

   Se tiene en cuenta que esta expresión simplifica la siguiente:

$$\hat{w}_k^{(L)} = \hat{w}_{k-1}^{(L)} p(y_k | x_{k-1}^{(L)}) \tag{C.10}$$

   Cuando:

$$\pi(x_k^{(L)} | w_{0:k-1}^{(L)}, y_{0:k}) = p(x_k^{(L)} | x_{k-1}^{(L)}) \tag{C.11}$$

3. Para $L = 1, \ldots, P$ se calcula los pesos de importancia normalizada:

$$w_k^{(L)} = \frac{\hat{w}_k^{(L)}}{\sum_{L=1}^{P} (w_k^{(L)})^2} \tag{C.12}$$

4. Se calcula una estimación del número efectivo de partículas:

$$\hat{N}_{eff} = \frac{1}{\sum_{L=1}^{P} (w_k^{(L)})^2} \tag{C.13}$$

5. Si el número efectivo de partículas es menor que un determinado umbral $\hat{N}_{eff} < N_{thr}$, se realiza el re-muestreo:

   - Ahora se dibujan las partículas P del conjunto actual de partículas con probabilidades proporcionales a sus pesos. Se reemplaza el conjunto de partículas actual por esta nueva.
   - Para $L = 1, \ldots, P$ se tiene $w_k^{(L)} = 1/P$.

El término "Sequencial Importance Resampling" es también utilizado cuando se hace referencia a los filtros SIS.

C.1.5 Sequencial importance sampling (SIS)

Es el mismo método que el "Sequential Importance Resampling" pero sin el paso del re-muestreo.

C.1.6 Versión directa del algoritmo

La versión directa del algoritmo es bastante simple en comparación con otros algoritmos de filtrado de partículas, y utiliza la composición y el rechazo. Para generar una simple muestra x en k de $p_{x_k|x_{1:k}}(x|y_{1:k})$:

1. Se establece $p = 1$.
2. Uniformemente generar L de $\{1, \ldots, P\}$
3. Se genera una prueba $\hat{x}$ para la distribución $p_{x_k|x_{k-1}}(x|x^{(L)}_{k-1|k-1})$.
4. Se genera la probabilidad de $\hat{y}$ utilizando $\hat{x}$ de $p_{y|x}(y_k|\hat{x})$ donde y_k es el valor mesurado.
5. Se genera otro uniforme u de $[0, m_k]$.
6. Se compara u y $p(\hat{y})$.
   - Si u es más grande repetimos desde el paso 2.
   - Si u es más pequeño entonces se guarda $\hat{x}$ como $x^{(p)}_{K|k}$ y se incrementa p.
7. Si $p > P$ entonces se termina.

El objetivo es generar P partículas en k utilizando sólo las partículas de $k-1$. Esto requiere que una ecuación de "Markov" se pueda escribir para generar un x_k basado solamente en x_{k-1}. Este algoritmo utiliza la composición de partículas P desde $k-1$ para generar una partícula en k y repite los pasos del 2 al 6 hasta que las partículas P se generen en k.

Esto puede ser más fácil de visualizar si x es visto como una matriz de dos dimensiones. Una dimensión es k y las otras dimensiones son el número de partículas.

Por ejemplo, $x(k, L)$ sería la L^{th} en k y puede ser escrita como $x_k^{(L)}$. El paso 3 genera un potencial x_k basado en unas partículas al azar $x_{k-1}^{(L)}$ en el instante $k-1$ y es aceptado o rechazado en el paso 6. En otras palabras, los valores x_k son generados utilizando los valores previamente generados x_{k-1}.

Apéndice D

Anexo: Cálculos previos para las proyecciones de las esferas en 2D

Se detallan los cálculos previos necesarios que se tienen que utilizar para dar solución a la Restricción 3 del filtro de partículas visto en la capítulo 4, sección 4.4.

D.1 Introducción

Los siguientes cálculos nos ayudarán a realizar las operaciones necesarias para dar solución a la restricción 3 en la sección 4.4.

D.1.1 Cálculos previos

Una elipse viene definida por: $c_1 = [x_1, y_1]$, eje menor r_a, eje mayor r_b, ángulo de rotación α. La forma de unir esta información en forma de vector es $el_1 = [x_1, y_1, r_a, r_b, \alpha]$.

La ecuación paramétrica de una elipse es:

$$eq_1 : ax^2 + 2bxy + cy^2 + 2dx + 2ey + f = 0 \tag{D.1}$$

donde x, y son las coordenadas de un punto de la elipse, y los valores a, b, c, d, e, f son los parámetros de la ecuación que dependen de el_1. La forma de calcular estos parámetros a partir de el_1 es:

$$\begin{array}{lll} c = cos(\alpha) & s = sin(\alpha) & P = \left(\frac{c}{a}\right)^2 + \left(\frac{s}{b}\right)^2 \\ a = r_a & b = r_b & Q = \left(\frac{s}{a}\right)^2 + \left(\frac{c}{b}\right)^2 \\ x_c = x_1 & y_c = y_1 & R = 2cs\left(\frac{1}{a^2} - \frac{1}{b^2}\right) \end{array} \tag{D.2}$$

Y realizando los cálculos oportunos:

$$ParA = \left[P \quad \frac{R}{2} \quad Q \quad \frac{-2Px_c - Ry_c}{2} \quad \frac{-2Qy_c - Rx_c}{2} \quad Px_c^2 + Qy_c^2 + Rx_cy_c - 1\right] \tag{D.3}$$

Donde:

$$ParA = [a \quad b \quad c \quad d \quad f] \tag{D.4}$$

D.1.2 Intersección entre dos rectas en el mismo plano

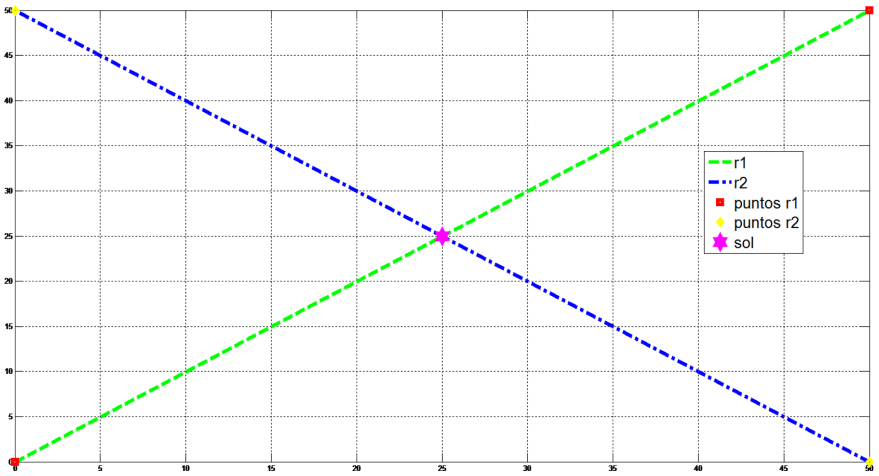


Figura D.1: Intersección entre rectas.

Cada recta r_1 y r_2 viene definida por su punto inicial y final, $\{p_1, p_2 | p_i = [x_i, y_i]\}/p_1, p_2 \in r_1$ y $\{p_3, p_4 | p_i = [x_i, y_i]\}/p_3, p_4 \in r_2$ donde $p_i \in \Re^2$. Las pendientes de dichas rectas vienen dadas por $m_1 = (y_2 - y_1)/(x_2 - x_1)$ y $m_2 = (y_4 - y_3)/(x_4 - x_3)$. Las

ecuaciones de las rectas son $r_1 : (y-y_1) = m_1(x-x_1)$ y $r_2 : (y-y_3) = m_2(x-x_3)$. Para calcular la intersección entre ambas rectas tenemos que resolver el siguiente sistema de ecuaciones:

$$\begin{cases}(y-y_1) = m_1(x-x_1)\\(y-y_3) = m_2(x-x_3)\end{cases} \Rightarrow \begin{cases}(y-y_1)(x_2-x_1) = (y_2-y_1)(x-x_1)\\(y-y_3)(x_4-x_3) = (y_4-y_3)(x-x_3)\end{cases} \quad \text{(D.5)}$$

donde $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4$ son parámetros conocidos y x, y son la solución al sistema de ecuaciones $sol = [x, y]$. Al resolver el sistema de coordenadas nos podemos encontrar con tres posibles casos.

D.1.2.1 Caso 1: Las dos rectas son paralelas

Si $r_1 || r_2$ el sistema de ecuaciones no tiene solución. Con lo cual la solución aportada es $sol = \{\emptyset\}$, $|sol| = 0$.

D.1.2.2 Caso 2: Las dos rectas son iguales

Si $r_1 = r_2$ la solución al sistema de ecuaciones tiene infinitas soluciones, $|sol| = \infty$. En nuestro problema, nos servirá en este caso coger como solución los extremos de las dos rectas, con lo cual se tienen 4 puntos en la solución, $|sol| = 4/sol = \{p_1, p_2, p_3, p_4\}$.

D.1.2.3 Caso 3: Las dos rectas son secantes entre ellas y se cortan en un punto

Si $r_1 \angle r_2$ la solución será única, $|sol| = 1/sol = \{p | p = [x, y]\}$.

D.1.3 Intersección entre dos elipses en el mismo plano

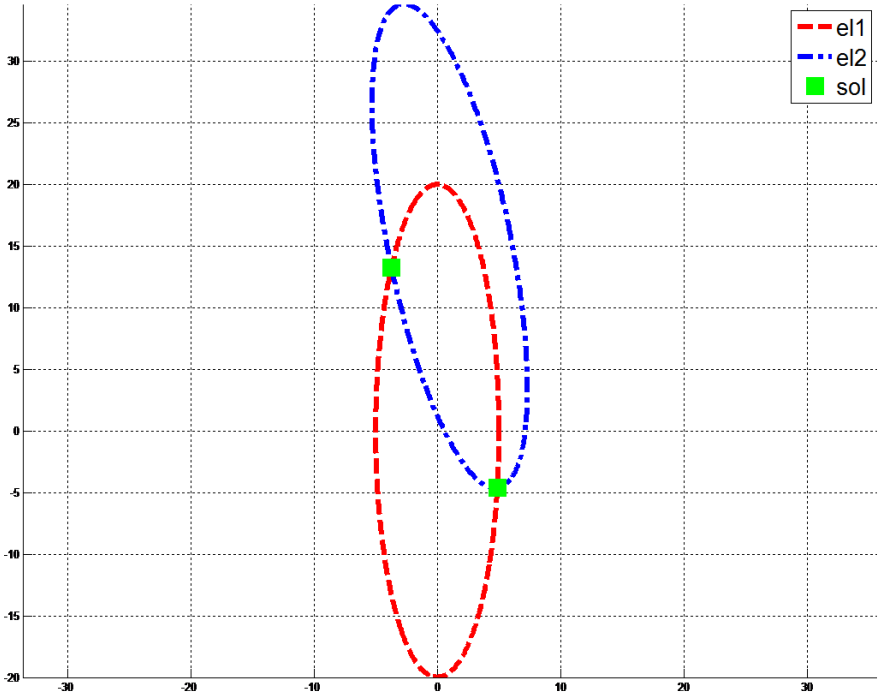


Figura D.2: Intersección entre dos elipses.

Los vectores de parámetros que definen cada una de las elipses son $e_1 = [a_1, b_1, c_1, d_1, e_1, f_1]$ y $e_2 = [a_2, b_2, c_2, d_2, e_2, f_2]$. Estos parámetros, dentro de la ecuación de una elipse, significan $el_1 : a_1x^2 + 2b_1xy + c_1y^2 + 2d_1x + 2e_1y + f_1 = 0$ y $el_2 : a_2x^2 + 2b_2xy + c_2y^2 + 2d_2x + 2e_2y + f_2 = 0$ respectivamente. Para calcular la intersección entre las dos elipses hay que resolver el siguiente sistema de ecuaciones:

$$\begin{cases} el_1 : a_1x^2 + 2b_1xy + c_1y^2 + 2d_1x + 2e_1y + f_1 = 0 \\ el_2 : a_2x^2 + 2b_2xy + c_2y^2 + 2d_2x + 2e_2y + f_2 = 0 \end{cases} \tag{D.6}$$

donde $a_1, b_1, c_1, d_1, e_1, f_1, a_2, b_2, c_2, d_2, e_2, f_2$ son los parámetros de las elipses y x, y son la solución al sistema de ecuaciones, $sol = [x, y]/sol \in \Re^2$.

Al resolver el sistema de ecuaciones se pueden encontrar singularidades. Estas singularidades se deben a los parámetros que definen las dos elipses. Esto se debe

a casos en que las dos elipses tengan los mismos ejes y el mismo ángulo de rotación. Debido a las singularidades, los pasos a seguir para calcular las intersecciones entre dos elipses son:

D.1.3.1 Paso 1: Calcular los puntos de intersección

Calcular los puntos intersección de las dos elipses utilizando el sistema de ecuaciones anterior.

D.1.3.2 Paso 2: Comprobar que los puntos pertenecen a las elipses

Comprobar que los puntos pertenecen a las dos elipses. Para ello utilizamos las siguientes ecuaciones:

$$\begin{aligned} error_1 &= a_1x_i^2 + 2b_1x_iy_i + c_1y_i^2 + 2d_1x_i + 2e_1y_i + f_1 \\ error_2 &= a_2x_i^2 + 2b_2x_iy_i + c_2y_i^2 + 2d_2x_i + 2e_2y_i + f_2 \end{aligned} \tag{D.7}$$

donde $error_1$ y $error_2$ son los parámetros que indicarán si los puntos pertenecen o no a la elipse. Si $error_i = 0 \Rightarrow p_i \in el_j$. Si $error_1$ y $error_2$ no superan un umbral de error introducido por el usuario, umb, si $error_1 < umb$ y $error_2 < umb$ entonces se termina y no se realizan los siguientes pasos, los puntos obtenidos son correctos. Si por el contrario $error_1 > umb$ y/o $error_2 > umb$ puede ser debido a las singularidades, con lo cual se continúa con los cálculos.

D.1.3.3 Paso 3: Incrementar el ángulo de una de las elipses

Modificar el ángulo de una de las dos elipses un umbral deseado, por ejemplo en 0,001 radianes.

D.1.3.4 Paso 4: Volver a calcular la intersección con la elipse modificada

Volver a calcular la intersección entre las dos elipses utilizando el sistema de ecuaciones anterior. Hay que tener en cuenta que la solución viene dada por este nuevo ángulo, con lo cual la solución no es la óptima.

D.1.3.5 Paso 5: Comprobar errores

Comprobar que el error ocasionado no es mayor al umbral introducido por el usuario, igual que en el paso 2. En este punto se tienen dos casos. Caso 1) $error_1 < umb$ y $error_2 < umb$, con lo cual se termina con los cálculos, los puntos obtenidos son los deseados. Caso 2) $error_1 > umb$ y/o $error_2 > umb$, debido a que la variación del ángulo provoca una gran variación de los puntos de intersección en las dos elipses. En este segundo caso lo que se hace es un proceso de optimización. Para ello se utiliza el "toolbox" de optimización de "MATLAB" con la ayuda del comando "fsolve". Como función de coste se tienen las dos ecuaciones de las dos elipses, y como parámetros las coordenadas x e y de los puntos intersección, se tiene:

$$\begin{aligned} eq(1) &= a_1x^2 + 2b_1xy + c_1y^2 + 2d_1x + 2e_1y + f_1 \\ eq(2) &= a_2x^2 + 2b_2xy + c_2y^2 + 2d_2x + 2e_2y + f_2 \end{aligned} \tag{D.8}$$

Los valores $eq(1)$ y $eq(2)$ son los funciones de coste a optimizar, de forma que, el proceso de optimización lo que pretende es variar las coordenadas x e y de los puntos intersección para que la solución de las dos ecuaciones sea menor al umbral introducido por el usuario, $eq(1) < umb$ y $eq(2) < umb$. La solución será el nuevo punto de intersección. Se repetirá este proceso para cada uno de los puntos intersección.

Al resolver el sistema de coordenadas, a parte de las singularidades, se pueden encontrar tres posibles casos.

D.1.3.6 Caso 1: Las elipses no se cortan en ningún punto

Si el_1 no se corta con el_2, el sistema de ecuaciones no tiene solución. En este caso se adopta que $|sol| = 0$ con lo que $sol = \{\emptyset\}$.

D.1.3.7 Caso 2: Las elipses son iguales

Si $el_1 = el_2$, el sistema de ecuaciones tiene infinitas soluciones. Calcular la intersección a dos elipses que sean exactamente iguales en nuestro caso no aporta ninguna información valida, con lo cual cuando se encuentra este caso se adopta que $|sol| = 0$ con lo que $sol = \{\emptyset\}$.

D.1.3.8 Caso 3: Las elipses se cortan en diferentes puntos

Si $el_1 \cap el_2$, el sistema de ecuaciones puede tener hasta un máximo de 4 puntos donde las dos elipses intersectan. Si $|sol| = 1 \Rightarrow sol = \{p_1 | p_1 = [x, y]\}$, si $|sol| = 2 \Rightarrow sol = \{p_1, p_2 | p_i = [x, y]\}$, si $|sol| = 3 \Rightarrow sol = \{p_1, p_2, p_3 | p_i = [x, y]\}$, si $|sol| = 4 \Rightarrow sol = \{p_1, p_2, p_3, p_4 | p_i = [x, y]\}$.

D.1.4 Intersección entre una elipse y una recta en el mismo plano

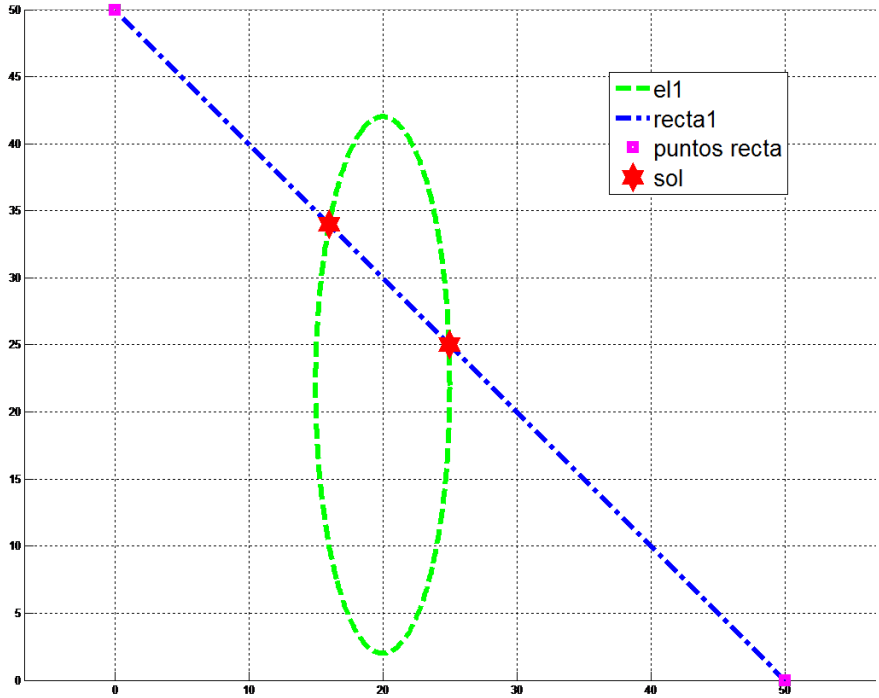


Figura D.3: Intersección entre una elipse y una recta en el mismo plano.

Una recta viene definida por dos puntos $\{p_1, p_2 | p_i = [x, y]\} / p_1, p_2 \in r_1$ donde $p_i \in \Re^2$. La ecuación de la recta es $r_1 : (y - y_1) = m_1(x - x_1)$ y su pendiente $m_1 = (y_2 - y_1)/(x_2 - x_1)$. Los parámetros que definen una elipse son $e_1 = [a_1, b_1, c_1, d_1, e_1, f_1]$. Estos parámetros, dentro de la ecuación de una elipse, significan $el_1 : a_1x^2 + 2b_1xy + c_1y^2 + 2d_1x + 2e_1y + f_1 = 0$. La intersección entre una recta y una elipse viene dada por el siguiente sistema de ecuaciones:

$$\begin{cases} (y - y_1) = \frac{y_2 - y_1}{x_2 - x_1}(x - x_1) \\ a_1x^2 + 2b_1xy + c_1y^2 + 2d_1x + 2e_1y + f_1 = 0 \end{cases} \tag{D.9}$$

donde x_1, x_2, y_1, y_2 son las coordenadas de los puntos de la recta, $a_1, b_1, c_1, d_1, e_1, f_1$ son los parámetros de la ecuación de la elipse y x, y son la solución al sistema de coordenadas, $sol = [x, y] / sol \in \Re^2$.

Para resolver el sistema de ecuaciones anterior se tienen dos posibles casos.

D.1.4.1 Caso 1: La elipse y la recta no se cortan

Si la elipse el_1 no se corta con la recta r_1 entonces el sistema de ecuaciones no tiene solución. En este caso se adopta que $|sol| = 0$ con lo que $sol = \{\emptyset\}$.

D.1.4.2 Caso 2: La elipse y la recta se cortan

Si $el_1 \cap r_1$ entonces se tienen dos posibles casos, que la recta sea tangente a la elipse y solo corte en un punto, con lo cual $|sol| = 1 \Rightarrow sol = \{p_3 | p_3 = [x, y]\}$, o que la recta corte a la elipse en dos puntos, con lo que $|sol| = 2 \Rightarrow sol = \{p_3, p_4 | p_i = [x, y]\}$.

D.1.5 Tangentes a dos elipses

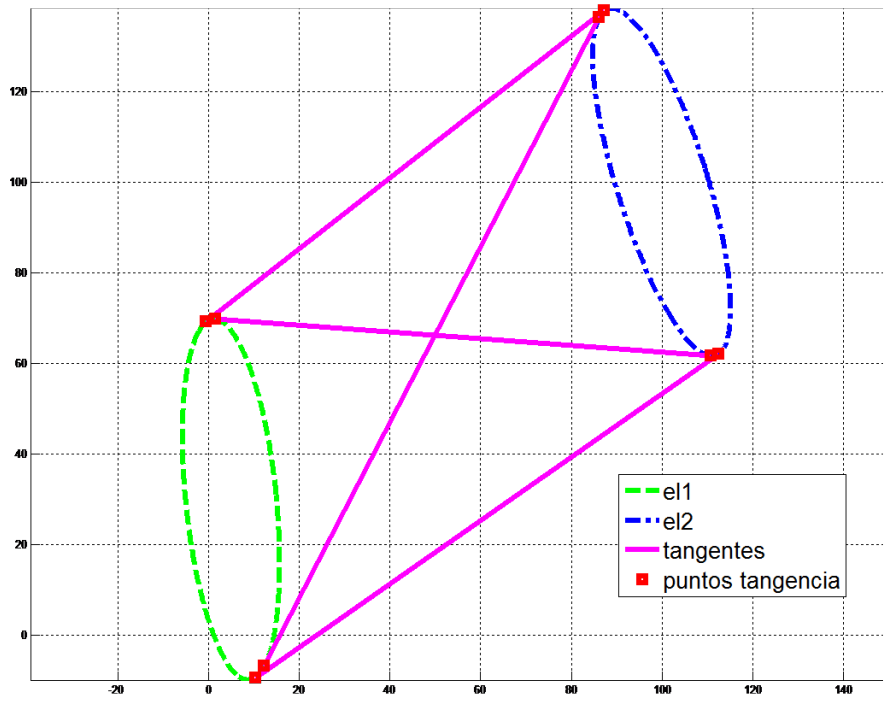


Figura D.4: Tangente a dos elipses.

Los vectores de parámetros que definen cada una de las elipses son $e_1 = [a_1, b_1, c_1, d_1, e_1, f_1]$ y $e_2 = [a_2, b_2, c_2, d_2, e_2, f_2]$. Estos parámetros, dentro de la ecuación de una elipse, significan $el_1 : a_1x^2 + 2b_1xy + c_1y^2 + 2d_1x + 2e_1y + f_1 = 0$ y $el_2 : a_2x^2 + 2b_2xy + c_2y^2 + 2d_2x + 2e_2y + f_2 = 0$ respectivamente. La ecuación de una recta tangente a una elipse dada es:

$$\begin{aligned} tan : (e^2 - cf)m^2 + 2(cd - bd)mk + (b^2 - ac)k^2 + \\ +2(de - bf)m + 2(bd - ae)k + (d^2 - af) = 0 \end{aligned} \tag{D.10}$$

Esta ecuación se obtiene de substituir la ecuación de una recta $r_1 : y = mx + k$ en la ecuación original de la elipse y haciendo nulo cada uno de los determinantes apropiados.

Para calcular las rectas tangentes a las dos elipses hay que resolver el siguiente sistema de ecuaciones:

$$\begin{cases} (e_1^2 - c_1f_1)m^2 + 2(c_1d_1 - b_1d_1)mk + (b_1^2 - a_1c_1)k^2 + \\ +2(d_1e_1 - b_1f_1)m + 2(b_1d_1 - a_1e_1)k + (d_1^2 - a_1f_1) = 0 \\ \\ (e_2^2 - c_2f_2)m^2 + 2(c_2d_2 - b_2d_2)mk + (b_2^2 - a_2c_2)k^2 + \\ +2(d_2e_2 - b_2f_2)m + 2(b_2d_2 - a_2e_2)k + (d_2^2 - a_2f_2) = 0 \end{cases} \tag{D.11}$$

donde $a_1, b_1, c_1, d_1, e_1, f_1, a_2, b_2, c_2, d_2, e_2, f_2$ son los parámetros de las elipses y k, m son los parámetros de la ecuación de la recta.

Una vez se tienen las rectas tangentes, se calculan los puntos intersección de estas rectas con las dos elipses, obteniendo dos puntos intersección para cada una de las rectas, cada punto para una elipse diferente.

Al resolver el sistema de ecuaciones pueden aparecer singularidades. Al igual que anteriormente, estas singularidades se deben a que las elipses tienen parámetros en común como puede ser el ángulo o las dimensiones de los ejes. Debido a las singularidades, los pasos a seguir para calcular las rectas tangentes son:

D.1.5.1 Paso 1: Calcular los puntos tangentes

Calcular los puntos tangentes de las dos elipses, utilizando el sistema de ecuaciones anterior para calcular los parámetros de la ecuación de la recta y posteriormente calcular las intersecciones de estas rectas con las dos elipses.

D.1.5.2 Paso 2: Comprobar que los puntos pertenecen a las elipses

Comprobar que los puntos de tangencia pertenecen a sus respectivas elipses a las que tendrían que pertenecer. Para ello se utilizan las siguientes ecuaciones:

$$\begin{aligned} error_1 &= a_1 x_i^2 + 2b_1 x_i y_i + c_1 y_i^2 + 2d_1 x_i + 2e_1 y_i + f_1 \\ error_2 &= a_2 x_i^2 + 2b_2 x_i y_i + c_2 y_i^2 + 2d_2 x_i + 2e_2 y_i + f_2 \end{aligned} \tag{D.12}$$

donde $error_1$ y $error_2$ son los parámetros que indicarán si los puntos pertenecen o no a la elipse. Si $error_i = 0 \Rightarrow p_i \in el_j$. Si $error_1$ y $error_2$ no superan un umbral de error introducido por el usuario, umb, si $error_1 < umb$ y $error_2 < umb$ entonces se termina y no se realizan los siguientes pasos, los puntos obtenidos son correctos. Si por el contrario $error_1 > umb$ y/o $error_2 > umb$ puede ser debido a las singularidades, con lo cual se continúa con los cálculos.

D.1.5.3 Paso 3: Incrementar el ángulo y el centro de una de las elipses

Modificar el ángulo de una de las dos elipses un umbral deseado, por ejemplo en 0,1 radianes. Incrementar el centro de una de las dos elipses, por ejemplo en 0,1 mm.

D.1.5.4 Paso 4: Volver a calcular las rectas tangentes

Volver a calcular las rectas tangentes entre las dos elipses utilizando el sistema de ecuaciones anterior. Volver a calcular las intersecciones de estas rectas con las dos elipses y obtener los puntos de tangencia.

D.1.5.5 Paso 5: Comprobar el error ocasionado

Comprobar que el error ocasionado no es mayor al umbral introducido por el usuario, igual que en el paso 2. En este punto se tienen dos casos. Caso 1) $error_1 < umb$ y $error_2 < umb$, con lo cual se termina con los cálculos, los puntos obtenidos son los deseados. Caso 2) $error_1 > umb$ y/o $error_2 > umb$, debido a que la variación del ángulo provoca una gran variación de los puntos de tangencia en las dos elipses. En este segundo caso lo que se hace es un proceso de optimización. Para ello se utiliza el "toolbox" de optimización de "MATLAB" con la ayuda del comando "fsolve". Como función de coste se tiene: que los puntos de tangencia pertenezcan a las elipses, que las pendientes de las dos rectas sean iguales, y que las dos rectas sean la misma. Como parámetros, las coordenadas x_1, y_1, x_2, y_2 de

los dos puntos de tangencia que se han calculado para cada recta tangente, se tiene:

$$\begin{aligned} eq(1) &= a_1x_1^2 + 2b_1x_1y_1 + c_1y_1^2 + 2d_1x_1 + 2e_1y_1 + f_1 \\ eq(2) &= a_2x_2^2 + 2b_2x_2y_2 + c_2y_2^2 + 2d_2x_2 + 2e_2y_2 + f_2 \\ eq(3) &= \frac{2a_1x_1+2b_1y_1+2d_1}{b_1x_1+2c_1y_1+2e_1} - \frac{2a_2x_2+2b_2y_2+2d_2}{b_2x_2+2c_2y_2+2e_2} \\ eq(4) &= \left(-\frac{2a_1x_1+2b_1y_1+2d_1}{b_1x_1+2c_1y_1+2e_1}x_1 + y_1\right) - \left(-\frac{2a_2x_2+2b_2y_2+2d_2}{b_2x_2+2c_2y_2+2e_2}x_2 + y_2\right) \end{aligned} \tag{D.13}$$

Los valores $eq(1), eq(2), e1(3), eq(4)$ son los funciones de coste a optimizar, de forma que el proceso de optimización lo que pretende es variar las coordenadas x_1, y_1, x_2, y_2 para que la solución de las dos ecuaciones sea menor al umbral introducido por el usuario. La solución serán los dos nuevos puntos de la recta tangente a las dos elipses. Se repetirá todo el proceso para cada una de las rectas tangentes.

Según la colocación de las dos elipses se tienen diferentes posibles casos.

D.1.5.6 Caso 1: Si las dos elipses se cortan

Si $el_1 \cap el_2$ entonces se tienen dos rectas tangentes a las elipses, que serán las rectas tangentes exteriores. $|sol| = 2 \Rightarrow sol = \{r_1, r_2 | r_i : y = mx + k\}$. Al calcular la intersección de dichas rectas con cada una de las elipses, se obtienen dos puntos de tangencia para cada una de ellas, se tiene que $|sol\_el_1 = 2 \Rightarrow sol = \{p_1, p_2 | p_i = [x, y]\}$ y $|sol\_el_2 = 2 \Rightarrow sol = \{p_3, p_4 | p_i = [x, y]\}$.

D.1.5.7 Caso 2: Si las dos elipses no se cortan

Si el_1 no intersecta con el_2 entonces se tienen 4 rectas tangentes, dos rectas tangentes interiores y dos rectas tangentes exteriores. $|sol| = 4 \Rightarrow sol = \{r_1, r_2, r_3, r_4 | r_i : y = mx + k\}$. Al calcular la intersección de dichas rectas con cada una de las elipses, se obtienen 4 puntos de tangencia para cada una de ellas, se tiene que $|sol\_el_1 = 4 \Rightarrow sol = \{p_1, p_2, p_3, p_4 | p_i = [x, y]\}$ y $|sol\_el_2 = 4 \Rightarrow sol = \{p_5, p_6, p_7, p_8 | p_i = [x, y]\}$.

D.1.6 Punto medio entre los centros de dos elipses

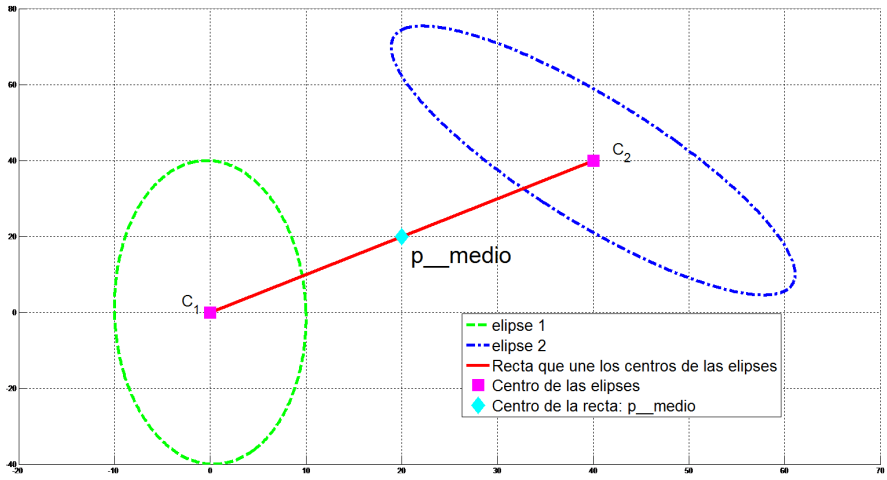


Figura D.5: Punto medio de la recta que une los centros de dos elipses.

Se trata de calcular el punto medio entre los centros de dos elipses. Si se conocen los centros de las elipses $c_1 = [x_1 y_1]$ y $c_2 = [x_2 y_2]$, calcular el punto medio de la recta que une los dos centros consiste en:

$$p\_medio = \left[\frac{x_1+x_2}{2} \quad \frac{y_1+y_2}{2}\right] \tag{D.14}$$

D.1.7 Ángulo que forma un punto respecto a su elipse

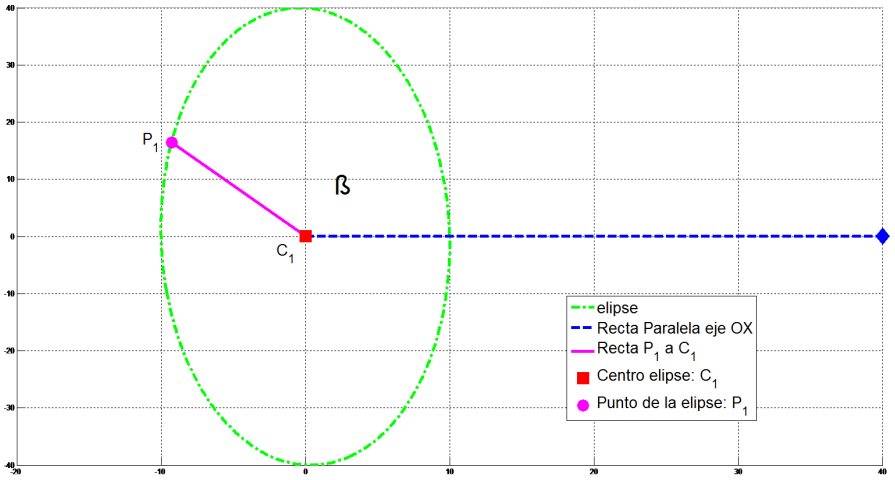


Figura D.6: Ángulo que forma un punto con respecto a su elipse.

Se trata de calcular el ángulo de un punto perteneciente a una elipse con respecto a un eje paralelo al eje OX que pasa por el punto medio entre los centros de dos elipses.

La información que se tiene es: un punto perteneciente a la elipse $p_1 = [x_1, y_1]$, el centro de la elipse $c_1 = [x_2, y_2]$, el ángulo de rotación de la elipse α. c_1 y α son los respectivos parámetros de la elipse sobre la que se encuentra el punto deseado.

Primero lo que se debe de hacer es definir los dos vectores. Una vector será paralelo al eje OX que pasa por c_1, por ejemplo $\vec{v}_1 = [x_2 + d, y_2] - [x_2, y_2]$, donde x_2, y_2 son las coordenadas del punto c_1 y d es una distancia arbitraria, y el otro vector definirá la recta que une los puntos c_1 y p_1, $v_2 = [x_1, y_1] - [x_2, y_2]$, donde x_2, y_2 y x_1, y_1 son las coordenadas del punto c_1 y p_1 respectivamente.

Se calcula la matriz de rotación para un ángulo α dado, rotación de la elipse:

$$R = \begin{bmatrix} cos(\alpha) & sin(\alpha) \\ -sin(\alpha) & cos(\alpha) \end{bmatrix} \tag{D.15}$$

Se realiza la rotación del vector v_2 para deshacer la rotación dada a la elipse y poder calcular el ángulo deseado.

$$rot_1 = Rv_2^{'} \tag{D.16}$$

Tras realizar la operación se tienen que aplicar las siguientes transformaciones para obtener el vector deseado:

$$v_3 = \left[\frac{rot_1(1)}{\sqrt{ra^2 - rot_1(1)^2}} \right] \tag{D.17}$$

donde $rot_1(1)$ es la primera componente del vector rot_1 y ra es la longitud del eje menor de la elipse donde se encuentra el punto de tangencia.

Ahora queda calcular el ángulo formado por los vectores v_3 y v_1:

$$ang = \frac{v_3(1)v_1(1) - v_3(2)v_1(2)}{\sqrt{v_3(1)^2 + v_3(2)^2}\sqrt{v_1(1)^2 + v_1(2)^2}} \tag{D.18}$$

donde ang es el ángulo que forman los dos vectores en radianes. Analizando el resultado de la componente $rot_1(2)$ se conoce en qué cuadrante se está trabajando. Hay que tener en cuenta que si $rot_1(2) >= 0$ no hay que realizar ninguna corrección, si por lo contrario $rot_1(2) < 0$ hay que realizar una corrección en ang, donde $ang = 2\pi - ang$. De forma que la solución final será el valor de ang en radianes.

D.1.8 Ángulo que forma un punto de la elipse con respecto a un eje paralelo a OX que pasa por el punto medio de la recta que une los centros de las elipses dadas

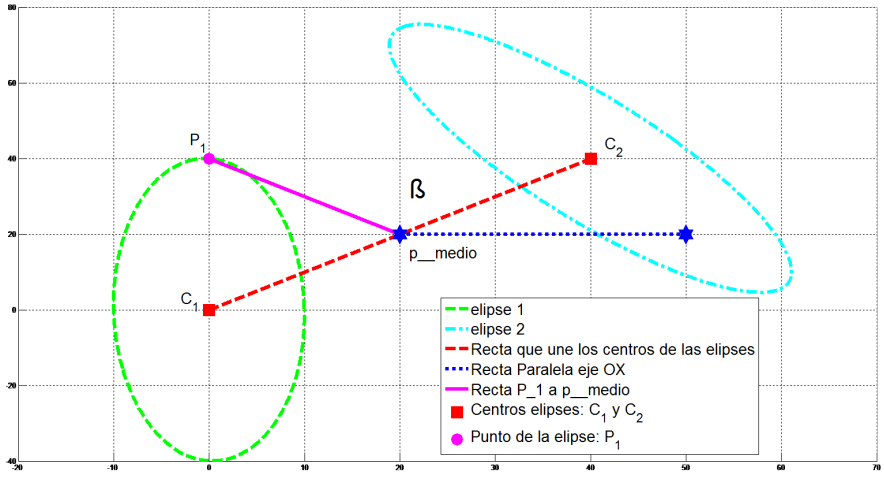


Figura D.7: Ángulo que forma un punto de la elipse con respecto a un eje paralelo a OX.

Se trata de calcular el ángulo de un punto perteneciente a una elipse con respecto a un eje paralelo al eje OX que pasa por el punto medio entro los centros de dos elipses.

La información que se tiene es: el punto de tangencia de la elipse $p_1 = [x, y]$, y el punto medio que une los dos centros de las elipses $p\_medio = [x2, y_2]$.

Primero lo que se debe de hacer es definir los dos vectores. Una vector será paralelo al eje OX que pasa por p_medio, por ejemplo $v_1 = [x_2 + d, y_2] - [x_2, y_2]$, donde x_2, y_2 son las coordenadas del punto p_medio y d es una distancia arbitraria, y el otro vector definirá la recta que une los puntos p_medioy p_1, $v_2 = [x_1, y_1] - [x_2, y_2]$, donde x_2, y_2 y x_1, y_1 son las coordenadas del punto p_1 y p_medio respectivamente.

Ahora queda calcular el ángulo formado por los vectores v_2 y v_1:

$$ang = \frac{v_2(1)v_1(1) - v_2(2)v_1(2)}{\sqrt{v_2(1)^2 + v_2(2)^2}\sqrt{v_1(1)^2 + v_1(2)^2}} \tag{D.19}$$

donde *ang* es el ángulo que forman los dos vectores en radianes. Analizando el resultado de la componente $p_1(2)$ se conoce en qué cuadrante se está trabajando. Hay que tener en cuenta que si $p_1(2) >= p\_medio(2)$ no hay que realizar ninguna corrección, si por lo contrario $p_1(2) < p\_medio(2)$ hay que realizar una corrección en ang, donde $ang = 2\pi - ang$.

D.1.9 Conocer si un punto está dentro de la elipse o no

Lo que se pretende es que dado un punto $p_1 = [x, y]$ conocer si este punto está dentro o fuera de la elipse $el_1 : (ax^2 + 2bxy + cy^2 + 2dx + 2ey + f = 0)$. La información dada de la elipse son: el centro de la elipse, $c_1 = [x_2, y_2]$, los ejes mayor y menor, r_1, r_b respectivamente, y el ángulo de la elipse, α.

Lo primero que hay que hacer es calcular el vector formado por p_1 y c_1, $v_1 = [x_1, y_1] - [x_2, y_2]$.

Se calculará la matriz de rotación para un ángulo α inicial de la elipse:

$$R = \begin{bmatrix} cos(\alpha) & sin(\alpha) \\ -sin(\alpha) & cos(\alpha) \end{bmatrix} \tag{D.20}$$

Se realiza la rotación del vector v_1 y se obtiene un nuevo punto.

$$p_2 = Rv_1^{'} \tag{D.21}$$

Substituir p_2 en la ecuación de la elipse de forma que la solución indica si el punto esta fuera o dentro de la elipse.

$$sol = ap_2(1)^2 + 2bp_2(1)p_2(2) + cp_2(2)^2 + 2dp_2(1) + 2ep_2(2) + f \tag{D.22}$$

Con lo que $|sol| = 1$, con lo cual si $sol \leq \Rightarrow p_2 \in el_1$, si por lo contrario $sol > 0 \Rightarrow p_2 \notin el_1$.

D.1.10 Dibujar el tramo deseado: recta

Para dibujar los tramos deseados en forma de recta solo se necesitan las coordenadas de los dos puntos extremos de la recta. Estos puntos son $p_1 = [x_1, y_1]$ y $p_2 = [x_2, y_2]$, punto inicial y final respectivamente. Estos dos puntos pertenecen a dos elipses diferentes.

Se creará un vector, $vect\_x$, que divida el tramo de x_1 a x_2 en tantos puntos como se desee dibujar la recta, Nb_recta indica el número de puntos. Para ello

se utiliza el comando de "MATLAB" "linspace". Este comando lo que hace es, coger un valor inicial x_1, coger un valor final x_2, y el tramo formado por estos dos puntos lo divide en tantas veces como indica la variable Nb_recta.

Se obtiene la pendiente de la recta, m, utilizando los dos puntos p_1 y p_2. $m = (y_2 - y_1)/(x_2 - x_1)$.

Una vez se tiene esta información, lo que se hace es sustituir cada coordenada x del vector calculado en la ecuación de la recta siguiente y calcular su coordenada y:

$$vect\_y = m(vect - x_1) + y_1 \tag{D.23}$$

donde m es la pendiente de la recta formada por p_1 y p_2, x_1 e y_1 son las coordenadas del punto inicial de partida, $vect\_x$ es el vector de las coordenadas x. La solución, $vect\_y$, será un vector que satisface la ecuación de la recta para cada uno de los elementos del vector $vect\_x$. Con lo cual $vect\_x$ y $vect\_y$ serán los vectores que se utilizarán para dibujar los puntos deseados. Estos vectores tendrán como dimensión el parámetro Nb_recta, que es en el número de puntos en que se ha dividido la recta.

D.1.11 Dibujar el tramo deseado: elipse

Para dibujar los tramos deseados en forma de elipse se necesitan las coordenadas de dos puntos de la elipse, $p_1 = [x_1, y_1]$ y $p_2 = [x_2, y_2]$, punto inicial y final respectivamente, que pertenecen a la misma elipse, los ángulos que forman estos puntos con respecto al centro de la elipse, α_1 y α_2, los valores de los ángulos está comprendido entre 0 y 2π, las coordenadas del centro de la elipse a la que pertenecen y los valores de los ejes r_a y r_b, menor y mayor respectivamente.

Partiendo de que los dos puntos pertenecen a la misma elipse, según los valores de α_1 y α_2 se tiene que: si $\alpha_1 < \alpha_2$ se crea el vector $vect\_ang$ utilizando el comando de "MATLAB" "linspace", este comando lo que hace es, coger un valor inicial α_1, coger un valor final α_2, y el tramo formado por estos dos ángulos lo divide en tantas veces como se quiere. Nb_elipse define el número de puntos en que se quiere dividir el tramo; si por el contrario $\alpha_1 > \alpha_2$ se tiene que hacer dos tramos para calcular el vector $vect\_ang$, el primer tramo comprenderá los valores de α_1 a 2π, y el segundo tramo de 0 a α_2, $[\alpha_1, 2\pi] \cup [0, \alpha_2]$. Se obtiene un vector con diferentes ángulos.

Se coge el vector $vect\_ang$ y se calcula:

$$\begin{aligned} co_1 &= cos(\alpha_1) \\ si_1 &= sin(\alpha_1) \\ xpos &= ra \cdot cos(vect\_ang) \cdot co_1 - si_1 \cdot rb \cdot sin(vect\_ang) + c_x \\ ypos &= ra \cdot cos(vect\_ang) \cdot si_1 + co_1 \cdot rb \cdot sin(vect\_ang) + c_y \end{aligned} \tag{D.24}$$

donde *xpos* e *ypos* son los vectores de las coordenadas x e y respectivamente de los puntos que forman el tramo deseado y son los que se utilizan para dibujar el tramo de la elipse deseado. Estos dos vectores tendrán la dimensión en que se ha dividido el tramo, serán de dimensión *Nb_elipse*. El vector *vect_ang* contiene los incrementos del ángulo desde el valor inicial α_1 al valor final α_2. Las variables r_a y r_b son los radios menor y mayor respectivamente de la elipse dada. Las coordenadas c_x y c_y son las coordenadas x e y respectivamente del centro de la elipse.

D.1.12 Diferencia entre dos elipses

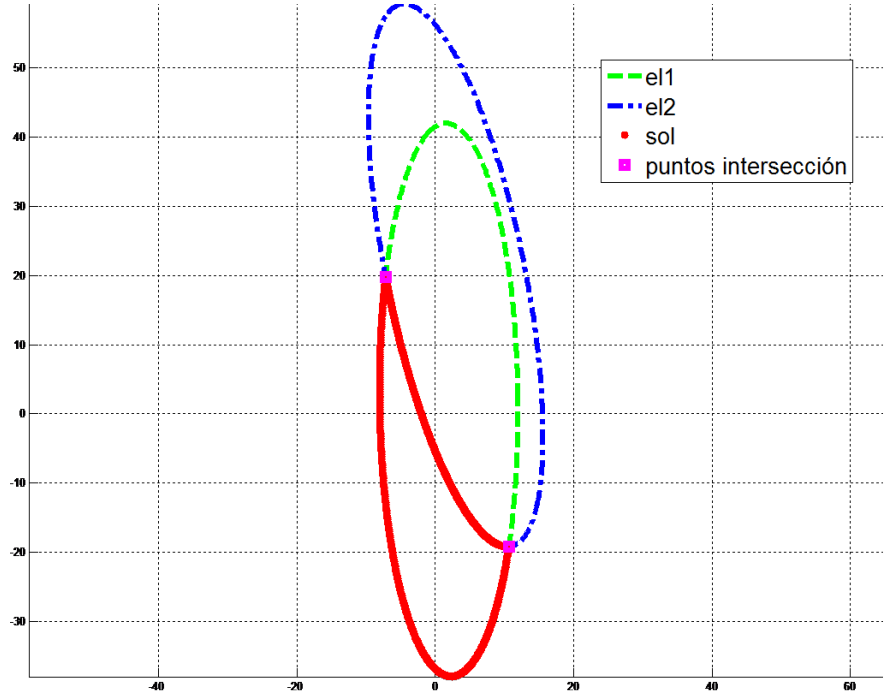


Figura D.8: Diferencia entre dos elipses.

Dadas dos elipses, una principal y una secundaria, $el_1 : (a_1x^2 + 2b_1xy + c_1y^2 + 2d_1x + 2e_1y + f_1 = 0)$ y $el_1 : (a_2x^2 + 2b_2xy + c_2y^2 + 2d_2x + 2e_2y + f_2 = 0)$ respectivamente. Hay que realizar la sustracción de el_2 a el_1.

Primero se tiene que realizar la intersección entre dos elipses visto en el anexo D, apartado D.1.3. Esto llevará a tener varias soluciones según el número de puntos de intersección.

D.1.12.1 Caso 1: No hay intersecciones, o un solo punto de corte

Si el_2 no intersecta con el_1 significa que la solución es la misma elipse el_1, $sol = el_1$. Del mismo modo si solo hay un punto de corte significa que las dos elipses son tangentes, con lo cual se sigue teniendo que $sol = el_1$.

D.1.12.2 Caso 2: Intersecta en dos puntos o más

Si $el_2 \cap el_1$ en dos o más puntos, entonces la solución viene dada por una parte de la elipse 1 y por otra parte de la elipse 2. Para conocer qué parte de cada elipse se tiene que dibujar lo que se hace es:

1. Ángulo que forman los puntos intersección con la elipse:

   Lo que se pretende es conocer el ángulo de cada punto intersección con la elipse a la cual pertenece. Los cálculos a realizar han sido vistos en la sección D.1.7.

   Esto implica que cada punto intersección tendrá dos ángulos diferentes, uno perteneciente a la elipse principal, y otro a la elipse secundaria.

2. Ordenar los puntos intersección según el ángulo:

   Una vez conocidos los ángulos para cada punto de intersección, se tienen que ordenar según el ángulo calculado. Cada elipse tendrá su propio orden de los puntos intersección.

3. Introducir nuevos puntos intermedios:

   Al ordenar los puntos según el ángulo formado, se definen los tramos de cada una de las elipses.

   Lo que hay que hacer es, en cada uno de estos tramos, definir un nuevo punto, de esta forma cada tramo se convierte ahora en dos tramos. Se disponen de los ángulos donde empieza y termina cada tramo, para calcular el nuevo

punto medio y definir los nuevos tramos, hay que calcular las coordenadas del punto que cumpla que el ángulo de ese nuevo punto sea el ángulo medio de los ángulos iniciales, es decir, por ejemplo si se tienen dos puntos de intersección, un punto en 0 radianes y el segundo en π radianes, el ángulo medio seria $\pi/2$. Se tiene que calcular el punto de la elipse que se encuentre a $\pi/2$ radianes.

Para conocer las coordenadas de dicho punto hay que realizar los siguientes cálculos:

$$\begin{cases} x = r_a \cdot cos(\beta) \cdot cos(\alpha) - sin(\alpha) \cdot r_b \cdot sin(\beta) + c_x \\ y = r_a \cdot cos(\beta) \cdot sin(\alpha) + cos(\alpha) \cdot r_b \cdot sin(\beta) + c_y \end{cases} \Rightarrow sol = [x \quad y] \quad \text{(D.25)}$$

donde x, y son las coordenadas del punto solución, α es el ángulo de rotación de la elipse, r_a el eje menor de la elipse, r_b el eje mayor de la elipse, c_x, c_y las coordenadas x e y respectivamente del centro de la elipse, β es el ángulo deseado al cual debe de estar el punto solución.

Hay que calcular estos puntos intermedios para cada una de las elipses, ya que cada punto intersección forma un ángulo diferente en cada una de las elipses.

4. Seleccionar los tramos de la elipse secundaria:

   Para seleccionar los tramos deseados de la elipse secundaria se disponen de los ángulos de los puntos intersección, las coordenadas de los nuevos puntos intermedios para esta elipse, y la ecuación de la elipse principal.

   Hay que comprobar qué puntos intermedios calculados están dentro de la elipse secundaria. Los cálculos a realizar han sido vistos en la sección D.1.9.

   Los tramos pertenecientes a los puntos intermedios que están fuera de la elipse secundaria serán los tramos deseados, por el contrario, los tramos pertenecientes a los puntos que están dentro de la elipse secundaria, deberán de ser descartados.

   Los tramos deseados están definidos por dos puntos intersección, cada punto tiene un ángulo distinto dentro de la elipse secundaria, el tramo seleccionado es el tramo que comprende estos dos ángulos.

5. Seleccionar los tramos de la elipse principal:

Para seleccionar los tramos deseados de la elipse principal se disponen de los ángulos de los puntos intersección, las coordenadas de los nuevos puntos intermedios para esta elipse, y la ecuación de la elipse secundaria.

Hay que comprobar qué puntos intermedios calculados están fuera de la elipse secundaria. Los cálculos a realizar han sido vistos en la sección D.1.9.

Los tramos pertenecientes a los puntos intermedios que están fuera de la elipse secundaria serán los tramos deseados, por el contrario, los tramos pertenecientes a los puntos que están dentro de la elipse secundaria, deberán de ser descartados.

Los tramos deseados están definidos por dos puntos intersección, cada punto tiene un ángulo distinto dentro de la elipse principal, el tramo seleccionado es el tramo que comprende estos dos ángulos.

Llegados a este punto ya disponemos de toda la información necesaria para calcular los puntos de cada tramo. Se obtiene una matriz ordenada según los ángulos para cada elipse:

$$\begin{bmatrix} \alpha_1 & c_{x_1} & c_{y_1} & \beta_1 & \text{elipse}_1 \\ \alpha_2 & c_{x_2} & c_{y_2} & \beta_2 & \text{elipse}_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_i & c_{x_i} & c_{y_i} & \beta_i & \text{elipse}_i \end{bmatrix} \tag{D.26}$$

donde α es el ángulo con respecto al centro de la recta que une las dos elipses, c_x y c_y son las coordenadas del centro de la elipse a la que pertenece el punto, β es el ángulo que forma el punto con respecto al centro de la elipse a la que pertenece, *elipse* es el número de la elipse a la que pertenece el punto tratado.

6. Calcular los puntos de cada tramo:

   Ahora hay que calcular los puntos de cada tramo, para ello se utilizan los procedimientos vistos en el apartado D.1.11. La información necesaria serán los dos puntos que forman cada tramo, el ángulo que forman estos con el centro de la elipse a la que pertenecen, los ejes mayores y menores y el centro de la elipse. Hay que unir los puntos de la matriz de dos en dos siempre y cuando sean de la misma elipse.

D.1.13 Diferencia entre "N" elipses

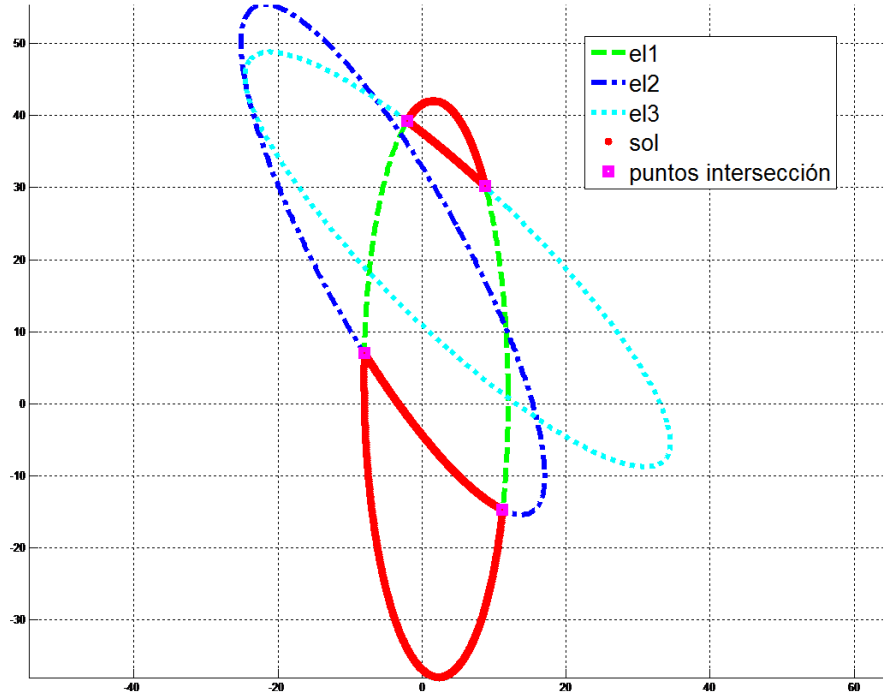


Figura D.9: Diferencia entre "N" elipses.

Se trata de una ampliación de la sección D.1.12, en este caso se pretende realizar la sustracción de "n" elipses a una elipse principal. Partiendo de los parámetros de cada una de las "n" elipses, los pasos a seguir son:

D.1.13.1 Paso 1: Intersección entre cada una de las elipses

Se tiene que realizar la intersección de todas las elipses entre ellas. La intersección entre elipses se ha visto en la sección D.1.3. El resultado es una matriz donde se encuentra cada punto intersección y la información de en qué elipses se encuentra ese punto.

D.1.13.2 Paso 2: Se seleccionan los puntos necesarios

Una vez se tiene la matriz de puntos, hay que seleccionar los puntos que nos interesan. Para ello los puntos a seguir son:

1. Se eliminan los puntos que estén fuera de la elipse principal:

   Los puntos intersección que están fuera de la elipse principal, se debe a intersecciones entre esferas diferentes a esta, con lo cual, estos puntos no interesan. Para eliminar estos puntos intersección se utilizará el procedimiento visto en la sección D.1.9.

2. Eliminar los puntos que estén dentro de alguna esfera que no sea la principal:

   Todos los puntos que estén dentro de alguna elipse además de estar dentro de la principal deben de ser eliminados ya que la zona en la que se encuentran será eliminada por alguna de las elipses. Para conocer los puntos que estén dentro de las elipses se utiliza el procedimiento visto en la sección D.1.9. Se obtiene la matriz de puntos final de los puntos que nos interesan.

D.1.13.3 Paso 3: Se seleccionan los tramos de las elipses que nos interesan

Una vez conocidos los puntos de intersección deseados, hay que conocer qué tramos de cada una de las elipses interesan. Para seleccionar estos tramos hay que seguir un proceso similar al visto en la sección D.1.12.

En la matriz obtenida se puede conocer qué puntos pertenecen a cada una de las elipses. Para cada una de las elipses, se deben realizar los siguientes puntos:

1. Seleccionar los puntos intersección de una misma elipse:

   Obtener una matriz con la información de todos aquellos puntos intersección que pertenecen a una misma elipse.

2. Calcular los ángulos de los puntos con respecto al centro de la elipse a la que pertenecen:

   Obtener para cada punto intersección, el ángulo que forma este con el centro de la elipse a la que pertenece. El procedimiento a seguir es el visto en la sección D.1.7.

3. Introducir puntos intermedios:

De la misma forma que en la sección D.1.12.2, punto 3, se pretende introducir unos puntos intermedios que pertenezcan a la elipse tratada. De esta forma se puede conocer los tramos que interesan.

4. Se seleccionan los tramos deseados:

   Cada uno de los puntos intermedios calculado, divide en dos un tramo dado. Para conocer si este tramo dado nos interesa o no hay que coger el punto intermedio y calcular si este punto está dentro de alguna de las otras elipses ya que se supone que este punto ya está dentro de la principal. Se tienen dos casos: Caso 1) si el punto intermedio está dentro de alguna elipse a parte de la principal, el tramo al cual pertenece este punto intermedio no nos interesa y es descartado; Caso 2) si por el contrario, el punto intermedio calculado no está dentro de ninguna otra esfera, solo dentro de la principal, el tramo al cual pertenece este punto intermedio nos interesa, se cogen los dos puntos extremos que definen este tramo.

   Para la elipse dada se obtienen unos tramos deseados, cada tramo definido por dos puntos extremos.

5. Repetir los puntos anteriores hasta la última elipse:

   Se repiten todos los puntos (Punto 1 al Punto 6) para cada una de las "n" elipses. Al finalizar se obtiene una matriz con la información de todos aquellos tramos deseados.

D.1.13.4 Paso 4: Calcular los puntos de cada tramo

La matriz final que define cada uno de los tramos tiene el siguiente aspecto:

$$\begin{bmatrix} \alpha_1 & c_{x_1} & c_{y_1} & \beta_1 & \text{elipse}_1 \\ \alpha_2 & c_{x_2} & c_{y_2} & \beta_2 & \text{elipse}_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \alpha_i & c_{x_i} & c_{y_i} & \beta_i & \text{elipse}_i \end{bmatrix} \tag{D.27}$$

donde α es el ángulo con respecto al eje paralelo a OX que pasa por el punto medio de los centro de todas las elipses, c_x y c_y son las coordenadas del centro de la elipse a la que pertenece el punto, β es el ángulo que forma el punto con respecto al eje paralelo a OX que pasa por el centro de la elipse a la que pertenece, *elipse* es el número de la elipse a la que pertenece el punto tratado.

Ahora se agrupan estos puntos de dos en dos (fila 1 con la fila 2, la fila 3 con la fila 4,...) esto proporciona la información de cada uno de los tramos, punto inicial y punto final respectivamente. Utilizando ahora los cálculos vistos en la sección D.1.11, se calculan los puntos de cada uno de los tramos.

Se obtienen dos vectores de coordenadas, el correspondiente a las coordenadas de las x y el correspondiente a las coordenadas de las y. Estos vectores son los que contienen todos los puntos deseados y definen el área de la elipse principal resultante de sustraer todas las "n" elipses a esta.

Apéndice E

Anexo: Álgebra de cuaterniones

A continuación se describen las ecuaciones báscias necesarias para dar solución a la cinemática directa e inversa utilizando cuaterniones duales. Estas ecuaciones sirven para desarrollar la cinemática de cadenas cinemáticas vista en el capítulo 5.

E.1 Introducción

En esta tesis se presenta un método de nueva formulación para resolver el problema cinemático de las multi-cadenas cinemáticas. El objetivo principal es formalizar el problema de cinemática en una forma cerrada compacta y evitar problemas de singularidad en la solución de la cinemática inversa. Este nuevo método se basa en la "screw theory" y en el "álgebra de cuaterniones". La teoría del tornillo o "screw theory" es una forma efectiva de establecer una descripción global de cuerpo rígido y evita las singularidades debidas al uso de las coordenadas locales. Y también el doble cuaternión que es el operador más compacto y eficiente para expresar el desplazamiento de "screw", se utiliza como un operador de movimiento de "screw" para obtener la formulación de una forma cerrada compacta. Las soluciones a la cinemática inversa se obtienen mediante el uso de los problemas de "Paden-Kahan".

El método más común para dar solución a la cinemática de cadenas cinemáticas, es el método de "Denavit-Hartenberg (DH)". Se usan las matrices homogéneas de $4x4$ como operador de transformación y sufren de problemas de singularidad Sariyildiz 2009. Otro método principal de la cinemática de cadenas cinemáticas es la teoría del tornillo (screw theory) que se basa en la línea de enfoque de las transformaciones. En la teoría del tornillo, cada transformación de un cuerpo rígido o un sistema de coordenadas con respecto a otro sistema de coordenadas se puede expresar por un desplazamiento de "screw", que es una translación a lo largo de un eje λ con una rotación por un ángulo θ sobre el mismo eje Ball R 1900. Esta descripción de la transformación es la base de la teoría del tornillo.

Hay dos ventajas principales de utilizar la teoría del tornillo para describir la cinemática de cuerpos rígidos. La primera es que permite una descripción global del movimiento del cuerpo rígido que no sufren de singularidades debidas al uso de coordenadas locales. La segunda es que la teoría del tornillo proporciona una descripción geométrica del movimiento rígido que simplifica enormemente el análisis de los mecanismos Huang y Yao 1999.

En este trabajo, se presenta un nuevo método de formulación para resolver el problema cinemático de las cadenas cinemáticas.

E.2 Conocimientos matemáticos previos

E.2.1 Cuaterniones

En matemática, los cuaterniones son números híper-complejos de rango 4, construyendo un vector del espacio de cuatro dimensiones sobre el campo de números reales. Los cuaterniones pueden ser representados de la siguiente forma:

$$q = (q_s, q_v) \tag{E.1}$$

donde q_s es un escalar y $q_v = (q_1, q_2, q_3)$ es un vector. Al cuaternión con $q_v = 0$ se le llama cuaternión real y al cuaternión con $q_s = 0$ se le llama cuaternión puro (o vector de cuaternión).

E.2.2 Operaciones con cuaterniones

E.2.2.1 Suma de cuaterniones

$$q_a + q_b = (q_{aS} + q_{bS}), (q_{aV} + q_{bV}) \tag{E.2}$$

E.2.2.2 Producto de cuaterniones

$$q_a \otimes q_b = (q_{aS} q_{bS} - q_{aV} \cdot q_{bV}), (q_{aS} q_{bV} + q_{bS} q_{aV} + q_{aV} \times q_{bv}) \tag{E.3}$$

donde $\otimes, \cdot, \times$ describen el producto de cuaterniones, el producto escalar, y el producto vectorial respectivamente. La suma de cuaterniones es asociativa y conmutativa. La multiplicación de cuaterniones es asociativa y distributiva sobre la suma, pero no es conmutativa.

E.2.2.3 Conjugado de cuaterniones

$$q^* = (q_s, -q_v) = (q_s, -q_1, -q_2, -q_3) \tag{E.4}$$

E.2.2.4 Norma de cuaterniones

$$\| q \|^2 = q \otimes q^* = q_s^2 + q_1^2 + q_2^2 + q_3^2 \tag{E.5}$$

E.2.2.5 Inversa de cuaterniones

$$q^{-1} = \frac{1}{\| q \|^2} q^* \quad y \quad \| q \| \neq 0 \tag{E.6}$$

donde en $\| q \|^2 = 1$ se tiene el cuaternión unidad. Cualquier cuaternión q puede ser normalizado dividiendo por su norma. Para el cuaternión unidad se tiene:

$$q^{-1} = q^* \tag{E.7}$$

El cuaternión unidad puede ser definido como operador de rotación Mukundan 2002; Hart, Fracis y Kaauffman 1994; Tan y Balchen 1993. La rotación sobre el eje "n" un ángulo θ puede ser expresado utilizando el cuaternión unidad como sigue:

$$q = \begin{bmatrix} cos(\frac{\theta}{2}) & sin(\frac{\theta}{2})n \end{bmatrix} \tag{E.8}$$

Teorema E.2.1 *Siendo q_a y q_b dos cuaterniones puros, el producto de estos dos cuaterniones es:*

$$\begin{aligned} q_a \otimes q_b = (q_{aS}q_{bS} - q_{aV} \cdot q_{bV}), (q_{aS}q_{bV} + q_{bS}q_{aV} + q_{aV} \times q_{bv}) = \\ = (-q_{aV} \cdot q_{bV}), (q_{bS}q_{aV} + q_{aV} \times q_{bV}) \end{aligned} \tag{E.9}$$

Entonces, se definen dos nuevas funciones utilizando el producto de dos cuaterniones puros:

$$\begin{aligned} V\{q_a \otimes q_b\} = q_{aV} \times q_{bV} \rightarrow \text{parte vectorial del producto de cuaterniones} \\ S\{q_a \otimes q_b\} = -(q_{aV} \cdot q_{bV}) \rightarrow \text{parte escalar del producto de cuaterniones} \end{aligned} \tag{E.10}$$

E.2.3 Cuaterniones duales

El "dual-quaternion" o cuaternión dual se representa en la forma:

$$\hat{q} = (\hat{q}_s, \hat{q}_v) \quad or \quad \hat{q} = q + \varepsilon q^0 \tag{E.11}$$

donde $\hat{q}_s = q_s + \varepsilon q_s^0$ es un escalar dual, $\hat{q}_v = q_v + \varepsilon q_v^0$ es un vector dual, q y q^0 son dos cuaterniones y ε es el factor dual Han, Wei y Li 2008.

E.2.4 Operaciones con cuaterniones duales

E.2.4.1 Suma de cuaterniones duales

$$\hat{q}_a + \hat{q}_b = (q_a + q_b) + \varepsilon(q_a^0 + q_b^0) \tag{E.12}$$

E.2.4.2 Producto de cuaterniones duales

$$\hat{q}_a \odot \hat{q}_b = (q_a \otimes q_b) + \varepsilon(q_a \otimes q_b^0 + q_a^0 \otimes q_b^0) \tag{E.13}$$

donde $\otimes, \odot$ describen el producto de cuaterniones y el producto de cuaterniones duales respectivamente. La multiplicación de cuaterniones duales es asociativa, y distributiva sobre la suma de cuaterniones duales pero no es conmutativa.

E.2.4.3 Conjugado de cuaterniones duales

$$\hat{q}^* = q^* + \varepsilon(q^0)^* \tag{E.14}$$

E.2.4.4 Norma de cuaterniones duales

$$\| \hat{q} \|^2 = \hat{q} \odot \hat{q}^* \tag{E.15}$$

E.2.4.5 Inversa de cuaterniones duales

$$\hat{q}^{-1} = \frac{1}{\| \hat{q} \|^2} \hat{q}^* \quad y \quad \| \hat{q} \| \neq 0 \tag{E.16}$$

cuando $\| \hat{q} \|^2 = 1$ se tiene el cuaternión dual unidad. Para los cuaterniones duales unidad se tiene:

$$\begin{gathered} \| \hat{q} \|^2 = \hat{q} \odot \hat{q}^* = \hat{q}^* \odot \hat{q} = 1 \\ q \otimes q^* = 1, \qquad q^* \otimes q^0 + (q^0)^* \otimes q = 0 \end{gathered} \tag{E.17}$$

El cuaternión dual unidad puede usarse como operador de transformación para un cuerpo rígido Han, Wei y Li 2008. A pesar de que cuenta con ocho parámetros y es no mínima, es el operador más compacto y eficiente, Funda y Paul 1990; Funda, Taylor y Paul 1990. Esta transformación es muy similar a una rotación pura, sin embargo, no para un punto pero si para una línea. Una línea en coordenadas de "Plücker" $L_a(m,d)$, ($\hat{l}_a = l_a + \varepsilon m_a \rightarrow$ en cuaterniones duales), puede ser transformada a $L_b(m,d)$ utilizando el cuaternión dual unidad como sigue:

$$\hat{l}_b = \hat{q} \odot \hat{l}_a \odot \hat{q}^* \tag{E.18}$$

donde $\hat{q}$ es el cuaternión dual unidad.

Teorema E.2.2 *Siendo $\hat{q}_a$ y $\hat{q}_b$ dos cuaterniones duales, el producto de estos dos cuaterniones duales es:*

$$q_{ab} = q_a \odot q_b = q_{ab} + \varepsilon q^0_{ab} = (q_{abS}, q_{abV}) + \varepsilon(q^0_{abS}, q^0_{abV}) \tag{E.19}$$

Entonces se puede definir 4 nuevas funciones utilizando el producto de dos cuaterniones duales y el teorema E.2.1:

$$\begin{gathered} S\{R\{\hat{q}_a \odot \hat{q}_b\}\} = q_{abS} \rightarrow (1) \\ S\{D\{\hat{q}_a \odot \hat{q}_b\}\} = q^0_{abS} \rightarrow (2) \\ V\{R\{\hat{q}_a \odot \hat{q}_b\}\} = q_{abV} \rightarrow (3) \\ V\{D\{\hat{q}_a \odot \hat{q}_b\}\} = q^0_{abV} \rightarrow (4) \end{gathered} \tag{E.20}$$

(1) parte escalar de la parte real del producto de CD
(2) parte escalar de la parte dual del producto de CD
(3) parte vectorial de la parte real del producto de CD
(4) parte vectorial de la parte dual del producto de CD

donde CD significa cuaterniones duales.

E.3 Teoría del tornillo o "screw theory"

Los conocimientos de la teoría del tornillo se remontan al trabajo de "Chasles" y "Poinsot" a principios de 1800. De acuerdo con "Chasles", todos los movimientos propios del cuerpo rígido en el espacio de tres dimensiones, con la excepción de la translación pura, son equivalentes a los movimientos de "screw", esto es, la rotación sobre la línea junto con la translación sobre la línea Selig 2004. Si la línea pasa sobre el origen, el movimiento de "screw" puede ser escrito como:

$$T = \begin{bmatrix} R(\theta, d) & \frac{\theta}{2\pi} pd \\ 0 & 1 \end{bmatrix} \tag{E.21}$$

donde $R(\theta, d)$ representa la matriz de rotación de $3x3$ sobre el eje en la dirección del vector unidad d a través de un ángulo θ. El número p, que es llamado paso de "screw" (the pitch of the screw), es la distancia recorrida a lo largo del eje para una vuelta completa alrededor del eje. Si el eje del movimiento de "screw" no pasa sobre el origen como muestra la siguiente figura, Figura E.1, se puede escribir el movimiento de "screw" como:

$$T = \begin{bmatrix} I_{3x3} & p \\ 0 & 1 \end{bmatrix} \begin{bmatrix} R(\theta, d) & \frac{\theta}{2\pi} pd \\ 0 & 1 \end{bmatrix} \begin{bmatrix} I_{3x3} & -p \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R(\theta, d) & \frac{\theta}{2\pi} pd + (I_{3x3} - R(\theta, d))p \\ 0 & 1 \end{bmatrix} \tag{E.22}$$

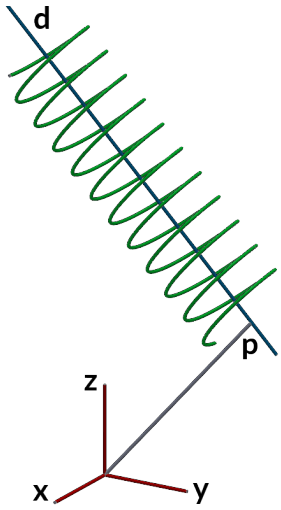


Figura E.1: Movimiento general de SCREW.

E.3.1 Movimiento SCREW utilizando cuaterniones duales

Se va a separar la formula general del movimiento de "screw" que se da en la ecuación E.21 como rotación y translación. La parte de rotación será igual que $R(\theta, d)$ y la parte de translación será igual a $t = \frac{\theta}{2\pi} pd + (I_{3x3} - R(\theta, d))p$.

El operador de movimiento de "screw" general puede ser representado utilizando los cuaterniones duales como sigue:

$$\hat{q} = cos\left(\frac{\hat{\theta}}{2}\right) + sin\left(\frac{\hat{\theta}}{2}\right)\hat{d} \tag{E.23}$$

$$\hat{q} = \begin{bmatrix} q_s \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} + \varepsilon \begin{bmatrix} q_s^0 \\ q_1^0 \\ q_2^0 \\ q_3^0 \end{bmatrix} = \begin{bmatrix} cos\left(\frac{\theta}{2}\right) \\ sin\left(\frac{\theta}{2}\right) d_x \\ sin\left(\frac{\theta}{2}\right) d_y \\ sin\left(\frac{\theta}{2}\right) d_z \end{bmatrix} + \varepsilon \begin{bmatrix} -\frac{d}{2} sin\left(\frac{\theta}{2}\right) \\ sin\left(\frac{\theta}{2}\right) d_x^0 + \frac{d}{2} cos\left(\frac{\theta}{2}\right) d_x \\ sin\left(\frac{\theta}{2}\right) d_y^0 + \frac{d}{2} cos\left(\frac{\theta}{2}\right) d_y \\ sin\left(\frac{\theta}{2}\right) d_z^0 + \frac{d}{2} cos\left(\frac{\theta}{2}\right) d_z \end{bmatrix} \tag{E.24}$$

donde $\hat{\theta} = \theta + \varepsilon k$ y $\hat{d} = d + \varepsilon m$ son números duales.

Aquí, θ y $d = [0, d]$ indican el ángulo de rotación y el eje del movimiento de "screw" respectivamente. $m = [0, p \times d]$ indica el momento del vector de rotación del eje. Donde p es cualquier punto en la dirección del vector d y $k = d \cdot t$. Se pueden encontrar más detalles sobre la formulación del movimiento de "screw" general utilizando cuaterniones duales en Daniilidis 1999.

E.4 Coordenadas de "PLÜCKER"

Cada línea puede ser completamente representada por un conjunto ordenado de dos vectores. El primer punto es un vector (p) que indica la posición de un punto arbitrario de la recta, y el segundo es el vector director (d) que nos proporciona la dirección de la línea. Las coordenadas de "Plücker" pueden ser representadas como:

$$L_p(m, d) \tag{E.25}$$

donde $m = p \times d$ es el momento del vector de (d) respecto al origen de referencia elegido. Tener en cuenta que m es independiente de qué punto p es elegido sobre la línea:

$$m = p \times d = (p + td) \times d \tag{E.26}$$

Los dos vectores tridimensionales m y d son siempre ortogonales.

$$d \cdot m = 0 \tag{E.27}$$

Tal y como se ha comentado anteriormente, una línea en coordenadas de "Plücker" puede representarse utilizando cuaterniones duales de la siguiente forma:

$$\hat{l}_p = l_p + \epsilon m_a \tag{E.28}$$

y esta puede ser transformada a $L_k(m, d)$ utilizando el cuaternión dual unidad:

$$\hat{l}_k = \hat{q} \otimes \hat{l}_p \otimes \hat{q}^* \tag{E.29}$$

donde $\hat{q}$ es el cuaternión dual unidad.

La representación de las coordenadas de "Plücker" es no mínima ya que utiliza seis parámetros para la representación de la línea. La ventaja principal de la representación de las coordenadas de "Plücker" es que es homogénea. $L_p(m, d)$ representa la misma línea que $L_p(km, kd)$ donde $k \in \Re$.

E.5 Números duales

El número dual fue introducido originalmente por "Clifford" en 1873, Brodsky y Shoham 2002; Gu y Luh 1987. En analogía con un número complejo, el número dual puede ser definido como:

$$\hat{u} = u + \varepsilon u^0 \tag{E.30}$$

donde u y u^0 son números reales y $\varepsilon = 0$. Los números duales pueden ser utilizados para expresar las coordenadas de "Plücker" propuestas por:

$$\hat{u} = d + \varepsilon m \tag{E.31}$$

donde d y $m = p \times d$ son la orientación y el vector del momento de la línea respectivamente.

E.6 Intersección de dos vectores ortogonales

La intersección de dos líneas ortogonales, dadas por dos vectores, puede observarse en la figura E.2.

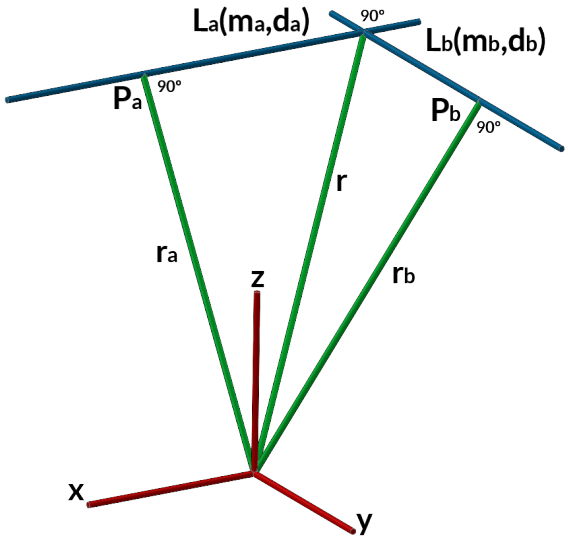


Figura E.2: Intersección de dos líneas.

$$L_a(m_a, d_a) \qquad L_b(m_b, d_b) \tag{E.32}$$

donde L_a y L_b son las dos líneas que forman una intersección ortogonal.

Las ecuaciones, que se muestran a continuación pueden ser escritas como dos vectores de intersección ortogonales:

$$r_a = m_a \times d_a \quad y \quad r_b = m_b \times d_b \tag{E.33}$$

$$r = r_a + \alpha d_a = r_b + \beta d_b. \tag{E.34}$$

$$\alpha d_a = (r_b - r_a) + \beta d_b \quad o \quad \beta d_b = (r_a - r_b) + \alpha d_a \tag{E.35}$$

donde $\alpha, \beta \in \Re$.

Si se multiplica la primera y segunda ecuación en (E.35) por d_a y d_b respectivamente, entonces se obtiene:

$$\begin{aligned} \alpha &= (r_b - r_a) \cdot d_a + \beta d_b \cdot d_a = r_b \cdot d_a \\ \beta &= (r_a - r_b) \cdot d_b + \beta d_a \cdot d_b = r_a \cdot d_b \end{aligned} \tag{E.36}$$

Notar que: $r_a \cdot d_a = r_b \cdot d_b = d_a \cdot d_b = 0$ porque son ortogonales.

Por lo tanto, se puede encontrar el punto de intersección entre dos líneas:

$$\begin{aligned} r &= d_b \times m_b + ((d_a \times m_a) \cdot d_b) d_b \\ r &= d_a \times m_a + ((d_b \times m_b) \cdot d_a) d_a \end{aligned} \tag{E.37}$$

E.7 Sub-problemas de "Paden-Kahan" utilizando el álgebra de cuaterniones

E.7.1 Sub-problema 1: rotación sobre un eje simple

En el sub-problema 1, un punto α rota sobre el eje l hasta que el punto a es coincidente con el punto b. Esta rotación es mostrada en la figura E.3.

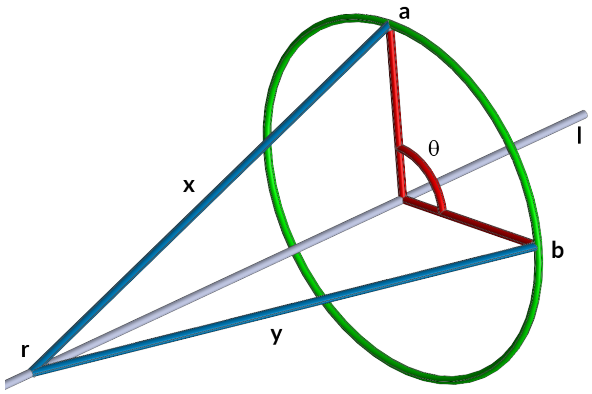


Figura E.3: Rotación de a sobre el eje l hasta ser coincidente con b.

donde r es un punto sobre el eje l, $x = a - r$ y $y = b - r$ son dos vectores. La fórmula general de rotación para el movimiento puede escribirse como:

$$y = q \otimes x \otimes q^* \tag{E.38}$$

donde $q = \left[cos\left(\frac{\theta}{2}\right) \quad sin\left(\frac{\theta}{2}\right) l\right]$, $x = [0, x]$ y $y = [0, y]$ son la forma del cuaternión del operador de rotación, vectores x e y respectivamente.

El ángulo de rotación sobre el eje l puede ser encontrado como sigue:

$$\theta = arctan2(S\{l \otimes x' \otimes y'\}, S\{x' \otimes y'\}) \tag{E.39}$$

donde:

$$x' = x + S\{l \otimes x\}l \tag{E.40}$$

$$y' = q \otimes x \otimes q^* + S\{l \otimes q \otimes x \otimes q^*\}l = y + S\{l \otimes y\}l \tag{E.41}$$

donde $l = [0, l]$ es la forma del cuaternión del vector director de l.

E.7.2 Sub-problema 2: rotación sobre dos ejes que se cruzan

En el sub-problema 2, primero un punto a rota sobre el eje l_1 un ángulo θ_1 seguido de un giro sobre el eje l_2 de θ_2, entonces la posición final de a coincide con la del punto b. La rotación es mostrada en la figura E.4.

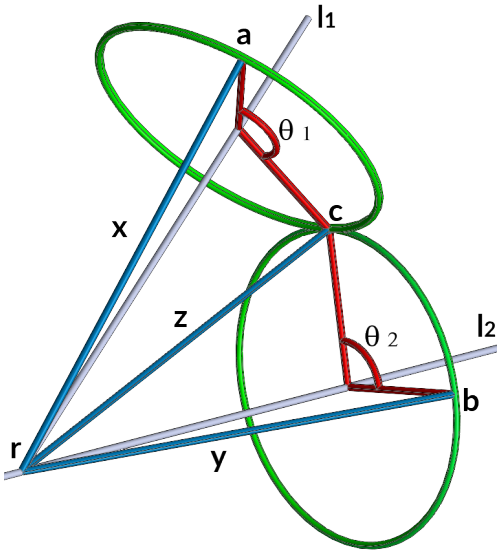


Figura E.4: Rotación de a sobre el eje l_1 seguido de una rotación sobre el eje l_2 hasta ser coincidente con el punto b.

Los dos ejes se deben de cortar en este sub-problema. Si los dos ejes son coincidentes, entonces este problema se reduce al sub-problema 1. Si los dos ejes no son paralelos, entonces $l_1 \times l_2 \neq 0$. Entonces r es el punto intersección de los dos ejes, $x = a - r$ e $y = b - r$ son dos vectores. La fórmula general de rotación para el movimiento puede ser escrita por:

$$y = q_1 \otimes q_2 \otimes x \otimes q_2^* \otimes q_1^* \tag{E.42}$$

donde $q_1 = \left[cos\left(\frac{\theta_1}{2}\right) \quad sin\left(\frac{\theta_1}{2}\right) l_1\right]$, $q_2 = \left[cos\left(\frac{\theta_2}{2}\right) \quad sin\left(\frac{\theta_2}{2}\right) l_2\right]$, $x = [0, x]$ y $y = [0, y]$ son la forma del cuaternión del operador de rotación, vectores x e y respectivamente. El punto c es el punto intersección de la rotación que se puede observar en la figura E.4 y $z = c - r$, $z = [0, z]$ es el vector que se define entre el punto c y r y la forma del cuaternión puro del vector z. Se puede definir las dos rotaciones por:

$$q_1 \otimes x \otimes q_1^* = z = q_3 \otimes y \otimes q_3^* \tag{E.43}$$

Que es el sub-problema 1 y donde $q_3 = \left[cos\left(-\frac{\theta_2}{2}\right) \quad sin\left(-\frac{\theta_2}{2}\right) l_2\right]$ es el cuaternión de rotación que rota sobre el eje l_2 un ángulo $-\theta_2$.

Si l_1, l_2 y $l_1 \times l_2$ son linealmente independientes, se puede escribir:

$$z = \alpha l_1 + \beta l_2 + \gamma \begin{bmatrix} 0 & V\{l_1 \otimes l_2\} \end{bmatrix} \tag{E.44}$$

donde:

$$\alpha = \frac{S\{l_1 \otimes l_2\} S\{l_2 \otimes x\} - S\{l_1 \otimes y\}}{\left(S\{l_1 \otimes l_2\}\right)^2 - 1} \tag{E.45}$$

$$\beta = \frac{S\{l_1 \otimes l_2\} S\{l_1 \otimes y\} - S\{l_2 \otimes x\}}{\left(S\{l_1 \otimes l_2\}\right)^2 - 1} \tag{E.46}$$

$$\gamma^2 = \frac{\| x \|^2 - \alpha^2 - \beta^2 - 2\alpha\beta S\{l_1 \otimes l_2\}}{\| V\{l_1 \otimes l_2\} \|^2} \tag{E.47}$$

Con lo que ahora el sub-problema 2 es reducido al sub-problema 1. Los ángulos de rotación de los ejes θ_1 y θ_2 pueden ser resueltos utilizando el sub-problema 1:

$$\begin{aligned} q_1 \otimes x \otimes q_1^* &= z \\ q_3 \otimes y \otimes q_3^* &= z \end{aligned} \tag{E.48}$$

E.7.3 Sub-problema 3: rotación a una distancia determinada

En el sub-problema 3, el punto a rota sobre el eje l hasta que el punto está a una distancia δ de b como se muestra en la figura E.5.

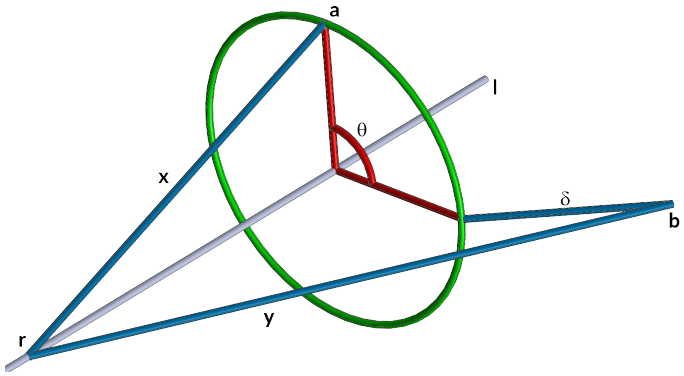


Figura E.5: Rotación de a sobre el eje l hasta estar a una distancia δ de b.

Considerar la proyección de todos los puntos en el plano perpendicular a l, sea r un punto en el eje l y define $x = a - r = [0, x]$ = y $y = b - r = [0, y] = [0, y - r]$ donde x, y, a, b, r son cuaterniones puros de los vectores x, y, a, b, r respectivamente. La ecuación general de este movimiento puede escribirse por:

$$\| y - q \otimes x \otimes e^* \| = \| \delta \| \tag{E.49}$$

donde $q = \left[cos\left(\frac{\theta}{2}\right) \quad sin\left(\frac{\theta}{2}\right) l\right]$, es el cuaternión del operador de rotación. Las proyecciones de x, y, δ son:

$$x' = \begin{bmatrix} 0 & x' \end{bmatrix} = x + S\{l \otimes x\} l \tag{E.50}$$

$$y' = \begin{bmatrix} 0 & y' \end{bmatrix} = y + S\{l \otimes y\} l = q \otimes x \otimes q^* + S\{l \otimes q \otimes x \otimes q^*\} \tag{E.51}$$

$$\delta'^2 = \delta^2 + \mid S\{l \otimes (a - b)\} \mid^2 \tag{E.52}$$

donde el ángulo θ_0 es el ángulo entre los vectores x' e y', se tiene:

$$\theta_0 = arctan2(S\{l \otimes x' \otimes y'\}, S\{x' \otimes y'\}) \tag{E.53}$$

por lo tanto:

$$\theta = \theta_0 \pm cos^{-1}\left(\frac{\| x' \|^2 + \| y' \|^2 - \delta'^2}{2 \| x' \| \| y' \|}\right) \tag{E.54}$$

E.8 Cinemática

E.8.1 Cinemática directa

El problema de la cinemática directa consiste en determinar la posición y orientación del efector final dados los valores de las variables de articulación de la cadena cinemática. Para encontrar la cinemática directa de una cadena cinemática se tienen que seguir los siguientes pasos:

E.8.1.1 Notación

- Etiquetar las articulaciones y la intersección de una articulación con la anterior: las articulaciones son numeradas del 1 hasta n, empezando por la base, y los puntos de rotación son numerados desde el número 0 al n.
- Configuración de los espacios de revolución: para las articulaciones de revolución, el movimiento rotacional se describe sobre el eje y todos los ángulos de articulación son calculados utilizando el sistema de coordenadas de la "mano derecha". Para las articulaciones prismáticas, el desplazamiento lineal es descrito a lo largo de la dirección del eje.
- Colocación de los sistemas de coordenadas: dos sistemas de coordenadas son necesarios para las cadenas cinemáticas de cadena abierta de n grados de libertad. Estos sistemas de coordenadas son llamados sistema de coordenadas de la base y sistema de coordenadas de la herramienta. El sistema de coordenadas de la base puede ser colocado arbitrariamente pero en general es colocado directamente en la articulación 1 y el sistema de coordenadas de la herramienta es colocado en el efector final de la cadena cinemática.

E.8.1.2 Formulación

- Determinar los ejes de articulación y los momentos de los vectores: primeramente, los vectores de los ejes que describen el movimiento de las articulaciones son colocados. Entonces, el momento de los vectores de estos ejes son obtenidos para movimientos de revolución. Por lo tanto, son obtenidas las coordenadas de "Plücker" para estos ejes.
- Obtención del operador de transformación: para todas las articulaciones, los operadores de transformación de los cuaterniones duales pueden ser obtenidos como sigue:

$$\hat{q}_i = (\hat{q}_{Si}, \hat{q}_{Vi}) \quad o \quad \hat{q}_i = q_i + \varepsilon q_i^0 \tag{E.55}$$

donde para las articulaciones prismáticas se tiene:

$$\begin{matrix} q_i = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} \\ q_i^0 = \begin{bmatrix} 0 & q_1^0 & q_2^0 & q_3^0 \end{bmatrix} \end{matrix} \tag{E.56}$$

y para las articulaciones de revolución:

$$\begin{array}{cc} q_i = \left[cos\left(\frac{\theta_i}{2}\right) \quad sin\left(\frac{\theta_i}{2}\right) d_i\right] \\ q_i^0 = \frac{1}{2}(p_i - q_i \otimes p_i \otimes q_i^*) \otimes q_i \quad o \quad q_i = \left[0 \quad sin\left(\frac{\theta_i}{2}\right) m_i\right] \end{array} \tag{E.57}$$

donde $i = 1, 2, ..., n$.

- Formulación del movimiento rígido: puede ser obtenida utilizando la ecuación (E.18). Para una cadena cinemática de n grados de libertad, la operación de transformación del cuerpo rígido general es dada por:

$$\hat{q}_{1n} = \hat{q}_1 \odot \hat{q}_2 \odot ... \odot \hat{q}_n \tag{E.58}$$

donde $\hat{q}_{1n} = q_{1n} + \varepsilon q_{1n}^0$.

La orientación y la posición del efector final pueden ser encontradas como sigue:

Que $\hat{l}_n = l_n + \varepsilon l_n^0$ y $\hat{l}_{n-1} = l_n + \varepsilon l_{n-1}^0$ son las representaciones de las coordenadas de "Plücker" n^{th} y $(n-1)^{th}$ respectivamente. Y también, que $\hat{l}_n^{'} = l_n^{'} + \varepsilon l_n^{'0} = \hat{q}_{1n} \odot \hat{l}_n \odot \hat{q}_{1n}^*$ y $\hat{l}_{n-1}^{'} = l_{n-1}^{'} + \varepsilon l_{n-1}^{'0} = \hat{q}_{1n-1} \odot \hat{l}_{n-1} \odot \hat{q}_{1n-1}^*$ son las representaciones de las coordenadas de "Plücker" después de la transformación. La orientación del efector final es $\hat{l}_6^{'}$. La posición del efector final puede ser encontrada utilizando las ecuaciones (E.10), (E.20) y (E.37):

$$\begin{array}{c} p_n = (V\{R\{\hat{q}_{1n} \odot \hat{l}_n \odot \hat{q}_{1n}^*\}\}) \times (V\{D\{\hat{q}_{1n} \odot \hat{l}_n \odot \hat{q}_{1n}^*\}\}) + \\ + \left(\begin{pmatrix} (V\{R\{\hat{q}_{1n-1} \odot \hat{l}_{n-1} \odot \hat{q}_{1n-1}^*\}\}) \times \\ (V\{D\{\hat{q}_{1n-1} \odot \hat{l}_{n-1} \odot \hat{q}_{1n-1}^*\}\}) \end{pmatrix} \cdot (V\{R\{\hat{q}_{1n} \odot \hat{l}_n \odot \hat{q}_{1n}^*\}\})\right) * \\ *(V\{R\{\hat{q}_{1n} \odot \hat{l}_n \odot \hat{q}_{1n}^*\}\}) \end{array} \tag{E.59}$$

E.8.2 Cinemática inversa

El problema de la cinemática inversa consiste en determinar los valores de las variables de articulación dada la posición y orientación del efecto final. Se utilizarán los sub-problemas de "Paden-Kahan" para obtener la solución a la cinemática inversa de las cadenas cinemáticas. Se tienen varios sub-problemas de "Paden-Kahan" y también nuevos sub-problemas extendidos Murray, Li y S.S. 1994; Paden 1986; Yue-sheng y Ai-ping 2008. Solo se utilizarán tres de estos sub-problemas que son los que más comúnmente aparecen en la cinemática inversa, estos tres problemas han sido vistos anteriormente. Para resolver el problema de la cinemática inversa, se va a reducir todo el problema de la cinemática inversa

en cada uno de los sub-problemas correspondientes.La solución a la cinemática inversa se encuentra en el capítulo 5.

Bibliografía

Agarwal, A. y B. Triggs (2004a). "3D Human Pose from Silhouettes by Relevance Vector Regresion". En: *IEEE Computer Vision and Pattern Recognition* (vid. pág. 146).

— (2004b). "3D Human Pose from Silhouettes by Relevance Vector Regression". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, USA; pp. 882-888* (vid. pág. 20).

— (2004c). "Learning to Track 3D Human Motion from Silhouettes". En: *Proceedings of the International Conference on Machine Learning, Banff, AB, Canada; pp. 9-16* (vid. pág. 20).

— (2004d). "Tracking Articulated Motion Using a Mixture of Autoregressive Models". En: *Proceedings of the European Conference on Computer Vision, Prague, Czech Republic; pp. 54-65* (vid. pág. 32).

— (2005). "Monocular Human Motion Capture with a Mixture of Regressors". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA; p. 72* (vid. pág. 25).

— (2006a). "A Local Basis Representation for Estimating Human Pose from Cluttered Images". En: *Proceedings of the Asian Conference on Computer Vision, Hyderabad, India; pp. 50-59* (vid. págs. 11, 21).

Agarwal, A. y B. Triggs (2006b). "Hyperfeatures-Multilevel Local Coding for Visual Recognition". En: *Proceedings of the European Conference on Computer Vision, Graz, Austria; pp. 30-43* (vid. pág. 11).

— (2006c). "Recovery of 3D Human Pose from Monocular Images". En: *IEEE Trans. Pattern Analysis and Machine Intelligence, 28(1). pp 44-58* (vid. págs. 10, 20, 21, 30, 141, 146).

Aggarwal, J.K. y Q. Cai (1995). "Nonrigid Motion Analysis: Articulated and Elastic Motion". En: *Computer Vision and Image Understanding. Vol 70(2). pp 142-156* (vid. pág. 128).

— (1999). "Human Motion Analysis: A Review". En: *Computer Vision and Image Understanding. Vol 73(3). pp 428-440.* (Vid. pág. 128).

Aggarwal, J.K. y M.S. Ryoo (2011). "Human Activity Analysis: A Review". En: *ACM Comput. Surv., 43, 16.* (Vid. pág. 9).

Akhter, I. y M.J. Black (2015). "Pose-Conditioned Joint Angle Limits for 3D Human Pose Reconstruction". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA; pp. 1446-1455* (vid. págs. 14, 21).

Allen, B., B. Curless y Z. Popovic (2002a). "Articulated Body Deformation from Range Scan Data". En: *ACM Trans. Graph., 21, 612-619* (vid. pág. 17).

— (2002b). "Articulated body deformation from range scan data". En: *Proc. ACM SIGGRAPH, pages 612-619* (vid. pág. 131).

Amit, Y. y D. Geman (1997). "Shape Quantization and Recognition with Randomized Trees". En: *Neural Comput., 9, 1545-1588* (vid. pág. 23).

Amit, Y. y A. Trouve (2007). "POP: Patchwork of parts models for object recognition". En: *IJCV, 75(2):267-282* (vid. pág. 155).

Andriluka, M., S. Roth y B. Schiele (2008). "People-Tracking-by-Detection and People-Detection-by-Tracking". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA; pp. 1-8* (vid. pág. 28).

— (2009). "Pictorial Structures Revisited: People Detection and Articulated Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA; pp. 1014-1021* (vid. págs. 14, 28).

— (2010). "Monocular 3D Pose Estimation and Tracking by Detection". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA; pp. 623-630* (vid. págs. 14, 21).

Anguelov, D. y col. (2005). "SCAPE: Shape Completion and Animation of People". En: *ACM Trans. Graph., 24, 408-416* (vid. pág. 17).

Arkin, E.M. y col. (1991). "An Efficiently Computable Metric for Comparing Polygonal Shapes". En: *IEEE Trans. Pattern Anal. Mach. Intell., 13, 209-216* (vid. pág. 11).

Aspragathos, N.A. y J.K. Dimitros (1998). "A Comparative Study of Three Methods for Robot Kinematics". En: *IEEE Transactions on Systems Man and Cybernetics - Part B: Cybernetics, Vol.28,NO.2* (vid. pág. 112).

"Attractive People: Assembling Loose-Limbed Models Using Non-Parametric Belief Propagation" (2016). En: *Available online: http://machinelearning.wustl.edu/mlpapers/* (vid. pág. 32).

Baak, A. y col. (2013a). "A Data-Driven Approach for Real-Time Full Body Pose Reconstruction from a Depth Camera". En: *Consumer Depth Cameras for Computer Vision; Springer: London, UK; pp. 71-98* (vid. pág. 21).

— (2013b). "A Data-Driven Approach for Real-Time Full Body Pose Reconstruction from a Depth Camera". En: *Consumer Depth Cameras for Computer Vision; Springer: Heidelberg, Germany; pp. 71-98* (vid. pág. 24).

Babagholami Mohamadabadi, B. y col. (2014). "A Bayesian Framework for Sparse Representation Based 3D Human Pose Estimation". En: *IEEE Signal Process. Lett., 21, 297-300* (vid. págs. 23, 24).

Balan, A.O. y M.J. Black (2006). "An adaptive appearance model approach for model-based articulated object tracking". En: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, New York, NY, USA; pp. 758-765* (vid. pág. 32).

Balan, A.O. y col. (2007a). "Detailed Human Shape and Pose from Images". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA; pp. 1-8* (vid. pág. 17).

Balan, A.O. y col. (2007b). "Shining a Light on Human Pose: On Shadows, Shading and the Estimation of Pose and Shape". En: *Proceedings of the IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil; pp. 1-8* (vid. pág. 17).

Ball R, S. (1900). "The Theory of Screws". En: *Cambridge, U.K. Cambridge Univ.Press* (vid. pág. 202).

Barbulescu, A. y col. (2012). "3D Human Pose Estimation Using 2D Body Part Detectors". En: *Proceedings of the International Conference on Pattern Recognition, Tsukuba, Japan; pp. 2484-2487* (vid. pág. 21).

Barron, C. e I.A. Kakadiaris (2003). "On the improvement of anthropometry and pose estimation from a single uncalibrated image". En: *Machine Vision Applications, 14.* (Vid. pág. 130).

Baumberg, A. y D. Hogg (1994). "Learning Flexible Models from Image Sequences". En: *Proceedings of the European Conference on Computer Vision, Stockholm, Sweden; pp. 297-308* (vid. pág. 16).

Belagiannis, V. y col. (2014a). "3D Pictorial Structures for Multiple Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA; pp. 1669-1676* (vid. pág. 28).

Belagiannis, V. y col. (2014b). "Holistic Human Pose Estimation with Regression Forests". En: *Proceedings of the International Conference on Articulated Motion and Deformable Objects, Palma de Mallorca, Spain; pp. 20-30* (vid. pág. 23).

Bera, A. y col. (2016). "GLMP - realtime pedestrian path prediction using global and local movement patterns". En: *Robotics and Automation (ICRA) 2016 IEEE International Conference on, pp. 5528-5535, 2016* (vid. pág. 133).

Bissacco, A., M. Yang y S. Soatto (2006). "Detecting Humans via Their Pose". En: *Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada; pp. 169-176* (vid. pág. 30).

Blank, B. y col. (2005). "Actions as space-time shapes". En: *Proc. International Conference on Computer Vision, (ICCV05), pp 1395-1402* (vid. pág. 149).

Blinn, J.F. (1977). "Models of Light Reflection for Computer Synthesized Pictures". En: *ACM SIGGRAPH Comput. Graph, 11, 192-198* (vid. pág. 17).

Bo, L. y C. Sminchisescu (2009). "Structured Output-Associative Regression". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA; pp. 2403-2410* (vid. pág. 21).

— (2010). "Twin Gaussian Processes for Structured Prediction". En: *Int. J. Comput. Vis., 87, 28-52* (vid. pág. 20).

Bo, Y. y H. Jiang (2013). "Scale and Rotation Invariant Approach to Tracking Human Body Part Regions in Videos". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Portland, OR, USA; pp. 1041-1047* (vid. pág. 28).

Bobick, A. (1997). "Movement, activity and action: The role of knowledge in the perception of motion". En: *Philosophical Trans. Royal Soc. London, 352. pp 1257-1265* (vid. pág. 148).

Bordes, A. y col. (2007). "Solving multiclass support vector machines with larank". En: *International Conference on Machine Learning* (vid. pág. 47).

Bosch, M., A. J. Sanchez y C. Ricolfe-Viala (2012). "Visual-based human action recognition on smart phones based on 2D and 3D descriptors". En: *International Journal of Pattern Recognition and Artificial Intelligence, 26(08), 1260009* (vid. pág. 10).

Bosch, M. y col. (2014). "Fall detection based on the gravity vector using a wide-angle camera". En: *Expert Systems with Applications, 41(17), 7980-7986* (vid. pág. 10).

Bourdev, L. y J. Malik (2009). "Poselets: Body part detectors trained using 3d human pose annotations". En: *IEEE Conference on Computer Vision and Pattern Recognition* (vid. pág. 48).

Bourdev, L. y col. (2010). "Detecting People Using Mutually Consistent Poselet Activations". En: *Proceedings of the European Conference on Computer Vision, Crete, Greece; pp. 168-181* (vid. pág. 12).

Bowden, R., T.A. Mitchell y M. Sarhadi (2000). "Non-linear Statistical Models for the 3D Reconstruction of Human Pose and Motion from Monocular Image Sequences". En: *Image Vis. Comput., 18, 729-737* (vid. pág. 19).

Bradski, G.R. y J.W. Davis (2002). "Motion Segmentation and Pose Recognition with Motion History Gradients". En: *Machine Vision and Applications, 13(3). pp 174-184* (vid. págs. 132, 133).

Brand, M. (1999). "Shadow Puppetry". En: *ICCV99, vol 2, pp.1237-1244 Corfu, Greece* (vid. pág. 141).

Bray, M., P. Kohli y P.H. Torr (2006). "Posecut: Simultaneous Segmentation and 3D Pose Estimation of Humans Using Dynamic Graph-Cuts". En: *Proceedings of the European Conference on Computer Vision, Graz, Austria; pp. 642-655* (vid. págs. 30, 31).

Bregler, C. y J. Malik (1998). "Tracking People with Twists and Exponential Maps". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Santa Barbara, CA, USA; pp. 8-15* (vid. págs. 13, 32).

Bregler, C., J. Malik y K. Pullen (2004). "Twist based acquisition and tracking of animal and human kinematics". En: *International Journal of Computer Vision, 56(3), pp. 179-194* (vid. págs. 24, 144).

Breiman, L. (2001). "Random Forests". En: *Mach. Learn., 45, 5-32.* (Vid. pág. 23).

Brodsky, V. y M. Shoham (2002). "Dual Numbers Representation of Rigid Body Dynamics". En: *PhD thesis Department of Mechanical Engineering Technion-Israel Institute of Technology* (vid. pág. 209).

Brostow, G.J. y col. (2004). "Novel Skeletal representation for Articulated Creatures". En: *Int. Proceedings of European Conference on Computer Vision, Vol III, pages 66-78.* (Vid. pág. 130).

Brox, T., B. Rosenhahn y J. Weickert (2005). "Three-Dimensional Shape Knowledge for Joint Image Segmentation and Pose Estimation". En: *Pattern Recognition; Springer: Berlin, Heidelberg, Germany; pp. 109-116* (vid. pág. 30).

Brubaker, M.A., D.J. Fleet y A. Hertzmann (2007). "Physics-Based Person Tracking Using Simplified Lower-Body Dynamics". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA; pp. 1-8* (vid. pág. 18).

— (2010). "Physics-Based Person Tracking Using the Anthropomorphic Walker". En: *Int. J. Comput. Vis., 87, 140-155* (vid. pág. 24).

Buntine, W. y T. Niblett (1992). "A Further Comparison of Splitting Rules for Decision-Tree Induction". En: *Mach. Learn., 8, 75-85* (vid. pág. 24).

Burl, M., M. Weber y P. Perona (1998). "A probabilistic approach to object recognition using local photometry and global geometry". En: *ECCV, pages II:628-641* (vid. pág. 155).

Buys, K. y col. (2014). "An Adaptable System for RGB-D Based Human Body Detection and Pose Estimation". En: *Vis. Commun. Image Represent, 25, 39-52.* (Vid. pág. 8).

Cai, Q., A. Mitiche y J.K. Agarwal (1995). "Tracking Human Motion in a Indoor Environment". En: *ICIP, pp 215-218* (vid. pág. 139).

Calinon, S., F. Guenter y A. Billard (2005). "Goal-directed imitation in a humanoid robot". En: *Proc IEEE Int Conf. on Robotics and Automation, ICRA05 pp 299-304. Barcelona, Spain* (vid. pág. 150).

Capellades, M.B. y col. (2003). "An Appearance Based Approach for Human and Object Tracking". En: *ICIP03, pp 14-17, Barcelona, Spain* (vid. pág. 138).

Carranza, J. y col. (2003). "Free-viewpoint video of human actors". En: *Proc. ACM SIGGRAPH. pp 565-577* (vid. págs. 130, 131, 143).

Carreira, J. y col. (2016). "Human Pose Estimation with Iterative Error Feedback". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA; pp. 4733-4742* (vid. pág. 23).

Cedras, C. y M. Shah (1995). "Motion Based Recognition: A Survey". En: *IEEE Proceedings of Image and Vision Computing Vol 13 (2). pp 129-155* (vid. pág. 128).

Chang, J.Y. y S.W. Nam (2013). "Fast Random-Forest-Based Human Pose Estimation Using a Multi-Scale and Cascade Approach". En: *ETRI J., 35, 949-959* (vid. pág. 23).

Chen, C., K. Liu y N. Kehtarnavaz (2016). "Real-time human action recognition based on depth motion maps". En: *Journal of Real-Time Image Processing, Volume 12, Issue 1, pp 155-163* (vid. pág. 13).

Chen, C. y col. (2011). "3D human pose recovery from image by efficient visual feature selection". En: *Comput. Vis. Image Underst., 115, 290-299* (vid. pág. 24).

Chen, L., H. Wei y J. Ferryman (2013). "A Survey of Human Motion Analysis Using Depth Imagery". En: *Pattern Recognit. Lett., 34, 1995-2006.* (Vid. pág. 9).

Chen, X. y A.L. Yuille (2014). "Articulated Pose Estimation by a Graphical Model with Image Dependent Pairwise Relations". En: *Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada; pp. 1736-1744* (vid. págs. 15, 23).

— (2015). "Parsing Occluded People by Flexible Compositions". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA; pp. 3945-3954* (vid. pág. 14).

Cheng, F., W.J. Christmas y J. Kittler (2002). "Recognising Human Running Behaviourin Sports Video Sequences". En: *International Conference on Pattern Recognition, Quebec, Canada. pp 11-15* (vid. págs. 142, 149).

Cheng, S.Y. y M.M. Trivedi (2004). "Human Posture Estimation Using Voxel Data for Smart Airbag Systems: Issues and Framework." En: *Proceedings of the IEEE Intelligent Vehicles Symposium, Parma, Italy; pp. 84-89.* (Vid. pág. 7).

Cherian, A. y col. (2014). "Mixing Body-Part Sequences for Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA; pp. 2353-2360* (vid. pág. 29).

Cheung, G., S. Baker y T. Kanade (2003). "Shape-from silhouette for articulated objects and its use for human body kinematics estimation and motion capture". En: *En Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR03), Madison, MI.* (Vid. págs. 31, 130).

Cheung, G.K. y col. (2004). "Markerless Human Motion Transfer". En: *Proceedings of the International Symposium on 3D Data Processing, Visualization and Transmission, Thessaloniki, Greece; pp. 373-378* (vid. pág. 18).

Christoudias, C.M. y T. Darrell (2005). "On Modelling Nonlinear Shape-and-Texture Appearance Manifolds". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA; pp. 1067-1074* (vid. pág. 21).

Chu, C.W., O.C. Jenkins y M.J. Mataric (2003). "Markerless Kinematic Model and Motion Capture from Volume Sequences". En: *Proc. IEEE Computer Vision and Pattern Recognition Vol 2. pp 475-482* (vid. págs. 133, 147).

Collins y col. (2000). "A System for Video Surveillance and Monitoring: VSAM Final Report". En: *Technical Report Technical report CMU-RI-TR-00-12, Robotics Institute, Carnegie Mellon University* (vid. pág. 133).

Comaniciu, D., V. Ramesh y P. Meer (2003). "Kernel-Based Object Tracking". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(5) pp 564-577* (vid. pág. 134).

Cootes, T.F. y col. (1995). "Active Shape Models-Their Training and Application". En: *Comput. Vis. Image Underst., 61, 38-59* (vid. pág. 16).

Cour, T. y col. (2008). "Movie/Script: Alignment and Parsing of Video and Text Transcription". En: *Proceedings of the European Conference on Computer Vision, Marseille, France; pp. 158-171* (vid. pág. 21).

Cour, T. y col. (2009). "Learning from Ambiguously Labeled Images". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA; pp. 919-926* (vid. pág. 21).

Crandall, D., P. Felzenszwalb y D. Huttenlocher (2005). "Spatial priors for part-based recognition using statistical models". En: *CVPR, pages 10-17* (vid. págs. 155, 156).

Cucchiara, R. y col. (2003). "Detecting Moving Objects, Ghosts, and Shadows in Video Streams". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(10) pp 1337-1342* (vid. pág. 132).

Cucchiara, R. y col. (2004). "Probabilistic People Tracking for Occlusion Handling". En: *International Conference on Pattern Recognition, Cambridge, UK, pp 23-26* (vid. pág. 137).

Dalal, N. y B. Triggs (2005a). "Histograms of Oriented Gradients for Human Detection". En: *Computer Vision and Pattern Recognition, San Diego, CA, USA. vol 1. pp 886-893* (vid. págs. 11, 135).

— (2005b). "Histograms of oriented gradients for human detection". En: *Computer Vision and Pattern Recognition, (CVPR). IEEE Computer Society Conference on, volume 1. IEEE, pp. 886-893* (vid. págs. 42, 51).

— (2005c). "Histograms of oriented gradients for human detection". En: *CVPR, pages I: 886-893* (vid. págs. 154-156, 158, 161).

Dalal, N., B. Triggs y C. Schmid (2006). "Human Detection Using Oriented Histograms of Flow and Appearance". En: *Proceedings of the European Conference on Computer Vision, Graz, Austria; pp. 428-441* (vid. pág. 12).

Daniilidis, K. (1999). "Hand-Eye Calibration Using Dual Quaternions". En: *The International Journal of Robotics Research Vol.18, No.3, March, pp.286-298* (vid. pág. 208).

Dantone, M. y col. (2013). "Pose Estimation Using Body Parts Dependent Joint Regressors". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 3041-3048* (vid. pág. 23).

Datta, A., Y. Sheikh y T. Kanade (2008). "Linear Motion Estimation for Systems of Articulated Planes". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA; pp. 1-8* (vid. pág. 31).

Davis, J. y H. Gao (2003). "Recognizing human action efforts: An adaptive three-mode pca framework". En: *Proc. Int. Conf. on Computer Vision, (ICCV03). Vol 2. pp 1463-1469* (vid. pág. 150).

Davis, James W. y Aaron F. Bobick (1997). "The Representation and Recognition of Human Movement Using Temporal Templates". En: *Proceedings Computer Vision and Pattern Recognition (CVPR-97). pp.928-934* (vid. pág. 133).

Davis, J.W. y S.R. Taylor (2002). "Analysis and Recognition of Walking Movements". En: *International Conference on Pattern Recognition, Quebec, Canada. Vol 1. pp 315-318* (vid. pág. 149).

Davis, L., V. Philomin y R. Duraiswami (2000). "Tracking Humans from a Moving Platform". En: *International Conference on Pattern Recognition, Barcelona, Spain. Vol 4. pp 171-178* (vid. pág. 135).

De Aguiar, E. y col. (2007). "Marker-Less Deformable Mesh Tracking for Human Shape and Motion Capture". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA; pp. 1-8* (vid. pág. 16).

Delamarre, Q. y O. Faugeras (2001). "3D Articulated Models and Multi-view Tracking with Physical Forces". En: *Computer Vision and Image Understanding, 81(3), pp 328-357* (vid. pág. 142).

Demirdjian, D., T. Ko y T. Darrell (2003a). "Constraining Human Body Tracking". En: *Proceedings of the IEEE International Conference on Computer Vision, Nice, France; pp. 1071-1078* (vid. págs. 14, 18).

— (2003b). "Constraining human body tracking". En: *Proc.IEEE Int.Conf.of Computer Vision. Vol 2. pp 1071-1078.* (Vid. pág. 130).

Deutscher, J., A. Blake e I. Reid (2000). "Articulated Body Motion Capture by Annealed Particle Filtering". En: *Computer Vision and Pattern Recognition, Hilton Head Island, South Carolina. Vol 2. pp 126-133* (vid. pág. 143).

Deutscher, J. e I. Reid (2005). "Articulated Body Motion Capture by Stochastic Search". En: *Int. J. Comput. Vis., 61, 185-205* (vid. pág. 24).

Dimitrijevic, M., V. Lepetit y P. Fua (2006). "Human Body Pose Detection Using Bayesian Spatio-Temporal Templates". En: *Comput. Vis. Image Underst., 104, 127-139.* (Vid. págs. 10, 30).

Dinh, D.L. y col. (2014). "Real-Time 3D Human Pose Recovery from a Single Depth Image Using Principal Direction Analysis." En: *Appl. Intell., 41, 473-486.* (Vid. págs. 8, 18).

Doucet, A., N. De Freitas y N. Gordon (2001). "Sequential Monte Carlo Methods in Practice". En: *Springer Verlag* (vid. pág. 76).

Duan, K., D. Batra y D.J. Crandall (2012). "A Multi-Layer Composite Model for Human Pose Estimation". En: *Proceedings of the British Machine Vision Conference, Surrey, UK; pp. 116.1-116.11* (vid. pág. 15).

Efros, A.A. y col. (2003). "Recognizing action at a distance". En: *Proc. Int. Conf. on Computer Vision, (ICCV03). Vol 2. pp 726-734* (vid. pág. 148).

Eichner, M. y V. Ferrari (2010). "We Are Family: Joint Pose Estimation of Multiple Persons". En: *Proceedings of the European Conference on Computer Vision, Heraklion, Crete, Greece; pp. 228-242* (vid. pág. 28).

— (2012). "Human Pose Co-Estimation and Applications". En: *IEEE Trans. Pattern Anal. Mach. Intell., 34, 2282-2288* (vid. pág. 28).

Eichner, M., V. Ferrari y S. Zurich (2009). "Better Appearance Models for Pictorial Structures". En: *Proceedings of the British Machine Vision Conference, London, UK; pp. 1-11* (vid. págs. 26, 29).

Ek, C.H., P.H. Torr y N.D. Lawrence (2007). "Gaussian Process Latent Variable Models for Human Pose Estimation". En: *Machine Learning for Multimodal Interaction; Springer: Heidelberg, Germany; pp. 132-143* (vid. pág. 32).

Elgammal, A., D. Harwood y L. Davis (2000). "Non-Parametric Model for Background Subtraction". En: *European Conference on Computer Vision, Dublin, Ireland* (vid. pág. 132).

Elgammal, A. y C.S. Lee (2014). "Inferring 3D Body Pose from Silhouettes Using Activity Manifold Learning". En: *Proceedings of the IEEE Conference on*

Computer Vision and Pattern Recognition, Washington, DC, USA; pp. 681-688. (Vid. págs. 10, 20, 21).

Elgammal, A. y col. (2003). "Learning dynamics for exemplar based gesture recognition". En: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition, Madinson, WI, USA. Vol 1. pp 571-578* (vid. pág. 149).

Eng, H.L. y col. (2003). "An automatic drowning detection surveillance system for challenging outdoor pool environments". En: *Proc. Int. Conf. on Computer Vision, (ICCV03). Vol 1, pp 532-539.* (Vid. pág. 133).

Epshtein, B. y S. Ullman (2007a). "Semantic hierarchies for recognizing objects and parts". En: *IEEE Conference on Computer Vision and Pattern Recognition* (vid. pág. 43).

— (2007b). "Semantic hierarchies for recognizing objects and parts". En: *CVPR* (vid. pág. 155).

Everingham, M. y col. (2010). "The pascal visual object classes (voc) challenge". En: *International Journal of Computer Vision, vol. 88, no. 2, pp. 303-338* (vid. págs. 99, 100).

Fan, R. y col. (2008). "Liblinear: A library for large linear classification". En: *Journal of Machine Learning Research, vol. 9, pp. 1871-1874* (vid. pág. 47).

Fan, X. y col. (2015). "Combining Local Appearance and Holistic View: Dual-Source Deep Neural Networks for Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA; pp. 1347-1355* (vid. pág. 23).

Fang, W. y L. Yi (2013). "Beyond physical connections: Tree models in human pose estimation". En: *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on. IEEE, pp. 596-603* (vid. pág. 111).

Fanti, C., L. Zelnik-Manor y P. Perona (2005). "Hybrid models for human motion recognition". En: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition, CVPR05. Vol 1. Pp 1166-1173* (vid. pág. 149).

Felzenszwalb, P. y D. Huttenlocher (2004). "Distance transforms of sampled functions". En: *Cornell Computing and Information Science Technical Report TR2004-1963* (vid. pág. 161).

— (2005a). "Pictorial structures for object recognition". En: *International Journal of Computer Vision, vol. 61, no. 1, pp. 55-79* (vid. págs. 14, 45).

— (2005b). "Pictorial structures for object recognition". En: *IJCV, 61(1)* (vid. págs. 155, 160, 161).

Felzenszwalb, P., D. McAllester y D. Ramanan (2008). "A Discriminatively Trained, Multiscale, Deformable Part Model". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA; pp. 1-8* (vid. págs. 14, 35, 107, 153).

Felzenszwalb, P. y col. (2010). "Object detection with discriminatively trained part-based models". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 32, no. 9, pp. 1627-1645* (vid. págs. 10, 14, 27, 31, 43, 45-48, 107, 108, 115, 116, 153, 160).

Fergus, R., P. Perona y A. Zisserman (2003). "Object class recognition by unsupervised scale-invariant learning". En: *CVPR* (vid. pág. 155).

Ferrari, V., M. Marin-Jimenez y A. Zisserman (2008). "Progressive search space reduction for human pose estimation". En: *IEEE Conference on Computer Vision and Pattern Recognition* (vid. págs. 27, 31, 48, 99).

— (2009). "Pose Search: Retrieving People Using Their Pose". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA; pp. 1-8* (vid. págs. 27, 28).

Ferrari, V., M. Marín-Jiménez y A. Zisserman (2009). "2D Human Pose Estimation in TV Shows". En: *Statistical and Geometrical Approaches to Visual Motion Analysis; Springer: Heidelberg, Germany; pp. 128-147* (vid. pág. 27).

Fischler, M. y R. Elschlager (1973). "The representation and matching of pictorial structures". En: *IEEE Transactions on Computer, 22(1):67-92* (vid. págs. 155, 160).

Flitti, F. y col. (2010). "Probabilistic Human Pose Recovery from 2D Images". En: *Proceedings of the IEEE International Conference on Image Processing, Hong Kong, China; pp. 1517-1520* (vid. págs. 20, 23, 25).

Forsythe, D.A. y M.M. Fleck (1997). "Body Plans". En: *Proc. IEEE Computer Vision and Pattern Recognition. pp 678-683* (vid. pág. 140).

Fragkiadaki, K., H. Hu y J. Shi (2013). "Pose from Flow and Flow from Pose". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 2059-2066* (vid. pág. 32).

Freifeld, O. y M.J. Black (2012). "Lie Bodies: A Manifold Representation of 3D Human Shape". En: *Proceedings of the European Conference on Computer Vision, Firenze, Italy; pp. 1-14* (vid. pág. 21).

Freifeld, O. y col. (2010). "Contour People: A Parameterized Model of 2D Articulated Human Shape". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA; pp. 639-646* (vid. págs. 16, 17).

Funda, J. y R. P. Paul (1990). "A computational analysis of screw transformations in robotics". En: *IEEE Trans. Robot. Automat., Vol. 6, pp. 348-356, June* (vid. pág. 206).

Funda, J., R. H. Taylor y R. P. Paul (1990). "On homogeneous transforms,quaternions, and computational efficiency". En: *IEEE Trans. Robot. Automat., Vol. 6, pp. 382-388, June* (vid. pág. 206).

Gall, J., A. Yao y L. Van Gool (2010). "2D Action Recognition Serves 3D Human Pose Estimation". En: *Proceedings of the European Conference on Computer Vision, Heraklion, Crete, Greece; pp. 425-438* (vid. págs. 21, 25).

Gall, J. y col. (2009). "Motion Capture Using Joint Skeleton Tracking and Surface Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA; pp. 1746-1753* (vid. pág. 16).

Gall, J. y col. (2010). "Optimization and filtering for human motion capture: A multi-layer framework". En: *Int. J. Comput. Vis., vol. 87, no. 1-2, pp. 75-92* (vid. págs. 24, 145).

Ganapathi, V., C. Plagemann y S. Koller D.and Thrun (2012). "Real-Time Human Pose Tracking from Range Data". En: *Proceedings of the European Conference on Computer Vision, Firenze, Italy; pp. 738-751* (vid. pág. 24).

Ganapathi, V. y col. (2010). "Real Time Motion Capture Using a Single Time-of-Flight Camera". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA; pp. 755-762* (vid. pág. 24).

Gao, J., A.G. Hauptmann y H.D Wactlar (2004). "Combining Motion Segmentation with Tracking for Activity Analysis". En: *International Conference on Automatic Face and Gesture Recognition, Seoul, Korea. pp 699-704* (vid. pág. 149).

Gavrila, D.M. (1999). "The Visual Analysis of Human Movement: A Survey". En: *Computer Vision and Image Understanding. Vol 73. No 1. pp 83-98* (vid. pág. 128).

— (2007). "A Bayesian, Exemplar-Based Approach to Hierarchical Shape Matching". En: *IEEE Trans. Pattern Anal. Mach. Intell., 29, 1408-1421* (vid. págs. 10, 30).

Güdükbay, U., I. Demir e Y. Dedeoglu (2013). "Motion Capture and Human Pose Reconstruction from a Single-View Video Sequence". En: *Digit. Signal Process., 23, 1441-1450* (vid. pág. 24).

Ge, S. y G. Fan (2015a). "Articulated Non-Rigid Point Set Registration for Human Pose Estimation from 3D Sensors." En: *Sensors, 15, 15218-15245* (vid. pág. 17).

— (2015b). "Non-rigid Articulated Point Set Registration for Human Pose Estimation". En: *Proceedings of the IEEEWinter Conference on Applications of Computer Vision,Waikoloa Beach, HI, USA; pp. 94-101* (vid. pág. 17).

"GestureTek" (2016). En: *Available online: http://www.gesturetek.com* (vid. pág. 8).

Ghosh, S. y col. (2015). "From Deformations to Parts: Motion-Based Segmentation of 3D Objects". En: *Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada; pp. 1997-2005* (vid. pág. 32).

Girshick, R. y col. (2011). "Efficient Regression of General-Activity Human Poses from Depth Images". En: *Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain; pp. 415-422* (vid. pág. 23).

Gkioxari, G., R. Girshick y J. Malik (2015). "Contextual Action Recognition with R-Cnn". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA; pp. 1080-1088* (vid. pág. 22).

Gkioxari, G. y col. (2014a). "R-CNNs for Pose Estimation and Action Detection". En: *arXiv 2014, arXiv:1406.5212* (vid. pág. 23).

— (2014b). "Using k-Poselets for Detecting People and Localizing Their Keypoints". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA; pp. 3582-3589* (vid. pág. 12).

Gloyer, B. (1995). "Video-Based Freeway Monitoring System Using Recursive Vehicle Tracking". En: *IS&T-SPIE Symposium on Electronic Imaging: Image and Video Processing, vol. 2421, pp. 173-180* (vid. pág. 133).

Gong, S., T. Xiang y S. Hongeng (2010). "Learning Human Pose in Crowd". En: *Proceedings of the ACM International Workshop on Multimodal Pervasive Video Analysis, Firenze, Italy; pp. 47-52* (vid. pág. 21).

Gong, W., J. Gonzalez y F.X. Roca (2012). "Human action recognition based on estimated weak poses". En: *EURASIP J. Adv. Signal Process. 2012, 1-14.* (Vid. págs. 9, 20).

Gong, W. y col. (2016). "Human Pose Estimation from Monocular Images: A Comprehensive Survey". En: *Sensors* (vid. pág. 139).

Gonzalez, J. y col. (2002). "aSpaces: Action spaces for recognition and synthesis of human actions". En: *AMDO, pp 189-200* (vid. pág. 150).

Gonzalez, J. y col. (2008). "La silla robotica SENA. Un enfoque basado en la interacción hombre-máquina". En: *Revista Iberoamericana de Automatica e Informatica Industrial pp. 38-47* (vid. pág. 76).

Gonzalez, J.J. y col. (2003). "Robust tracking and Segmentation of Human Motion in an Image Sequence". En: *IEEE Int. Conference on Acoustics, Speech, and Signal Processing, Hong Kong. Vol 3. pp 29-32* (vid. pág. 133).

Gordon, N.J., D.J. Salmond y A.F.M. Smith (1993). "Novel approach to nonlinear/non-Gaussian Baysesian state estimation". En: *IEE Proceedings-F, Vol. 140, No 2, April.* (Vid. pág. 170).

Gorelick, L. y col. (2003). "Shape representation and recognition using the poisson equation". En: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition, vol 2, pp 61-67, Washington DC* (vid. pág. 149).

Gorelick, L. y col. (2006). "Shape Representation and Classification Using the Poisson Equation". En: *IEEE Trans. Pattern Anal. Mach. Intell., 28, 1991-2005* (vid. pág. 11).

Grauman, K., G. Shakhnarovich y T. Darrell (2003). "Inferring 3D Structure with a Statistical Image-based Shape Model". En: *Proc. IEEE Int. Conf. of Computer Vision. Vol 1. Pp 641-647* (vid. págs. 10, 141).

Gu, Y.L. y J.Y.S. Luh (1987). "Dual Number Transformation and Its Application to Robotics". En: *IEEE Journal of Robotics and Automation, Vol.RA-3,no.6,December* (vid. pág. 209).

Guha, P., A. Mukerjee y K.S. Venkatesh (2005). "Efficient Occlusion Handling forMultiple Agent Tracking by Reasoning with Surveillance Event Primitives". En: *Int. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, Beijing, China. pp 49-56* (vid. págs. 132, 137).

Guo, F. y G. Qian (2008). "Monocular 3D Tracking of Articulated Human Motion in Silhouette and Pose Manifolds". En: *J. Image Video Process., 2008, 4* (vid. pág. 32).

Gupta, A. y col. (2008). "Context and Observation Driven Latent Variable Model for Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA; pp. 1-8* (vid. pág. 21).

Hahn, M. y col. (2007). "Tracking of Human Body Parts Using the Multiocular Contracting Curve Density Algorithm". En: *Proceedings of the International Conference on 3-D Digital Imaging and Modeling, Montreal, QC, Canada; pp. 257-264* (vid. pág. 31).

Han, D., Q. Wei y Z. Li (2008). "Kinematic Control of Free Rigid Bodies Using Dual Quaternions". En: *International Journal of Automation and Computing, July, pp. 319-324* (vid. págs. 204, 206).

Hao, W., M. Fanhui y F. Baofu (2014). "Iterative Human Pose Estimation Based on a New Part Appearance Model". En: *Appl. Math., 8, 311-317* (vid. pág. 27).

Hara, K. y R. Chellappa (2013). "Computationally Efficient Regression on a Dependency Graph for Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 3390-3397* (vid. págs. 21, 23).

Hara, K. y T. Kurokawa (2011). "Human Pose Estimation Using Patch-Based Candidate Generation and Model-Based Verification". En: *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition, Santa Barbara, CA, USA; pp. 687-693* (vid. pág. 11).

Haritaoglu, I., M. Flickner y D. Beymer (2002). "Ghost3D: Detecting Body Posture and Parts Using Stereo". En: *Workshop on Motion and Video Computing, Orlando, Florida. Vol 46(3). Pp 34-39.* (Vid. págs. 135, 136).

Haritaoglu, I., D. Harwood y L.S. Davis (1998a). "Ghost: A Human Body Part Labeling System Using Silhouettes". En: *International Conference on Pattern Recognition. Vol 1. pp 77-82* (vid. pág. 139).

— (1998b). "W4: Who? When? Where? What? - A Real Time System for Detecting and Tracking People". En: *International Conference on Automatic Face and Gesture Recognition, Nara, Japan. pp 222-227* (vid. pág. 138).

Haritaoglu, Isamail, David Harwood y Larry S. Davis (2000). "W4: Real-Time Surveillance of People and Their Activities". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol 22, No 8. pp 809-830* (vid. págs. 132, 133, 135, 137, 140, 149).

Hart, J. C., G. K. Fracis y L.H. Kaauffman (1994). "Visualizing Quaternion Rotation". En: *ACM Transactions on Graphics, Vol. 13, No. 3, July, p. 256-276* (vid. pág. 204).

Hayashi, K. y col. (2004). "Multiple-Person Tracker with a Fixed Slanting Stereo Camera". En: *International Conference on Automatic Face and Gesture Recognition, Seoul, Korea. pp 681-686* (vid. pág. 135).

Heikkila, M. y M. Pietikainen (2006). "A Texture-Based Method for Modeling the Background and Detecting Moving Objects". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, 28(4). pp 657-662* (vid. pág. 132).

Herda, L., R. Urtasun y P. Fua (2005). "Hierarchical implicit surface joint limits for human body tracking". En: *Computer Vision and Image Understanding, 99(2). pp 189-209* (vid. pág. 130).

Hernández, N. y col. (2008). "Relevance Vector Machines for Multivariate Calibration Purposes". En: *J. Chemom., 22, 686-694* (vid. pág. 30).

Hernández-Vela, A. y col. (2012). "Graph Cuts Optimization for Multi-Limb Human Segmentation in Depth Maps". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA; pp. 726-732* (vid. pág. 26).

Hierarchical Models for Visual Recognition and Learning of Objects, Scenes, and Activities, Studies in Systems, Decision and Control, Cap. 6 (2015). Springer International Publishing Switzerland (vid. pág. 139).

Hilton, A. y col. (1999). "Virtual People: Capturing human models to populate virtual worlds". En: *International Conference on Computer Animation, pp 174-185* (vid. pág. 130).

Hirota, M., Y. Nakajima y M. Saito (2003). "Uchiyama, M. Human Body Detection Technology by Thermoelectric Infrared Imaging Sensor". En: *Proceedings of the International Technical Conference on the Enhanced Safety of Vehicles, Nagoya, Japan, pp. 1-10.* (Vid. pág. 8).

Hogg, David (1983). "Model-based vision: a program to see a walking person". En: *Image and Vision Computing. Vol 1 no 1. Pp 5-20* (vid. pág. 139).

Holub, A. y P. Perona (2005). "A discriminative framework for modelling object classes". En: *CVPR, pages I: 664-671* (vid. pág. 155).

Horprasert, T., D. Harwood y L.S. Davis (1999). "A Statistical Approach for Real-time Robust Background Subtraction and Shadow Detection". En: *IEEE ICCV-99 FRAMERATE WORKSHOP, Corfu, Greece. pp 1-19* (vid. pág. 132).

Hou, S. y col. (2007). "Real-time Body Tracking Using a Gaussian Process Latent Variable Model". En: *Proceedings of the IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil; pp. 1-8* (vid. pág. 32).

Howe, N.R. (2004). "Silhouette Lookup for Automatic Pose Tracking". En: *IEEE Workshop on Articulated and Non-Rigid Motion. pp 15-22* (vid. pág. 141).

Howe, N.R., M.E. Leventon y W.T. Freeman (2000). "Bayesian Reconstruction of 3D Human Motion from Single- Camera Video". En: *Advances in Neural Information Processing Systems 12. MIT Press. pp 820-826* (vid. pág. 146).

Hu, M., W. Hu y T. Tan (2004). "Tracking People through Occlusion". En: *International Conference on Pattern Recognition, Cambridge, UK. Vol 2. pp 724-727* (vid. pág. 134).

Huang, J.B. y M.H. Yang (2009). "Estimating Human Pose from Occluded Images". En: *Proceedings of the Asian Conference on Computer Vision, Xi'an, China; pp. 48-60* (vid. pág. 24).

— (2010). "Fast Sparse Representation with Prototypes". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA; pp. 3618-3625* (vid. pág. 24).

Huang, Z. e Y. L. Yao (1999). "Extension of Usable Workspace of Rotational Axes in Robot Planing". En: *Robotica Vol. 17, pp. 293-301. Printed in the United Kingdom - Cambridge University Press* (vid. pág. 202).

"Inferring Body Pose without Tracking Body Parts." (2016). En: *Available online: http://ieeexplore.ieee.org/ document/854946/* (vid. pág. 10).

"Integrating Bottom-up / Top-down for Object Recognition by Data Driven Markov Chain Monte Carlo" (2016). En: *Available online: http://ieeexplore.ieee.org/ document/855894/* (vid. págs. 25, 32).

Iofee, Sergey y David Forsyth (1999). "Finding People by sampling". En: *ICCV 1999 pp 1092-1097* (vid. pág. 140).

Ioffe, S. y D. Forsyth (2001). "Probabilistic methods for finding people". En: *IJCV, 43(1):45-68* (vid. pág. 155).

Ionescu, C., L. Bo y C. Sminchisescu (2009). "Structural SVM for Visual Localization and Continuous State Estimation". En: *Proceedings of the IEEE International Conference on Computer Vision, Kyoto, Japan; pp. 1157-1164* (vid. pág. 21).

Ionescu, C., F. Li y C. Sminchisescu (2011). "Latent Structured Models for Human Pose Estimation". En: *Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain; pp. 2220-2227* (vid. págs. 11, 21, 30).

Ivanov, Y.A., A.F. Bobick y J. Liu (2000). "Fast Lighting Independent Background Subtraction". En: *International Journal of Computer Vision, 37(2). pp 49-55* (vid. págs. 135, 136).

Iwase, S. y H. Saito (2004). "Parallel Tracking of All Soccer Players by Integrating Detected Positions in Multiple View Images". En: *International Conference on Pattern Recognition, Cambridge, UK. Vol 4. pp 751-754* (vid. pág. 135).

Jaeggli, T., E. Koller-Meier y L. Van Gool (2007). "Learning Generative Models for Monocular Body Pose Estimation". En: *Proceedings of the Asian Conference on Computer Vision, Tokyo, Japan; pp. 608-617* (vid. pág. 18).

Jain, A. y col. (2014a). "Learning Human Pose Estimation Features with Convolutional Networks". En: *arXiv 2014, arXiv:1312.7302* (vid. pág. 22).

Jain, A. y col. (2014b). "Modeep: A Deep Learning Framework Using Motion Features for Human Pose Estimation". En: *Proceedings of the Asian Conference on Computer Vision, Singapore; pp. 302-315* (vid. pág. 22).

Jalala, A. y col. (2017). "Robust human activity recognition from depth video using spatiotemporal multi-fused features". En: *Pattern Recognition Volume 61, Pages 295-308* (vid. pág. 13).

Janabi-Sharifi, F. y M. Marey (2010). "A kalman-filter-based method for pose estimation in visual servoing". En: *Robotics IEEE Transactions, vol. 26, no. 5, pp. 939-947* (vid. pág. 144).

Jenkins, O.C. y M.J. Mataric (2002). "Deriving action and behavior primitives from human motion data". En: *Proc. IEEE Int. Conf. on Intelligent Robots and Systems, pp 2551-2556, Lausanne, Switzerland* (vid. pág. 150).

Jiang, H. (2010). "Finding Human Poses in Videos Using Concurrent Matching and Segmentation". En: *Proceedings of the Asian Conference on Computer Vision, Queenstown, New Zealand; pp. 228-243* (vid. págs. 16, 27).

— (2011). "Human Pose Estimation Using Consistent Max Covering". En: *IEEE Trans. Pattern Anal. Mach. Intell., 33, 1911-1918* (vid. págs. 11, 30).

Jin, Y. y S. Geman (2006). "Context and hierarchy in a probabilistic image model". En: *CVPR, pages II: 2145-2152* (vid. pág. 155).

Johnson, S. y M. Everingham (2010). "Clustered Pose and Nonlinear Appearance Models for Human Pose Estimation". En: *Proceedings of the British Machine Vision Conference, Aberystwyth, Wales, UK; pp. 12.1-12.11* (vid. pág. 14).

— (2011). "Learning Effective Human Pose Estimation from Inaccurate Annotation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA; pp. 1465-1472* (vid. pág. 15).

Jordan, M.I. y R.A. Jacobs (1994). "Hierarchical Mixtures of Experts and the EM Algorithm". En: *Neural Comput., 6,181-214* (vid. pág. 20).

Ju, S.X., M.J. Black e Y. Yacoob (1996). "Cardboard People: A Parameterized Model of Articulated Image Motion". En: *Proceedings of the International Conference on Automatic Face and Gesture Recognition, Killington, VT, USA; pp. 38-44* (vid. pág. 31).

Kakadiaris, I.A. y D. Metaxas (1995). "3D Human Body Model Acquisition from Multiple Views". En: *Proceedings of the IEEE International Conference on Computer Vision, Cambridge, MA, USA; pp. 618-623* (vid. pág. 10).

Kale, A. y col. (2002). "Human identification using gait". En: *Proc. Int. Conf. on Automatic Face and Gesture Recognition, Washington, DC, USA. pp 137-142* (vid. págs. 133, 148).

Kanaujia, A. (2014). "Coupling Top-down and Bottom-up Methods for 3D Human Pose and Shape Estimation from Monocular Image Sequences". En: *arXiv 2014, arXiv:1410.0117* (vid. pág. 25).

Kanaujia, A., C. Sminchisescu y D. Metaxas (2007). "Semi-Supervised Hierarchical Models for 3D Human Pose Reconstruction". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA; pp. 1-8* (vid. págs. 11, 21).

Kehl, R., M. Bray y L. VanGool (2005). "Full Body Tracking from Multiple Views Using Stochastic Sampling". En: *Proc. IEEE Computer Vision and Pattern Recognition. Vol 2. pp 129-136* (vid. págs. 10, 144).

Khan, S. y M. Shah (2000). "Tracking People in Presence of Occlusion". En: *Asian Conference on Computer Vision, Taipei, Taiwan. pp 1132-1137* (vid. págs. 134, 137, 138).

Kiefel, M. y P.V. Gehler (2014). "Human Pose Estimation with Fields of Parts". En: *Proceedings of the European Conference on Computer Vision, Zurich, Switzerland; pp. 331-346* (vid. pág. 28).

Kim, H. y col. (2015). "Real-Time Human Pose Estimation and Gesture Recognition from Depth Images Using Superpixels and SVM Classifier". En: *Sensors, Volume 15, Issue 6* (vid. pág. 20).

Kim, K. y col. (2005). "Real-Time Foreground-Background Segmentation using Codebook Model". En: *Elsevuer Real-Time Imaging, 11(3). pp 172-185.* (Vid. págs. 132, 133).

Kohli, P., J. Rihan y P.H. Bray M.and Torr (2008). "Simultaneous Segmentation and Pose Estimation of Humans Using Dynamic Graph Cuts". En: *Int. J. Comput. Vis., 79, 285-298* (vid. pág. 30).

Kohli, P. y P.H. Torr (2005). "Efficiently Solving Dynamic Markov Random Fields Using Graph Cuts". En: *Proceedings of the IEEE International Conference on Computer Vision, Beijing, China; pp. 922-929* (vid. pág. 31).

Kolmogorov, V. y R. Zabin (2004). "What Energy Functions Can be Minimized via Graph Cuts?" En: *IEEE Trans. Pattern Anal. Mach. Intell., 26, 147-159* (vid. pág. 31).

Komodakis, N., N. Paragios y G. Tziritas (2011). "MRF Energy Minimization and Beyond via Dual Decomposition". En: *IEEE Trans. Pattern Anal. Mach. Intell., 33, 531-552* (vid. pág. 15).

Kong, A., J. S. Liu y W. H. Wong (1994). "Sequential imputations and Bayesian missing data problems". En: *J. Amer. Stat. Assoc., 89(425) pp. 278-288* (vid. pág. 76).

Konietschke, R. y col. (2006). "Kinematic Design Optimization of an Actuated Carrier for the DLR Multi-Arm Surgical System". En: *Intelligent Robots and Systems, 2006 IEEE RSJ International Conference on Volume , Issue , 9-15 Oct. pp. 4381- 4387* (vid. pág. 75).

Kostrikov, I. y J. Gall (2014). "Depth Sweep Regression Forests for Estimating 3D Human Pose from Images". En: *BMVC, bmva.org* (vid. pág. 18).

Krahnstoever, N. y R. Sharma (2004). "Articulated Models from Video". En: *Proc.IEEE Computer Vision and Pattern Recognition. Vol 1. pp 894-901.* (Vid. pág. 130).

Krotosky, S.J. y M.M. Trivedi (2004). "Occupant Posture Analysis Using Reflectance and Stereo Image for Smart Airbag Deployment". En: *Proceedings of the IEEE Intelligent Vehicles Symposium, Parma, Italy; pp. 698-703.* (Vid. pág. 9).

Kruger, J., T.K. Lien y A. Verl (2009). "Cooperation of human and machines in assembly lines". En: *CIRP Annals-Manufacturing* (vid. pág. 76).

Kumar, M., A. Zisserman y P. Torr (2009). "Efficient discriminative learning of parts-based models". En: *IEEE International Conference on Computer Vision* (vid. pág. 46).

Kuo, P., D. Makris y J.C. Nebel (2011). "Integration of Bottom-up / Top-down Approaches for 2D Pose Estimation Using Probabilistic Gaussian Modelling". En: *Comput. Vis. Image Underst., 115, 242-255* (vid. pág. 25).

Ladicky, L., P. Torr y A. Zisserman (2013). "Pose Estimation Using a Joint Pixel-Wise and Part-Wise Formulation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 3578-3585* (vid. págs. 27, 31).

Lallemand, J., M. Szczot y S. Ilic (2014). "Human Pose Estimation in Stereo Images". En: *Proceedings of the International Conference on Articulated Motion and Deformable Objects, Palma de Mallorca, Spain; pp. 10-19* (vid. pág. 11).

Lazebnik, S., C. Schmid y J. Ponce (2006). "Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, USA; pp. 2169-2178* (vid. pág. 11).

"Leap Motion" (2016). En: *Available online: http://www.leapmotion.com* (vid. pág. 8).

LeCun, Y. y col. (2006). "A tutorial on energy-based learning". En: *G. Bakir, T. Hofman, B. Scholkopf, A. Smola, and B. Taskar, editors, Predicting Structured Data. MIT Press* (vid. pág. 163).

Lee, C.S. y A. Elgammal (2007). "Modeling View and Posture Manifolds for Tracking". En: *Proceedings of the IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil; pp. 1-8* (vid. pág. 25).

— (2010). "Coupled Visual and Kinematic Manifold Models for Tracking". En: *Int. J. Comput. Vis., 87, 118-139* (vid. pág. 20).

Lee, M.W. e I. Cohen (2004a). "Human Upper Body Pose Estimation in Static Images". En: *Proceedings of the European Conference on Computer Vision, Prague, Czech Republic; pp. 126-138* (vid. pág. 32).

— (2004b). "Proposal Maps driven MCMC for Estimating Human Body Pose in Static Images". En: *Proc. IEEE Computer Vision and Pattern Recognition. Vol 2. pp 334-341* (vid. pág. 145).

Lee, M.W. y R. Nevatia (2006). "Human Pose Tracking Using Multi-Level Structured Models". En: *Proceedings of the European Conference on Computer Vision, Graz, Austria; pp. 368-381* (vid. pág. 29).

Lehrmann, A., P. Gehler y S. Nowozin (2013). "A Non-parametric Bayesian Network Prior of Human Pose". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 1281-1288* (vid. págs. 14, 15).

Leibe, B., E. Seemann y B. Schiele (2005). "Pedestrian Detection in Crowded Scenes". En: *Computer Vision and Pattern Recognition, San Diego, CA, USA. Vol 1. pp 878-885* (vid. pág. 135).

Leo, M. y col. (2004). "Complex human activity recognition for monitoring wide outdoor environments". En: *Proc. Int. Conf. on Pattern Recognition, Cambridge, UK. Vol 4. pp 913-916* (vid. pág. 149).

Lepetit, V. y P. Fua (2005). "Monocular Model-Based 3D Tracking of Rigid Objects: A Survey". En: *Found. Trends Comput. Graph. Vis, 1, 1-89.* (Vid. págs. 9, 24).

— (2006). "Keypoint Recognition Using Randomized Trees". En: *IEEE Trans. Pattern Anal. Mach. Intell., 28, 1465-1479* (vid. pág. 23).

Li, S. y A.B. Chan (2014). "3D Human Pose Estimation from Monocular Images with Deep Cnvolutional Neural Network". En: *Proceedings of the Asian Conference on Computer Vision, Singapore; pp. 332-347* (vid. pág. 12).

Li, S., Z.Q. Liu y A. Chan (2014). "Heterogeneous Multi-Task Learning for Human Pose Estimation with Deep Convolutional Neural Network". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, Columbus, OH, USA; pp. 482-489* (vid. págs. 12, 22).

Li, Y., A. Hilton y J. Ilingworth (2002). "A Relaxation Algorithm for Real-Time Multiple View 3D-Tracking". En: *Image and Vision Computing, 20(12). pp 841-859* (vid. pág. 137).

Li, Y. y Z. Sun (2009). "Vision-Based Human Pose Estimation for Pervasive Computing". En: *Proceedings of the ACM Workshop on Ambient Media Computing, Beijing, China; pp. 49-56.* (Vid. pág. 9).

Lim, S.N. y col. (2005). "Fast Illumination-Invariant Background Subtraction Using Two Views: Error Analysis, Sensor Placement and Applications". En: *Computer Vision and Pattern Recognition, San Diego, CA, USA. Vol 1. pp 1071-1078* (vid. pág. 136).

Liu, W. y col. (2016). "Real time pose estimation based on extended Kalman filter for binocular camera". En: *Intelligent Robot Systems (ACIRS), Asia-Pacific Conference* (vid. pág. 144).

Liu, Y. y col. (2011). "Markerless Motion Capture of Interacting Characters Using Multi-view Image Segmentation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA; pp. 1249-1256* (vid. pág. 18).

Liu, Z. y col. (2015). "A Survey of Human Pose Estimation: The Body Parts Parsing based Methods". En: *J. Vis. Commun. Image Represent, 32, 10-19.* (Vid. pág. 9).

Lowe, D.G. (1999). "Object Recognition from Local Scale-Invariant Features". En: *Proceedings of the IEEE International Conference on Computer Vision, Corfu, Greece; pp. 1150-1157* (vid. pág. 11).

— (2004). "Distinctive Image Features from Scale-Invariant Keypoints". En: *Int. J. Comput. Vis., 60, 91-110* (vid. pág. 11).

Loy, G., M. Eriksson y J. Sullivan (2004). "Monocular 3D Reconstruction of Human Motion in Long Action Sequences". En: *Proc. European Conf. of Computer Vision, LNCS, Springer-Verlag. Vol 3024. pp 442-455* (vid. pág. 144).

Lu, C. y N. Ferrier (2004). "Repetitive motion analysis: Segmentation and event classification". En: *IEEE Trans. Pattern Analysis and Machine Intelligence, 26(2). pp 258-263* (vid. pág. 150).

Lu, H., X. Shao e Y. Xiao (2013). "Pose Estimation with Segmentation Consistency". En: *IEEE Trans. Image Process., 22, 4040-4048* (vid. págs. 26, 30).

Lu, Y. y H. Jiang (2013). "Human Movement Summarization and Depiction from Videos". En: *Proceedings of the IEEE International Conference on Multimedia and Expo, San Jose, CA, USA; pp. 1-6* (vid. pág. 12).

Luo, Y., T.W. Wu y J.N. Hwang (2003). "Object-based analysis and interpretation of human motion in sports video sequences by dynamic bayesian networks". En: *Computer Vision and Image Understanding, 92. pp 196-216* (vid. pág. 149).

MacCormick, J. y M. Isard (2000). "Partitioned sampling, articulated objects, and interface quality hand tracking". En: *Proc. European Conference on Computer Vision. Vol 2. pp 3-19* (vid. pág. 143).

Marin, J. y col. (2013). "Random Forests of Local Experts for Pedestrian Detection". En: *Computer Vision (ICCV) 2013 IEEE International Conference on, pp. 2592-2599, ISSN 1550-5499.* (Vid. pág. 133).

Martinez, E., F. Benimeli y A. Sanchez (2012). "Kalman Filter for Tracking Robotic Arms Using low cost 3D Vision Systems". En: *5th International Conference on Advances in Computer-Human Interactions (ACHI 2012)* (vid. pág. 120).

Martinez, E., D.A. Hernandez y A. Sanchez (2011). "Visión 3D De Bajo Coste Para La Interacción Humano-Robot Utilizando Filtro De Partículas". En: *XXXII Jornadas de Automática* (vid. pág. 120).

Martinez, E. y A. Sanchez (2011). "Cinemática Directa e Inversa Robot Paralelo". En: *XXXII Jornadas de Automática* (vid. pág. 120).

Martinez, E. y col. (2016). "Human Pose Estimation for RGBD Imagery with Multi-Channel Mixture of Parts and Kinematic Constraints". En: *WSEAS TRANSACTIONS on COMPUTERS, Volume 15 - 2016, E-ISSN: 2224-2872.* (Vid. pág. 120).

Martinez, E. y col. (2017). "Optimized 4D-DPM For Pose Estimation On RGBD Channels using polisphere models". En: *12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - (Volume 5) - P. 281 - 288* (vid. pág. 120).

Matas, J. y col. (2004). "Robust wide baseline stereo from maximally stable extremal regions". En: *Image and vision computing 22(10): 761-767* (vid. pág. 39).

McKenna, S.J. y col. (2000). "Tracking Interacting People". En: *The fourth International Conference on Automatic Face and Gesture Recognition, Grenoble, France. pp 348-353* (vid. págs. 132, 134, 137, 138).

"Meet Kinect forWindows." (2016). En: *Available online: http://www.microsoft.com/en-us/kinectforwindows* (vid. pág. 8).

Memisevic, R., L. Sigal y D.J. Fleet (2012). "Shared Kernel Information Embedding for Discriminative Inference". En: *IEEE Trans. Pattern Anal. Mach. Intell., 34, 778-790* (vid. pág. 20).

Menache, A. (1999). "Understanding Motion Capture for Computer Animation and Video Games". En: *Morgan Kaufmann* (vid. pág. 124).

Metaxas, D. y D. Terzopoulos (1993). "Shape and Nonrigid Motion Estimation through Physics-Based Synthesis". En: *IEEE Trans. Pattern Anal. Mach. Intell., 15, 580-591* (vid. pág. 18).

"Metric Regression Forests for Human Pose Estimation" (2016). En: *Available online: www.bmva.org/bmvc/2013/ Papers/paper0004/paper0004.pdf* (vid. pág. 20).

Micilotta, A., E. Ong y R. Bowden (2005). "Detection and tracking of humans by probabilistic body part assembly". En: *Proc. British Machine Vision Conf., Vol. 1, pp 429-438* (vid. pág. 131).

Mikic, Ivana y col. (2002). "Human Body Model Acquisition and Motion Capture Using Voxel Data". En: *AMDO 2002. pp 104-118* (vid. pág. 141).

Mitchelson, J. y A. Hilton (2005). "Hierarchical tracking of multiple people". En: *British Machine Vision Conference* (vid. pág. 143).

Mittal, A. y L.S. Davis (2005). "M2Tracker: A Multi-View Approach to Segmenting and Tracking People in a Cluttered Scene". En: *International Journal of Computer Vision, 51(3). Pp 189-203* (vid. págs. 135, 136).

Moeslund, T.B. (2001). "A Survey of Computer Vision-Based Human Motion Capture". En: *Computer Vision and Image Understanding: CVIU. Vol 81. No 3. pp 231-268.* (Vid. págs. 128, 129).

Moeslund, T.B. y E. Granum (2001a). "A Survey of Computer Vision-based Human Motion Capture". En: *Comput. Vis. Image Underst., 81, 231-268* (vid. pág. 18).

— (2001b). "Pose Estimation of a Human Arm using Kinematic Constraints". En: *12th Scandinavian Conference on Image Analysis, Bergen, Norway.* (Vid. pág. 130).

Moeslund, T.B., A. Hilton y V. Kruger (2006). "A Survey of Advances in Vision-Based Human Motion Capture and Analysis". En: *Comput. Vis. Image Underst. 2006, 104, 90-126.* (Vid. pág. 9).

Moeslund, T.B., C.B. Madsen y E. Granum (2005). "Modelling the 3D Pose of a Human Arm and the Shoulder Complex utilising only Two Parameters". En: *International Journal on Integrated Computer-Aided Engineering, 12(2)* (vid. pág. 145).

Moeslund, T.B. y col. (2002). "Estimating the 3D Shoulder Position using Monocular Vision". En: *International Conference on Imaging Science, Systems, and Technology, Las Vegas, Nevada. pp 21-27.* (Vid. pág. 130).

Mohan, A., C. Papageorgiou y T. Poggio (2001). "Example-Based Object Detection in Images by Components". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, 23(4). pp 349-361* (vid. págs. 134, 140).

Moon, S. y col. (2016). "Multiple Kinect Sensor Fusion for Human Skeleton Tracking Using Kalman Filtering". En: *International Journal of Advanced Robotic Systems* (vid. pág. 142).

Morariu, V.I. y O.I. Camps (2006). "Modeling Correspondences for Multi-Camera Tracking Using Nonlinear Manifold Learning and Target Dynamics". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, USA; pp. 545-552* (vid. pág. 21).

Mori, G. (2005). "Guiding Model Search Using Segmentation". En: *Proceedings of the IEEE International Conference on Computer Vision, Beijing, China; pp. 1417-1423* (vid. pág. 27).

Mori, G., S. Belongie y J. Malik (2005). "Efficient Shape Matching Using Shape Contexts". En: *IEEE Trans. Pattern Anal. Mach. Intell., 27, 1832-1837* (vid. pág. 11).

Mori, G. y J. Malik (2006). "Recovering 3D Human Body Configurations Using Shape Contexts". En: *IEEE Trans. Pattern Anal. Mach. Intell., 28, 1052-1062* (vid. págs. 20, 30).

Mori, G. y col. (2015). "Pose Embeddings: A Deep Architecture for Learning to Match Human Poses". En: *arXiv 2015, arXiv:1507.00302* (vid. pág. 21).

Mukundan, R. (2002). "Quaternions: From Classical Mechanics to Computer Graphics and Beyond". En: *Proceedings of the 7th Asian Technology Conference in Mathematics* (vid. pág. 204).

Mulligan, J. (2005). "Upper Body Pose Estimation from Stereo and Hand-Face Tracking". En: *Canadian Conference on Compuer and Robot Vision, Victoria, British Columbia, Canada. pp 413-420* (vid. pág. 130).

Murray, M., Z. Li y Sastry S.S. (1994). "A mathematical introduction to robotic manipulation". En: *Boca Raton FL: CRC Press* (vid. págs. 112, 217).

Muybridge, E. (1955). "The Human Figure in Motion". En: *Dover Publications* (vid. pág. 123).

Nakariyakul, S. (2014). "A Comparative Study of Suboptimal Branch and Bound Algorithms". En: *Inf. Sci., 278, 545-554* (vid. pág. 27).

Navaratnam, R. y col. (2005). "Hierarchical partbased human body pose estimation". En: *Proc. British Machine Vision Conf., Vol. 1, pp 429-438* (vid. pág. 145).

Nayak, S., S. Sarkar y B. Loeding (2009). "Distribution-Based Dimensionality Reduction Applied to Articulated Motion Recognition". En: *IEEE Trans. Pattern Anal. Mach. Intell., 31, 795-810* (vid. pág. 11).

Nie, X., C. Xiong y S.C. Zhu (2015). "Joint Action Recognition and Pose Estimation from Video". En: *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition, Boston, WA, USA; pp. 1293-1301.* (Vid. pág. 9).

Ning, H. y col. (2008). "Discriminative Learning of Visual Words for 3D Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA; pp. 1-8* (vid. págs. 20, 22, 25, 29).

Nowozin, S. (2012). "Improved Information Gain Estimates for Decision Tree Induction". En: *Proceedings of the International Conference on Machine Learning, Edinburgh, UK; pp. 297-304* (vid. pág. 24).

Okada, R. y S. Soatto (2008). "Relevant Feature Selection for Human Pose Estimation and Localization in Cluttered Images". En: *Proceedings of the European Conference on Computer Vision, Marseille, France; pp. 434-445* (vid. pág. 20).

Oleinikov, G. y col. (2014). "Task-based Control of Articulated Human Pose Detection for Openvl". En: *Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Colorado Springs, CO, USA; pp. 682-689.* (Vid. págs. 8, 12).

Oliver, N., B. Rosario y A. Pentland (2000). "A Bayesian Computer Vision System for Modeling Human Interactions". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(8). pp 831-843* (vid. pág. 134).

Olshausen, B.A. y D.J. Field (1997). "Sparse Coding with an Overcomplete Basis Set: A Strategy Employed by V1". En: *Vis. Res., 37, 3311-3325* (vid. pág. 21).

Ong, E.J. y A. Hilton (2005). "Learnt Inverse Kinematics for Animation Synthesis". En: *IMA Conference on Vision, Video and Graphics, pp 11-20* (vid. pág. 146).

Orrite-Urunuela, C., J.E. Herrero-Jaraba y G. Rogez (2004). "2D Silhouette and 3D Skeletal Models for Human Detection and Tracking". En: *Proceedings of the International Conference on Pattern Recognition, Cambridge, UK; pp. 244-247* (vid. pág. 25).

Ouyang, W., X. Chu y X. Wang (2014). "Multi-Source Deep Learning for Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA; pp. 2329-2336* (vid. pág. 22).

Ozer, I.B. y W.H. Wolf (2002). "A Hierarchical Human Detection System in (Un) Compressed Domains". En: *IEEE Transactions on Multimedia, 4(2). pp 283-300* (vid. págs. 128, 134, 149).

Paden, B. (1986). "Kinematics and Control Robot Manipulators". En: *PhD thesis Department of Electrical Engineering and Computer Science, University of California,Berkeley,* (vid. pág. 217).

Parameswaran, V. y R. Chellappa (2003). "View Invariants for Human Action Recognition". En: *Computer Vision and Pattern Recognition, Madison, Wisconsin. Vol 2, pp 613-619* (vid. pág. 149).

Parameswaran, V. y R. Chellappa (2004). "View Independent Human Body Pose Estimation from a Single Perspective Image". En: *Proc. IEEE Computer Vision and Pattern Recognition. Vol 2. pp 16-22.* (Vid. págs. 24, 130).

Park, S. y J.K. Aggarwal (2002). "Segmentation and tracking of interacting human body parts under occlusion and shadowing". En: *Workshop on Motion and Video Computing, Orlando, Florida. pp 105-111* (vid. pág. 138).

Park, S.I. y J.K. Hodgins (2006). "Capturing and Animating Skin Deformation in Human Motion". En: *ACM Trans. Graph., 25, 881-889* (vid. pág. 17).

Peng, G. y col. (2009). "Estimating Human Shape and Pose from a Single Image". En: *Proceedings of the IEEE International Conference on Computer Vision, Kyoto, Japan ; pp. 1381-1388* (vid. pág. 17).

Penmetsa, S. y col. (2014). "Autonomous UAV for Suspicious Action Detection Using Pictorial Human Pose Estimation and Classification". En: *Electron. Lett. Comput. Vis. Image Anal., 13, 18-32* (vid. pág. 28).

Perez-Sala, X. y col. (2014). "A Survey on Model Based Approaches for 2D and 3D Visual Human Pose Recovery". En: *Sensors, 14, 4189-4210.* (Vid. pág. 9).

Pfister, T., J. Charles y A. Zisserman (2015). "Flowing Convnets for Human Pose Estimation in Videos". En: *Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile; pp. 1913-1921* (vid. pág. 12).

Pfister, T. y col. (2014). "Deep Convolutional Neural Networks for Efficient Pose Estimation in Gesture Videos". En: *Proceedings of the Asian Conference on Computer Vision, Singapore; pp. 538-552* (vid. pág. 23).

Pinto, N., D.D. Cox y J.J. DiCarlo (2008). "Why is Real-World Visual Object Recognition Hard?" En: *PLoS Comput. Biol., 4, e27* (vid. pág. 22).

Pishchulin, L., A. Jain y M. Andriluka (2012). "Articulated People Detection and Pose Estimation: Reshaping the Future". En: *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Providence, Rhode Island* (vid. pág. 145).

Pishchulin, L. y col. (2013). "Poselet conditioned pictorial structures". En: *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, pages 588-595* (vid. págs. 28, 36).

Plankers, R. y P. Fua (2003). "Articulated Soft Objects for Multiview Shape and Motion Capture". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(9). pp 1182-1187* (vid. págs. 130, 142).

Pons-Moll, G. y B. Rosenhahn (2011). "Model-Based Pose Estimation". En: *Visual Analysis of Humans; Springer: Heidelberg, Germany; pp. 139-170* (vid. pág. 24).

Pons-Moll, G. y col. (2011a). "Efficient and Robust Shape Matching for Model Based Human Motion Capture". En: *Proceedings of the Joint Pattern Recognition Symposium, Frankfurt/Main, Germany; pp. 416-425* (vid. pág. 24).

Pons-Moll, G. y col. (2011b). "Outdoor Human Motion Capture Using Inverse Kinematics and von Mises-Fisher Sampling". En: *Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain; pp. 1243-1250* (vid. pág. 24).

Poppe, R. (2007). "Vision-Based Human Motion Analysis: An Overview". En: *Comput. Vis. Image Underst., 108, 4-18.* (Vid. pág. 9).

— (2010). "A Survey on Vision-Based Human Action Recognition". En: *Image Vis. Comput., 28, 976-990* (vid. pág. 9).

Probst, T., L. Fossati y L.V. Gool (2016). "Combining Human Body Shape and Pose Estimation for Robust Upper Body Tracking Using a Depth Sensor". En: *European Conference on Computer Vision ECCV: Computer Vision - ECCV Workshops pp 285-301* (vid. pág. 24).

Quattoni, A. y col. (2007). "Hidden conditional random fields". En: *PAMI, 29(10):1848-1852* (vid. pág. 155).

Radwan, I., A. Dhall y R. Goecke (2013). "Monocular Image 3D Human Pose Estimation under Self-Occlusion". En: *Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia; pp. 1888-1895* (vid. pág. 29).

Ramakrishna, V., T. Kanade e Y. Sheikh (2013). "Tracking Human Pose by Tracking Symmetric Parts". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 3728-3735* (vid. pág. 15).

Ramakrishna, V. y col. (2014). "Pose machines: Articulated pose estimation via inference machines". En: *Computer Vision-ECCV 2014, pages 33-47* (vid. pág. 36).

Ramanan, D. (2007). "Learning to parse images of articulated bodies". En: *Advances in Neural Information Processing System* (vid. págs. 26, 48, 99).

— (2012). "Dual coordinate descent solvers for large structured prediction problems". En: *UC Irvine, Tech. Rep* (vid. pág. 47).

Ramanan, D., D.A. Forsyth y A. Zisserman (2005). "Strike a Pose: Tracking People by Finding Stylized Poses". En: *Proc. IEEE Computer Vision and Pattern Recognition. Vol 1. pp 271-278.* (Vid. págs. 131, 140).

Ramanan, D. y C. Sminchisescu (2006). "Training deformable models for localization". En: *CVPR, pages I: 206-213* (vid. pág. 155).

Rao, C., A. Yilmaz y M. Shah (2002). "View-Invariant Representation and Recognition of Actions". En: *Journal of Computer Vision, 50(2)* (vid. pág. 150).

Rasmussen, C.E. (2006). "Gaussian Processes for Machine Learning". En: *MIT Press: London, UK* (vid. pág. 32).

Rehg, J.M. y T. Kanade (1995). "Model-Based Tracking of Self-Occluding Articulated Objects". En: *Proceedings of the International Conference on Computer Vision, Cambridge, MA, USA; pp. 612-617* (vid. pág. 32).

Ren, X., A.C. Berg y J. Malik (2005). "Recovering Human Body Configurations using Pairwise Constraints between Parts". En: *Proc. IEEE Int. Conf. of Computer Vision. Vol 1. pp 824-831* (vid. pág. 140).

Ricolfe, C. y A. J. Sanchez (2011). "Optimal conditions for camera calibration using a planar template". En: *Image Processing (ICIP), 2011 18th IEEE International Conference on (pp. 853-856). IEEE* (vid. pág. 36).

Ricolfe, C., A. J. Sanchez y A. Valera (2013). "Efficient lens distortion correction for decoupling in calibration of wide angle lens cameras". En: *IEEE Sensors Journal, 13(2), 854-863* (vid. pág. 36).

Ricolfe, C., A.J. Sanchez y E. Martinez (2011). "Calibration of a wide angle stereoscopic system". En: *OPTICS LETTERS, ISSN 0146-9592, pag 3064-3067* (vid. pág. 119).

— (2012). "Accurate calibration with highly distorted images". En: *APPLIED OPTICS, ISSN 0003-6935, pag 89-101* (vid. pág. 119).

Rittscher, J., A. Blake y S.J. Roberts (2002). "Towards the Automatic Analysis of Complex Human Body Motions". En: *Image and Vision Computing, 20 (12). pp 905-916* (vid. pág. 148).

Rius, I. y col. (2009). "Action-Specific Motion Prior for Efficient Bayesian 3D Human Body Tracking". En: *Pattern Recognit., 42, 2907-2921* (vid. pág. 18).

Roberts, T.J., S.J. McKenna e I.W. Ricketts (2002). "Adaptive Learning of Statistical Appearance Models for 3D Human Tracking". En: *British Machine Vision Conference, Cardiff, UK. pp 333-342* (vid. pág. 131).

— (2004). "Human Pose Estimation using Learnt Probabilistic Region Similarities and Partial Configurations". En: *Proc. European Conf. of Computer Vision, LNCS 3024, Springer-Verlag. pp 291-303* (vid. págs. 12, 140).

Robertson, N. e I. Reid (2005). "Behaviour understanding in video: a combined method". En: *Proc. Int. Conf. on Computer Vision, (ICCV05). Vol 1. pp 808-815* (vid. pág. 148).

Rogez, G., C. Orrite-Urunuela y J. Martínez-del Rincón (2008). "A Spatio-Temporal 2D-Models Framework for Human Pose Recovery in Monocular Sequences". En: *Pattern Recognit., 41, 2926-2944* (vid. pág. 30).

Rogez, G. y col. (2008). "Randomized Trees for Human Pose Detection". En: *Proceedings of the Computer Vision and Pattern Recognition, Anchorage, AK, USA; pp. 1-8* (vid. págs. 27, 29, 30).

Ronfard, R., C. Schmid y B. Triggs (2002a). "Learning to Parse Pictures of People". En: *Proceedings of the European Conference on Computer Vision, Copenhagen, Denmark; pp. 700-714* (vid. pág. 20).

— (2002b). "Learning to Parse Pictures of People". En: *Proc. European Conf. of Computer Vision, LNCS, Springer-Verlag. Vol 253. pp 700-714* (vid. págs. 131, 140).

Rosales, R. y S. Sclaroff (2006). "Combining Generative and Discriminative Models in a Framework for Articulated Pose Estimation". En: *Int. J. Comput. Vis., 67, 251-276* (vid. pág. 25).

Rosales, R. y col. (2001). "3D Hand Pose Reconstruction Using Specialized Mappings". En: *Proceedings of the IEEE International Conference on Computer Vision, Vancouver, BC, Canada; pp. 378-385* (vid. pág. 21).

Roth, D., P. Doubek y L.V. Gool (2005). "Bayesian Pixel Classification for Human Tracking". En: *IEEE Workshop on Motion and Video Computing (MOTION-05), Breckenridge, Colorado. pp 78-83* (vid. págs. 137, 138).

Roth, S., L. Sigal y M. Black (2004). "Gibbs Likelihoods for Bayesian Tracking". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Washington, DC, USA; pp. 886-893* (vid. pág. 21).

Rothrock, B., S. Park y S.C. Zhu (2013). "Integrating Grammar and Segmentation for Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 3214-3221* (vid. págs. 27, 29).

Sabzmeydani, P. y G. Mori (2007). "Detecting Pedestrians by Learning Shapelet Features". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN, USA; pp. 1-8* (vid. págs. 11, 30).

Samitha, H., H. Mehrtash y P. Fatih (2017). "Going deeper into action recognition: A survey". En: *Image and Vision Computing 60 - 4-21* (vid. pág. 147).

Sand, P., L. McMillan y J. Popovic (2003). "Continuous Capture of Skin Deformation". En: *ACM Trans. Graph., 22, 578-586* (vid. pág. 17).

Sapp, B., C. Jordan y B. Taskar (2010). "Adaptive Pose Priors for Pictorial Structures". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA; pp. 422-429* (vid. págs. 14, 28).

Sapp, B. y B. Taskar (2013). "Modec: Multimodal Decomposable Models for Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 3674-3681* (vid. pág. 15).

Sapp, B., A. Toshev y B. Taskar (2010). "Cascaded Models for Articulated Pose Estimation". En: *Proceedings of the European Conference on Computer Vision, Crete, Greece; pp. 406-420* (vid. págs. 10, 12, 15, 28).

Sapp, B., D. Weiss y B. Taskar (2011). "Parsing human motion with stretchable models". En: *IEEE Conference on Computer Vision and Pattern Recognition* (vid. págs. 29, 43).

Sariyildiz, E. (2009). "A New Approach to Inverse Kinematic Solutions of Serial Robot Arms Based on Quaternions in the Screw Theory Framework". En: *MsC Thesis Mechatronics Engineering Graduate Program, Istanbul Thecnical University Mechatronics Engineering* (vid. pág. 202).

Sato, K. y J.K. Aggarwal (2001). "Tracking and Recognizing Two-person Interactions in Outdoor Image Sequences". En: *Workshop on Multi-Object Tracking, Vancouver, Canada. Pp 87-94* (vid. pág. 149).

"Scene Constraints-aided Tracking of Human Body" (2016). En: *Available online: http://ieeexplore.ieee.org/ document/855813/* (vid. pág. 13).

Schneiderman, H. y T. Kanade (2004). "Object detection using the statistics of parts". En: *IJCV, 56(3):151-177* (vid. pág. 155).

Schwarz, M., H. Schulz y S. Behnke (2015). "RGB-D object recognition and pose estimation based on pre-trained convolutional neural network features". En: *Robotics and Automation (ICRA), IEEE International Conference* (vid. pág. 22).

Scovanner, P., S. Ali y M. Shah (2007). "A 3-Dimensional Sift Descriptor and Its Application to Action Recognition". En: *Proceedings of the International*

Conference on Multimedia, Augsburg, Bavaria, Germany; pp. 357-360 (vid. pág. 11).

Sedai, S., M. Bennamoun y D. Huynh (2009). "Context-Based Appearance Descriptor for 3D Human Pose Estimation from Monocular Images". En: *Proceedings of the International Conference on Digital Image Computing: Techniques and Applications, Melbourne, Australia; pp. 484-491* (vid. pág. 20).

— (2011). "Evaluating Shape and Appearance Descriptors for 3D Human Pose Estimation". En: *Proceedings of the IEEE Conference on Industrial Electronics and Applications, Beijing, China; pp. 293-298* (vid. pág. 20).

Sedai, S., M. Bennamoun y D.Q. Huynh (2013). "Discriminative Fusion of Shape and Appearance Features for Human Pose Estimation". En: *Pattern Recognit., 46, 3223-3237* (vid. pág. 13).

Sedai, S. y col. (2010). "Localized Fusion of Shape and Appearance Features for 3D Human Pose Estimation". En: *Proceedings of the British Machine Vision Conference, Aberystwyth, UK; pp. 1-10* (vid. pág. 13).

Selig, J.M. (2004). "Geometrical Fundamentals Of Robotics". En: *Springer November 2nd edition, pp. 9-20, 25-34,41-49,81-90* (vid. pág. 206).

Serre, T., L. Wolf y T. Poggio (2005). "Object Recognition with Features Inspired by Visual Cortex". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA; pp. 994-1000* (vid. pág. 11).

Shakhnarovich, G., P. Viola y T. Darrell (2003). "Fast Pose Estimation with Parameter-Sensitive Hashing". En: *Proceedings of the IEEE International Conference on Computer Vision, Madison, WI, USA; pp. 750-757* (vid. págs. 11, 30).

Shotton, J. y col. (2011). "Real-time human pose recognition in parts from single depth images". En: *International Conference on Computer Vision and Pattern Recognition (CVPR)* (vid. págs. 23, 24, 145).

Shotton, J. y col. (2013a). "Efficient human pose estimation from single depth images". En: *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 35(12):2821-2840* (vid. pág. 107).

Shotton, J. y col. (2013b). "Real-Time Human Pose Recognition in Parts from Single Depth Images". En: *Commun. ACM, 56, 116-124.* (Vid. pág. 8).

— (2013c). "Real-time human pose recognition in parts from single depth images". En: *Communications of the ACM, 56(1):116-124* (vid. págs. 36, 107, 108, 111, 116).

Sidenbladh, H. y M.J. Black (2001). "Learning Image Statistics for Bayesian Tracking". En: *Proceedings of the IEEE International Conference on Computer Vision, Vancouver, BC, Canada; pp. 709-716* (vid. pág. 13).

— (2003). "Learning the Statistics of People in Images and Video". En: *Int. Journal of Computer Vision, 54(1/2/3). pp 183-209* (vid. pág. 131).

Sidenbladh, H., M.J. Black y D.J. Fleet (2010). "Stochastic Tracking of 3D Human Figures Using 2D Image Motion". En: *Proceedings of the European Conference on Computer Vision, Heraklion, Crete, Greece; pp. 702-718* (vid. pág. 25).

Sidenbladh, H., M.J. Black y L. Sigal (2002). "Implicit Probabilistic Models of Human Motion for Synthesis and Tracking". En: *European Conference on Computer Vision, Copenhagen, Denmark. pp 784-800* (vid. pág. 146).

Sidenbladh, H., F. De la Torre y M.J. Black (2000). "A Framework for Modeling the Appearance of 3D Articulated Figures". En: *Proceedings of the IEEE International Conference on Automatic Face and Gesture Recognition, Grenoble, France; pp. 368-375* (vid. pág. 16).

Sigal, L., A. Balan y M.J. Black (2007). "Combined Discriminative and Generative Articulated Pose and Non-Rigid Shape Estimation". En: *Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada; pp. 1337-1344* (vid. pág. 25).

Sigal, L. y M.J. Black (2006a). "Measure Locally, Reason Globally: Occlusion-Sensitive Articulated Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, USA; pp. 2041-2048* (vid. pág. 29).

— (2006b). "Predicting 3D People from 2D Pictures". En: *Proceedings of the International Conference on Articulated Motion and Deformable Objects, Port d'Andratx, Mallorca, Spain; pp. 185-195* (vid. pág. 20).

Sigal, L. y col. (2004). "Tracking Loose-limbed People". En: *Proc. IEEE Computer Vision and Pattern Recognition. Vol 1. pp 421-428* (vid. pág. 146).

Sigal, L. y col. (2012). "Loose-Limbed People: Estimating 3D Human Pose and Motion Using Non-Parametric Belief Propagation". En: *Int. J. Comput. Vis., 98, 15-48* (vid. pág. 32).

Simo-Serra, E. y col. (2013). "A Joint Model for 2D and 3D Pose Estimation from a Single Image". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 3634-3641* (vid. pág. 21).

Slama, R., H. Wannous y M. Daoudi (2013). "Extremal Human Curves: A New Human Body Shape and Pose Descriptor". En: *Proceedings of the IEEE International Conference and Workshops on Automatic Face and Gesture Recognition, Shanghai, China; pp. 1-6* (vid. pág. 11).

Sminchisescu, C. y A. Jepson (2014). "Generative Modeling for Continuous Non-Linearly Embedded Visual Inference". En: *Proceedings of the International Conference on Machine Learning, Beijing, China; pp. 759-766* (vid. pág. 21).

Sminchisescu, C. y A. Telea (2002). "Human Pose Estimation from Silhouettes-A Consistent Approach Using Distance Level Sets". En: *Proceedings of the International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, Bory, Czech Republic* (vid. págs. 10, 13, 25).

Sminchisescu, C. y B. Triggs (2001). "Covariance Scaled Sampling for Monocular 3D Body Tracking". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Kauai, HI, USA; pp. I-447-I-454* (vid. pág. 12).

— (2003). "Estimating Articulated Human Motion with Covariance Scaled Sampling". En: *Int. Journal of Robotics Research, 22(5). pp 371-391* (vid. págs. 18, 144).

Sminchisescu, C. y col. (2005). "Discriminative Density Propagation for 3D Human Motion Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA; pp. 390-397* (vid. págs. 10, 20, 24).

Sminchisescu, C. y col. (2011). "Feature-Based Pose Estimation". En: *Visual Analysis of Humans; Springer: London, UK; pp. 225-251* (vid. pág. 20).

Smith, K., D.G. Perez y J.M. Odobez (2005). "Using Particles to Track Varying Numbers of Interacting People". En: *Computer Vision and Pattern Recognition, San Diego, CA, USA. Vol 1. pp 962-969* (vid. pág. 137).

Song, Y., L. Goncalves y P. Perona (2003). "Unsupervised Learning of Human Motion". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(7). pp 814-827.* (Vid. pág. 130).

Starck, J. y A. Hilton (2003). "Model-based multiple view reconstruction of people". En: *IEEE International Conference on Computer Vision, pp 915-922* (vid. págs. 130, 131).

— (2005). "Spherical Matching for Temporal Correspondence of Non-Rigid Surfaces". En: *IEEE Int. Conf. Computer Vision, pp 1387-1394* (vid. pág. 142).

Stauffer, C. y W.E.L. Grimson (1998). "Adaptive Background Mixture Models for Real-Time Tracking". En: *Computer Vision and Pattern Recognition, Santa Barbara, CA, USA* (vid. pág. 132).

— (2000). "Learning patterns of activity using real-time tracking". En: *IEEE Trans. Pattern Analysis and Machine Intelligence, 22(8). pp 747-757* (vid. pág. 147).

Stoll, C. y col. (2011). "Fast Articulated Motion Tracking Using a Sums of Gaussians Body Model". En: *Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain; pp. 951-958* (vid. pág. 24).

Sun, L. y col. (2014). "Motionlet LLC Coding for Discriminative Human Pose Estimation". En: *Multimed. Tools Appl., 73, 327-344* (vid. págs. 13, 21, 33).

Sun, M. y S. Savarese (2011). "Articulated Part-Based Model for Joint Object Detection and Pose Estimation". En: *Proceedings of the IEEE International Conference on Computer Vision, Barcelona, Spain; pp. 723-730* (vid. pág. 15).

Sun, M. y col. (2012). "An Efficient Branch-and-Bound Algorithm for Optimal Human Pose Estimation". En: *Proceedings of the IEEE Conference on Com-*

puter Vision and Pattern Recognition, Providence, RI, USA; pp. 1616-1623 (vid. pág. 27).

Tan, Q. y J.G. Balchen (1993). "General Quaternion Transformation Representation For Robot Application Systems". En: *Man and Cybernetics, Systems Engineering in the Service of Humans, 17-20 Oct.pp.319-324 Vol.3* (vid. pág. 204).

Tashiro, K. y col. (2014). "Refinement of Ontology-Constrained Human Pose Classification". En: *Proceedings of the IEEE International Conference on Semantic Computing, Newport Beach, CA, USA; pp. 60-67* (vid. pág. 15).

Taylor, J. y col. (2012). "The Vitruvian Manifold: Inferring Dense Correspondences for One-Shot Human Pose Estimation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA; pp. 103-110* (vid. págs. 21, 23, 24).

Thalmann, M.N. y H. Seo (2004). "Data-driven approaches to digital human modeling". En: *Proc. 2nd International Symposium on 3D Data Processing, Visualization, and Transmission, Thessalonica, Greece, IEEE Computer Society Press. pp 380-387* (vid. pág. 131).

Thanh Nguyen, D. y col. (2010). "Object detection using Non-Redundant Local Binary Patterns". En: *IEEE International Conference on Image Processing, Page 4609* (vid. pág. 133).

Tian, J., L. Li y W. Liu (2014). "Multi-Scale Human Pose Tracking in 2D Monocular Images". En: *J. Comput. Commun., 2, 78* (vid. pág. 32).

Tian, T.P., R. Li y S. Sclaroff (2005). "Articulated Pose Estimation in a Learned Smooth Space of Feasible Solutions". En: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, San Diego, CA, USA*. (Vid. pág. 32).

Tian, T.P. y S. Sclaroff (2010). "Fast Globally Optimal 2D Human Detection with Loopy Graph Models". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA; pp. 81-88* (vid. pág. 27).

Tian, Y., C.L. Zitnick y S.G. Narasimhan (2012). "Exploring the Spatial Hierarchy of Mixture Models for Human Pose Estimation". En: *Proceedings of the European Conference on Computer Vision, Firenze, Italy; pp. 256-269* (vid. págs. 15, 28).

Tian, Y. y col. (2010). "Latent Gaussian Mixture Regression for Human Pose Estimation". En: *Proceedings of the Asian Conference on Computer Vision, Queenstown, New Zealand; pp. 679-690* (vid. pág. 32).

Tompson, J.J. y col. (2014). "Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation". En: *Proceedings of the Advances in Neural Information Processing Systems, Beijing, China; pp. 1799-1807* (vid. págs. 22, 23).

Torres, F. y W.G. Kropatsch (2013). "Top-down 3D Tracking and Pose Estimation of a Die Using Check-Points". En: *Proceedings of the Computer Vision Winter Workshop, Hernstein, Austria.* (Vid. pág. 30).

Toshev, A. y C. Szegedy (2014). "Deeppose: Human Pose Estimation via Deep Neural Networks". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA; pp. 1653-1660* (vid. págs. 22, 23).

Toyama, K. y A. Blake (2002). "Probabilistic Tracking with Exemplars in a Metric Space". En: *Int. J. Comput. Vis., 48, 9-19* (vid. pág. 20).

Tran, D. y D. Forsyth (2010). "Improved human parsing with a full relational model". En: *European Conference on Computer Vision* (vid. pág. 99).

Tsochantaridis, I. y col. (2004). "Support vector machine learning for interdependent and structured output spaces". En: *International Conference on Machine Learning* (vid. pág. 47).

Urtasun, R. y T. Darrell (2008). "Sparse Probabilistic Regression for Activity-Independent Human Pose Inference". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA; pp. 1-8* (vid. págs. 20, 29).

Urtasun, R., D.J. Fleet y P. Fua (2005). "Monocular 3D Tracking of the Golf Swing". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA, USA; pp. 932-938* (vid. pág. 32).

— (2006). "3D People Rracking with Gaussian Process Dynamical Models". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, USA; pp. 238-245* (vid. pág. 32).

Urtasun, R., D.J. Fleet y N.D. Lawrence (2007). "Modeling Human Locomotion with Topologically Constrained Latent Variable Models". En: *Human Motion Understanding, Modeling, Capture and Animation; Springer: Heidelberg, Germany; pp. 104-118* (vid. pág. 32).

Urtasun, R. y P. Fua (2004a). "3D Human Body Tracking Using Deterministic Temporal Motion Models". En: *Proceedings of the European Conference on Computer Vision, Prague, Czech Republic; pp. 92-106* (vid. pág. 16).

— (2004b). "3D Human Body Tracking using Deterministic Temporal Motion Models". En: *Proc. European Conf. of Computer Vision, LNCS, Springer-Verlag* (vid. pág. 146).

Urtasun, R. y col. (2005). "Priors for People Tracking from Small Training Sets". En: *Proceedings of the IEEE International Conference on Computer Vision, Beijing, China; pp. 403-410* (vid. pág. 18).

Utsumi, A. y N. Tetsutani (2002). "Human Detection using Geometrical Pixel Value Structures". En: *International Conference on Automatic Face and Gesture Recognition, Washington D.C., USA. pp 34-39* (vid. pág. 134).

Vecchio, D.D., R.M. Murray y P. Perona (2003). "Decomposition of Human Motiom into Dynamics-based Primitives with Application to Drawing Tasks". En: *Automatica. Vol 39. pp 2085-2098* (vid. pág. 150).

Viola, P., M.J. Jones y D. Snow (2005). "Detecting Pedestrians Using Patterns of Motion and Appearance". En: *International Journal of Computer Vision, 63(2). pp 734-741* (vid. págs. 10, 30, 133).

Wachter, S. y H.H. Nagel (1997). "Tracking of Persons in Monocular Image Sequences". En: *Workshop on Motion of Non-Rigid and Articulated Objects, Puerto Rico, USA. pp 2-9* (vid. pág. 139).

— (1999). "Tracking Persons in Monocular Image Sequences". En: *Computer Vision and Image Understanding, 74(3). pp 174-192* (vid. págs. 142, 144).

Wang, C., Y. Wang y A. Yuille (2013). "An Approach to Pose-based Action Recognition". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 915-922* (vid. pág. 26).

Wang, F. e Y. Li (2013). "Beyond physical connections: Tree models in human pose estimation". En: *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on, pages 596-603* (vid. págs. 29, 36).

Wang, H. y D. Koller (2011). "Multi-Level Inference by Relaxed Dual Decomposition for Human Pose Segmentation". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA; pp. 2433-2440* (vid. pág. 26).

Wang, J. y col. (2010). "Locality-Constrained Linear Coding for Image Classification". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA; pp. 3360-3367* (vid. págs. 13, 21, 33).

Wang, J.M., D.J. Fleet y A. Hertzmann (2007). "Multifactor Gaussian Process Models for Style-Content Separation". En: *Proceedings of the International Conference on Machine Learning, Las Vegas, NV, USA; pp. 975-982* (vid. págs. 18, 33).

Wang, L. y col. (2003a). "Fusion of Static and Dynamic Body Biometrics for Gait Recognition". En: *International Conference on Computer Vision, Nice, France. Vol 2. pp 1449-1454* (vid. pág. 149).

Wang, L. y col. (2003b). "Silhouette Analysis-Based Gait Recognition for Human Identification". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(12). pp 1501-1518* (vid. págs. 148, 149).

Wang, Y. y G. Mori (2008). "Multiple Tree Models for Occlusion and Spatial Constraints in Human Pose Estimation". En: *Proceedings of the European Conference on Computer Vision, Marseille, France; pp. 710-724* (vid. pág. 15).

Wang, Y., D. Tran y Z. Liao (2011). "Learning Hierarchical Poselets for Human Parsing". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA; pp. 1705-1712* (vid. pág. 29).

Wang, Y. y col. (2012). "Discriminative Hierarchical Part-Based Models for Human Parsing and Action Recognition". En: *J. Mach. Learn. Res., 13, 3075-3102* (vid. pág. 31).

Wang, Y.K. y K.Y. Cheng (2010). "3D Human Pose Estimation by an Annealed Two-Stage Inference Method". En: *Proceedings of the International Conference on Pattern Recognition, Istanbul, Turkey; pp. 535-538* (vid. pág. 18).

Wei, X. y J. Chai (2010). "Videomocap: Modeling Physically Realistic Human Motion from Monocular Video Sequences". En: *ACM Trans. Graph., 29, 42* (vid. pág. 18).

Wei, X.K. y J. Chai (2009). "Modeling 3D Human Poses from Uncalibrated Monocular Images". En: *Proceedings of the IEEE International Conference on Computer Vision, Kyoto, Japan; pp. 1873-1880* (vid. pág. 25).

Weinland, D., R. Ronfard y E. Boyer (2011). "A Survey of Vision-based Methods for Action Representation, Segmentation and Recognition". En: *Comput. Vis. Image Underst, 115, 224-241.* (Vid. pág. 9).

Weinrich, C., M. Volkhardt y H.M. Gross (2013). "Appearance-based 3D Upper-Body Pose Estimation and Person Re-Identification on Mobile Robots". En: *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, Manchester, UK; pp. 4384-4390* (vid. pág. 10).

Weiss, D., B. Sapp y B. Taskar (2010). "Sidestepping Intractable Inference with Structured Ensemble Cascades". En: *Proceedings of the Advances in Neural Information Processing Systems, Vancouver, BC, Canada; pp. 2415-2423* (vid. págs. 28, 30).

Wren, C.R. y col. (1997). "Pfinder: Real-Time Tracking of the Human Body". En: *Transactions on Pattern Analysis and Machine Intelligence, 19(7). pp 780-785* (vid. págs. 10, 139).

Wright, J. y col. (2009). "Robust Face Recognition via Sparse Representation". En: *IEEE Trans. Pattern Anal. Mach. Intell., 31, 210-227* (vid. pág. 24).

Wu, B. y R. Nevatia (2005). "Detection of Multiple, Partially Occluded Humans in a Single Image by Bayesian Combination of Edgelet Part Detection". En: *International Conference on Computer Vision, Beijing, China. pp 90-97* (vid. págs. 135, 140).

Wu, Y., J. Lin y T.S. Huang (2005). "Analyzing and Capturing Articulated Hand Motion in Image Sequences". En: *IEEE Trans. Pattern Anal. Mach. Intell., 27, 1910-1922* (vid. págs. 11, 30).

Xiao, Y., H. Lu y S. Li (2012). "Posterior Constraints for Double-Counting Problem in Clustered Pose Estimation". En: *Proceedings of the IEEE International Conference on Image Processing, Orlando, FL, USA; pp. 5-8* (vid. pág. 15).

Xie, J. y col. (2007). "Inverse Kinematics Problem for 6-DOF Sapce Manipulator Based On The Theory of Screw". En: *International Conference on Robotics and Biomimetics,* (vid. pág. 112).

Xu, L.Q. y P. Puig (2005). "A Hybrid Blob and Appearance-Based Framework for Multi-Object Tracking through Complex Occlusions". En: *Int. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, Beijing, China. pp 73-80* (vid. págs. 134, 137, 138).

Yacoob, Y. y M.J. Black (1998). "Parameterized Modeling and Recognition of Activities". En: *Proceedings of the International Conference on Computer Vision, Bombay, India; pp. 120-127* (vid. pág. 30).

Yam, C., M. Nixon y J. Carter (2002). "On the Relationship of Human Walking and Running: Automatic Person Identification by Gait". En: *International Conference on Pattern Recognition, Quebec, Canada. Vol 1. pp 287-290* (vid. pág. 148).

Yang, C., R. Duraiswami y L. Davis (2005). "Fast Multiple Object Tracking via a Hierarchical Particle Filter". En: *International Conference on Computer Vision, Beijing, China. Vol 1. pp 212-219* (vid. pág. 134).

Yang, D.B., H.H.G. Banos y L.J. Guibas (2003). "Counting People in Crowds with a Real-Time Network of Simple Image Sensors". En: *International Conference on Computer Vision, Nice, France. Vol 1. pp 122-129* (vid. págs. 135, 136).

Yang, M.H. y A. Bissacco (2010). "Fast Human Pose Estimation Using Appearance and Motion via Multi-Dimensional Boosting Regression". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, San Francisco, CA, USA; pp. 1-8* (vid. págs. 13, 30).

Yang, Y. y D. Ramanan (2011). "Articulated Pose Estimation with Flexible Mixtures-of-Parts". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Colorado Springs, CO, USA; pp. 1385-1392* (vid. págs. 14, 27).

— (2013). "Articulated human detection with flexible mixtures of parts". En: *Pattern Analysis and Machine Intelligence, IEEE Transactions on, 35(12):2878-2890* (vid. págs. 35, 36, 49, 52, 53, 118, 153).

Yang, Y., I. Saleemi y M. Shah (2013). "Discovering Motion Primitives for Unsupervised Grouping and One-Shot Learning of Human Actions, Gestures, and Expressions". En: *IEEE Trans. Pattern Anal. Mach. Intell., 35, 1635-1648* (vid. pág. 21).

Yao, A., J. Gall y L. Van Gool (2012). "Coupled Action Recognition and Pose Estimation from Multiple Views". En: *Int. J. Comput. Vis, 100, 16-37.* (Vid. págs. 9, 21).

Yao, A. y col. (2011). "Does Human Action Recognition Benefit from Pose Estimation?" En: *In Proceedings of the British Machine Vision Conference, Dundee, UK; pp. 67.1-67.11.* (Vid. pág. 9).

Yi, H., D. Rajan y L.T. Chia (2004). "A new motion histogram to index motion content in video segments". En: *Pattern Recognition Letters, 26. pp 1221-1231* (vid. pág. 149).

Yilmaz, A. y M. Shah (2005). "Actions sketch: A novel action representation". En: *Proc. IEEE Conf. on Computer Vision and Pattern Recognition, CVPR05. Vol 1. pp 984-989* (vid. pág. 149).

Yu, H. y col. (2005). "Human motion recognition based on neural networks". En: *Proc. Int. Conf. on Communications, Circuits and Systems, ICCCS05. Vol 2. pp 979-982* (vid. pág. 149).

Yu, T.H., T.K. Kim y R. Cipolla (2013). "Unconstrained Monocular 3D Human Pose Estimation by Action Detection and Cross-Modality Regression Forest". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA; pp. 3642-3649* (vid. pág. 30).

Yu, X. y S.X. Yang (2005). "A study of motion recognition from video sequences". En: *Computing and Visualization in Science, 8. pp 19-25* (vid. pág. 151).

Yue-sheng, T. y X. Ai-ping (2008). "Extension of the Second Paden-Kahan Subproblem and its Application in the Inverse Kinematics of a Manipulator". En: *ISBN: 978-1-4244-1676-9/08* (vid. pág. 217).

Yuille, A. y A. Rangarajan (2003). "The concave-convex procedure". En: *Neural Computation, vol. 15, no. 4, pp. 915-936* (vid. pág. 48).

Zahn, C.T. y R.Z. Roskies (1972). "Fourier Descriptors for Plane Closed Curves". En: *IEEE Trans. Comput., 100, 269-281* (vid. pág. 11).

Zhang, J. y col. (2007). "Local features and kernels for classification of texture and object categories: A comprehensive study". En: *IJCV, 73(2):213-238* (vid. pág. 155).

Zhang, W., L. Shang y A.B. Chan (2014). "A Robust Likelihood Function for 3D Human Pose Tracking". En: *IEEE Trans. Image Process., 23, 5374-5389* (vid. pág. 24).

Zhang, W. y col. (2014). "A Latent Clothing Attribute Approach for Human Pose Estimation". En: *Proceedings of the Asian Conference on Computer Vision, Singapore; pp. 146-161* (vid. pág. 20).

Zhao, L. y C.E. Thorpe (2000). "Stereo -and Neural Network-Based Pedestrian Detection". En: *IEEE Transactions on Intelligent Trasnportation Systems, 1(3). pp 148-154* (vid. pág. 135).

Zhao, T. y R. Nevatia (2004a). "Tracking Multiple Humans in Complex Situations". En: *IEEE Transactions on Pattern Analysis and Machine Intelligence, 26(9). pp 1028-1221* (vid. pág. 132).

— (2004b). "Tracking Multiple Humans in Crowded Environments". En: *Computer Vision and Pattern Recognition, Washington DC, USA. Vol 2. pp 406-413* (vid. pág. 137).

Zhao, X. y col. (2008). "Discriminative Estimation of 3D Human Pose Using Gaussian Processes". En: *Proceedings of the International Conference on Pattern Recognition, Tampa, FL, USA; pp. 1-4* (vid. págs. 20, 32).

Zhu, Q. y col. (2006). "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, New York, NY, USA; pp. 1491-1498* (vid. págs. 9, 11, 30).

Zhu, Y., B. Dariush y K. Fujimura (2008). "Controlled Human Pose Estimation from Depth Image Streams". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA; pp. 1-8* (vid. pág. 24).

Zuffi, S. y M.J. Black (2015). "The Stitched Puppet: A Graphical Model of 3D Human Shape and Pose". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA; pp. 3537-3546* (vid. pág. 17).

Zuffi, S., O. Freifeld y M.J. Black (2012). "From Pictorial Structures to Deformable Structures". En: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA; pp. 3546-3553* (vid. págs. 12, 14).