



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería Informática

Automatización del proceso de obtención de imágenes
GigaPixel

Trabajo Fin de Grado

Grado en Ingeniería Informática

AUTOR/A: Nacher Verdeguer, Francisco Javier

Tutor/a: Bonastre Pina, Alberto Miguel

Cotutor/a: Cabezos Bernal, Pedro Manuel

Cotutor/a: Capella Hernández, Juan Vicente

CURSO ACADÉMICO: 2021/2022

*“A mis padres y a todas las personas que también me quieren y han creído en mí,
aunque yo no me dé cuenta, gracias por estar siempre ahí.”*

Resumen

El siguiente trabajo tiene como principal objetivo la automatización del diseño e implementación de un sistema de control de una plataforma robotizada de tipo tijera, desarrollada en el ámbito del Proyecto de Investigación denominado Captura Fotográfica de resolución gigapíxel para la documentación y divulgación del patrimonio pictórico, financiado con ayuda a Primeros Proyectos de Investigación (PAID-06-18) del Vicerrectorado de Investigación, Innovación y transferencia de la Universitat Politècnica de València. El sistema consiste en el desarrollo de distintos módulos informáticos como son una aplicación de control para dispositivos móviles Android, que permitirá gobernar de forma remota y establecer el movimiento programado de la plataforma. Esto permitirá el posicionado libre de la cámara que irá instalada sobre la bandeja superior de la plataforma, además del posicionamiento programado de la cámara para la captura automatizada de las distintas fotografías que compondrán el mosaico que sirve para componer la imagen gigapíxel. Esta programación del posicionamiento deberá calcularse en función de diversos parámetros, como la distancia focal, el tamaño de la obra y la distancia entre la plataforma y el cuadro. En segundo lugar, se desarrollará la programación del módulo receptor, que será implementado en una placa Arduino Mega. Este módulo atenderá las órdenes enviadas por el módulo de control y accionará los motores paso a paso de la plataforma, lo que permitirá el ascenso o descenso de la tijera y el avance o retroceso en el sentido horizontal. La comunicación entre estos dos módulos se realizará de forma inalámbrica a través de una conexión bluetooth. El trabajo incluye además todo el conexionado y cableado del hardware integrante de la plataforma y la calibración y lectura de los sensores ultrasónicos de medición que permiten situar la plataforma automatizada a la distancia óptima de captura.

Palabras clave: gigapíxel, Android, Arduino, fotografías.

Abstract

The goal of this project is the design and implementation of a control system for a scissor-type robotic platform, developed within the scope of the Research Project called Gigapixel Resolution Photographic Capture for the documentation and dissemination of pictorial heritage, financed with the help of Primeros Research Projects (PAID-06-18) of the Vice President for Research, Innovation and Transfer of the Universitat Politècnica de València. The system consists of the development of different computer modules such as a control application for Android mobile devices, which will allow remote control and establish the programmed movement of the platform. This will enable the free positioning of the camera that will be installed on the upper tray of the platform, in addition to the programmed positioning of the camera for the automated capture of the different photographs that will make up the mosaic that is used to compose the gigapixel image. This positioning programming must be calculated based on various parameters, such as the focal length, the size of the painting and the distance between the platform and the frame. Secondly, the programming of the receiver module will be developed, which will be implemented in an Arduino Mega board. This module will respond to the orders sent by the control module and will activate the stepper motors of the platform, which will allow the lifting or lowering of the scissors and the forward or backward movement in the horizontal direction. The communication between these two modules will be carried out wirelessly through a bluetooth connection. The project also includes all the connection and wiring of the hardware that is part of the platform and the calibration and reading of the ultrasonic measurement sensors that allow the automated platform to be positioned at the optimal capture distance.

Keywords : gigapíxel, Android, Arduino, pictures.

Resum

El següent treball té com a principal objectiu el disseny i la implementació d'un sistema de control d'una plataforma robotitzada de tipus tiora, desenvolupada en l'àmbit del Projecte de Recerca anomenat Captura Fotogràfica de resolució gigapíxel para la documentación y divulgación del patrimonio pictórico, finançat amb ajuda a Primers Projectes de Recerca (PAID-06-18) del Vicerectorat de Recerca, Innovació i transferència de la Universitat Politècnica de València. El sistema consisteix en el desenvolupament de diferents mòduls informàtics com són una aplicació de control per a dispositius mòbils Android, que permetrà governar de manera remota i establir el moviment programat de la plataforma. Això permetrà el posicionament lliure de la càmera que anirà instal·lada sobre la safata superior de la plataforma, a més del posicionament programat de la càmera per a la captura automatitzada de les diferents fotografies que compondran el mosaic que serveix per compondre la imatge gigapíxel. Aquesta programació del posicionament s'haurà de calcular en funció de diversos paràmetres, com ara la distància focal, la mida de l'obra i la distància entre la plataforma i el quadre. En segon lloc, es desenvoluparà la programació del mòdul receptor, que serà implementat a una placa Arduino Mega. Aquest mòdul atindrà les ordres enviades pel mòdul de control i accionarà els motors pas a pas de la plataforma, cosa que permetrà l'ascens o descens de les tiores i l'avenç o reculada en el sentit horitzontal. La comunicació entre aquests dos mòduls es realitzarà sense fils a través d'una connexió bluetooth. El treball inclou a més tot el connexionat i cablejat del maquinari integrant de la plataforma i el calibratge i lectura dels sensors ultrasònics de mesurament que permeten situar la plataforma automatitzada a la distància òptima de captura.

Paraules clau: gigapíxel, Android, Arduino, fotografies.

Tabla de contenidos

Contenido	
Resumen	ii
Abstract	iii
Resum	iv
Tabla de ilustraciones	vi
1 INTRODUCCION	1
1.1 Motivación	4
1.2 Objetivos	5
1.3 Metodología	5
1.4 Estructura de la memoria	18
2. ESTADO DEL ARTE	20
3. DESARROLLO DEL PROYECTO	24
3.1 La plataforma Robotizada	24
3.2 El Módulo Receptor Arduino	29
3.3 El Módulo Bluetooth	31
3.4 El Sensor Ultrasónico	32
3.5 Módulo Android	34
3.5.1 Especificación de requisitos	34
3.5.2 Diseño de la aplicación	36
3.5.3 Implementación de la aplicación	37
4. EXPERIMENTACION Y PRUEBAS	49
5. CONCLUSIONES Y TRABAJO FUTURO	56
6. REFERENCIAS	61
ANEXO I. OBJETIVOS DE DESARROLLO SOSTENIBLE	63

Tabla de ilustraciones

Ilustración 1. Sensor cámara fotográfica digital	2
Ilustración 2. Tamaños típicos de sensores digitales	3
Ilustración 3. Prototipo cámara AWARE-2	6
Ilustración 4. Prototipo cámara AWARE-10	7
Ilustración 5. Google Art Camera	8
Ilustración 6. Captura de punto de vista único	10
Ilustración 7. Captura de punto de vista múltiple paralela.....	11
Ilustración 8. Captura de punto de vista múltiple oblicua	11
Ilustración 9. Captura de la web del Museo Reina Sofía	21
Ilustración 10. Robot GIGAmacro Magnify2	22
Ilustración 11. Plataforma Robot y sus dimensiones	25
Ilustración 12. Cajas impresas en una impresora 3D.....	26
Ilustración 13. Esquema eléctrico del robot	27
Ilustración 14. Microstepping.....	28
Ilustración 15. Arduino Mega 2560	29
Ilustración 16. Seudocódigo aplicación Arduino	30
Ilustración 17. Módulo Bluetooth HC-06	32
Ilustración 18. Sensor ultrasónico HY-SRF05.....	33
Ilustración 19. Boceto Actividad Modo Manual	36
Ilustración 20. Boceto Actividad Modo Programado	36
Ilustración 21. Boceto Actividad Ajustes	37
Ilustración 22. Ciclo de vida de una Actividad Android	39
Ilustración 23. Eventos onPause() y onResume().....	41
Ilustración 24. Menú superior de la aplicación	41
Ilustración 25. Actividad Principal.....	43
Ilustración 26. Actividad de Preferencias.....	43
Ilustración 27. Pantalla uso manual.....	44
Ilustración 28. Pantalla programación	44
Ilustración 29. Ejemplo de movimientos calculados para el mosaico	45
Ilustración 30. Correspondencia entre pasos y milímetros.....	47
Ilustración 31. Ciclo de vida de un servicio Android.....	50
Ilustración 32. Cableado del robot.....	51
Ilustración 33. Instantánea de parte de la electrónica.....	52
Ilustración 34. Trabajos de soldadura.....	53

Ilustración 35. Pruebas de los motores.....	53
Ilustración 36. Chip esp8266	57
Ilustración 37. Chip ESP32.....	58
Ilustración 38. Arduino Uno comparado con ESP32	59
Ilustración 39. Cinemática impresora 3D	60

1 INTRODUCCION

Aunque las imágenes gigapíxel ya llevan tiempo entre nosotros, no se están explotando todas sus ventajas. Gran cantidad de museos están comenzando a digitalizar su catálogo de obras en resolución gigapíxel para poder estudiarlas y analizarlas con mayor facilidad, convirtiéndose además en una alternativa a la divulgación clásica de las mismas.

El uso de imágenes de resolución gigapíxel en el campo de las obras de arte, nos va a permitir observar las obras tan de cerca, que podremos ver el craquelado de los pigmentos y otros detalles que nos permitan evaluar el daño de las obras, sin ni siquiera moverlas del lugar donde están expuestas.

Esto resulta sumamente interesante, no sólo para los conservadores y estudiosos del arte, sino también para el público en general, ya que los resultados pueden ser difundidos a través de visitas virtuales que permiten al espectador adentrarse en la obra, de tal forma, que se pueden apreciar detalles imposibles de percibir a simple vista (Cabezos, 2021).

Una imagen gigapíxel es un mapa de bits compuesto por mil millones de píxeles, esto es 1000 veces la información capturada por una cámara digital de 1 megapíxel. Las cámaras fotográficas actuales profesionales disponen de sensores de alrededor de 40 megapíxeles. Por lo que para obtener imágenes de resolución gigapíxel es necesario una técnica de captura especial o una cámara que sea capaz de crear una imagen gigapíxel en un solo disparo (Gálvez, 2017).

El sensor de una cámara digital sería el equivalente a la película fotosensible utilizada en las cámaras tradicionales analógicas. Este sensor, está formado por millones de celdas fotosensibles (ver Ilustración 1) de un tamaño aproximado de 4 micrómetros (1 micrómetro es la milésima parte de 1 milímetro).

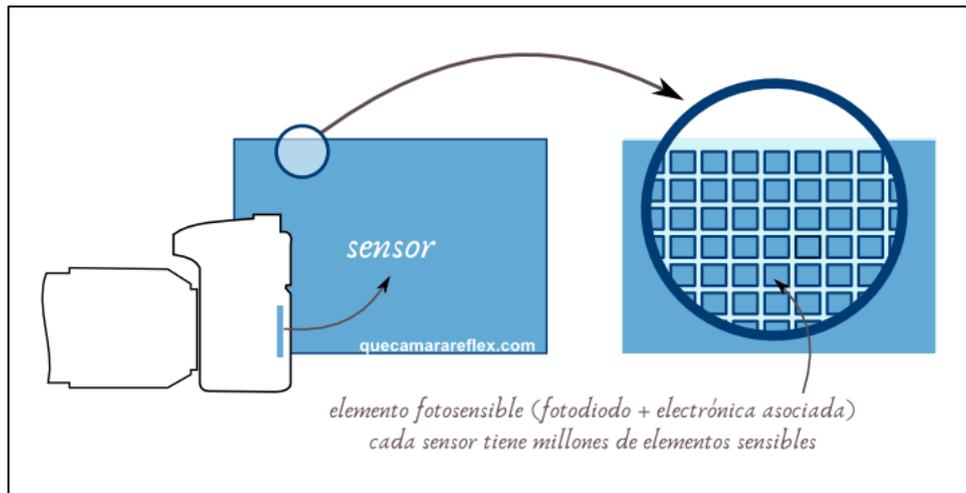


Ilustración 1. Sensor cámara fotográfica digital

Imagen obtenida de <https://quecamarareflex.com/como-funciona-el-sensor-de-una-camara-digital/>

Cada una de estas celdas se corresponden con un píxel y son las encargadas de captar la información de la luz que les llega (intensidad). Es importante tener en cuenta la sensibilidad ISO del sensor, ya que indica la cantidad de luz que debe recoger. Para detectar el color, estos sensores incluyen filtros ópticos (RGB) que descomponen la luz en tres componentes: rojo, verde y azul.

Por lo tanto, a mayor tamaño de sensor, mayor número de celdas fotosensibles y, por consiguiente, mayor número de píxeles tendrá la imagen captada. Aunque este parámetro por sí solo no es indicativo de mayor calidad de la imagen final. Los tamaños de sensores más comunes son (ver Ilustración 2):

- Sensor **Full Frame**, o sensor 35mm. Dimensiones: 36x24mm
- Sensor **APS-H**. Dimensiones: 28,7x19mm
- Sensor **APS-C** (Nikon, Pentax y Sony). Dimensiones: 23,4x15,6mm
- Sensor **APS-C** (Canon). Dimensiones: 22,2x14,8mm
- Sensor **Micro Cuatro Tercios**. Dimensiones: 17,3x13,8mm

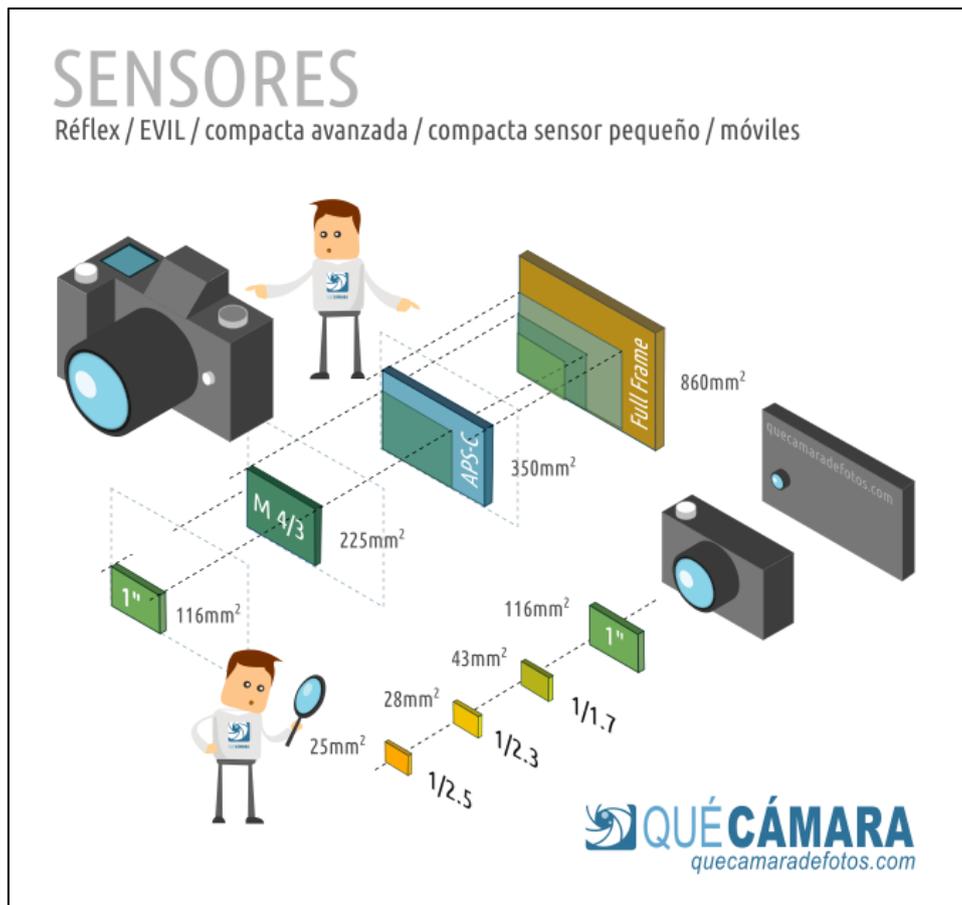


Ilustración 2. Tamaños típicos de sensores digitales

Imagen obtenida de <https://quecamaradefotos.com>

Aparte del tamaño, también es importante la tecnología que utilizan los sensores. Principalmente hay cuatro tipos:

- CCD y SuperCCD
- CCD RGBE
- CMOS
- Foveon X3

Los dos tipos de sensores que más se utilizan en las cámaras actuales son los que utilizan tecnologías CCD y CMOS. Aunque los CCD fueron los primeros en utilizarse, actualmente la tecnología CMOS es más económica ofreciendo la misma calidad, por lo que son los más usados.

1.1 Motivación

La idea de este proyecto surge a partir del proyecto de investigación del profesor de la ETS. de Arquitectura de la Universitat Politècnica de València Pedro M. Cabezos Bernal, consistente en documentar y divulgar importantes obras de arte pictóricas del patrimonio valenciano mediante imágenes fotográficas de ultra alta resolución o resolución gigapíxel¹, en el cual me vi involucrado en algunos aspectos de asesoramiento técnico informático.

Muchos museos internacionales están optando por esta tecnología para mostrar al público general sus obras de arte sin necesidad de desplazarse hasta ellos y con un extraordinario nivel de detalle que no sería posible apreciar a simple vista. Algunos ejemplos son:

- Museo Nacional de Arte Reina Sofía², España.
- Museo Mural Diego Rivera³, México.
- Museo Mauritshuis⁴, Países Bajos.

Estos proyectos han contado con empresas especializadas que o bien han utilizado su propia solución hardware/software automatizada o han realizado las fotografías de forma manual. Lo que pretendemos en este proyecto es realizar un sistema robotizado propio de captura de bajo coste.

Algunas de las empresas especializadas en la captura de imágenes gigapíxel son Google, Haltadefinizione⁵ o la española Madpixel⁶.

¹ <https://gpix.upv.es/>

² <https://gigapixel.museoreinasofia.es/>

³ <http://museosmexico.cultura.gob.mx/gigapixel/museo-mural-diego-rivera/>

⁴ <https://www.mauritshuis.nl/en/>

⁵ <https://www.haltadefinizione.com/>

⁶ <https://www.madpixel.es/>

1.2 Objetivos

El objetivo de este proyecto es desarrollar un pequeño robot grúa, que pueda transportarse fácilmente, y pueda moverse de forma semiautomática controlado por una aplicación que podremos llevar en nuestro móvil y que nos permitirá automatizar en gran medida la toma de fotografías para la obtención final de imágenes gigapíxel.

Este prototipo de robot se encuentra gobernado por una placa Arduino Mega 2560 y junto con una aplicación de control, facilitará la tarea de crear imágenes de resolución gigapíxel de obras de arte.

Estas imágenes serán tomadas por el robot con una cámara digital réflex. Simplemente tendremos que situar al robot en la posición adecuada y dándole unos parámetros con la aplicación desarrollada para Android, este será capaz de realizar todas las fotografías de forma semiautomática.

Una vez procesadas las fotos, se crearán las imágenes de resolución gigapíxel que se podrán consultar a través de una página web, que nos permitirá desplazarnos por la imagen y acercarnos para lograr una visualización muy detallada de la obra.

1.3 Metodología

Cuando intentamos visualizar una imagen digital en pantalla y la ampliamos para ver sus detalles, estamos limitados por la resolución inicial con la que se tomó la imagen. El uso de imágenes gigapíxel, nos permite lidiar con este problema ya que hace posible ampliar imágenes hasta alcanzar un nivel muy superior de detalle en comparación con una imagen convencional. Para conseguir imágenes de resolución gigapíxel, algunos investigadores han intentado realizar cámaras que permitan obtener directamente una imagen gigapíxel, pero, dada la complejidad de este proceso, se han desarrollado escasos dispositivos capaces de obtener imágenes de resolución gigapíxel.

El físico jubilado Graham Flint, desarrolló a finales del año 2000 el proyecto Gigapxl (Flint, 2000), que en la actualidad se encuentra accesible pero desactivado. El objetivo de este proyecto era tomar fotografías de paisajes de gran formato y detalle de

ciudades, parques y monumentos de los EE. UU. y Canadá, para lo que desarrolló su propio prototipo de cámara fotográfica, que, combinada con equipos de software, escaneo e impresión digital, proporcionaba imágenes con una resolución de 4 gigapíxel. Esta cámara, utilizaba un formato de película especial de 450 mm por 225mm.

El programa AWARE⁷ iniciado en 2011 en la Universidad de Duke se centra en el diseño y fabricación de micro cámaras como base para cámaras de alta resolución. Así pues, cada micro cámara captura una parte de la escena, que son posteriormente ensambladas por el procesador encargado de realizar el proceso de stitching para crear la imagen final. Usando esta plataforma, el equipo de AWARE ha publicado diseños para cámaras con una resolución de 1 a 50 gigapíxel. De este programa liderado por David Brady, han surgido dos prototipos de cámaras:

- La cámara AWARE-2 (2012), desarrollada en colaboración con la Agencia de Proyectos de Investigación Avanzados para la Defensa (DARPA), es un cubo de 75x75x50cm formado por 98 micro cámaras de 14 megapíxeles con sensor monocromo (ver Ilustración 3).



Ilustración 3. Prototipo cámara AWARE-2

⁷ <http://disp.duke.edu/research/aware2-multiscale-gigapixel-camera>

- El prototipo más reciente de este programa es la cámara AWARE-10 (ver Ilustración 4), iniciada en 2012. Esta cámara está compuesta por 382 micro cámaras con una resolución combinada de casi 3 giga píxeles. Este prototipo se encuentra todavía en fase de desarrollo y por su elevada complejidad queda lejos de convertirse en una opción accesible (Brady, 2012).

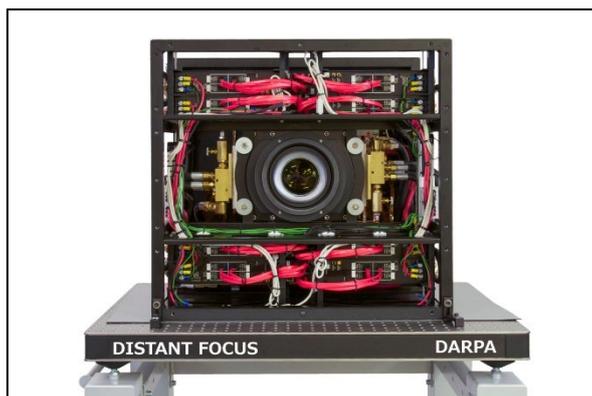


Ilustración 4. Prototipo cámara AWARE-10

En 2011 Google decidió llevar su tecnología desarrollada para Google Street View al interior de los edificios iniciando así el Google Art Project. Este proyecto permitió a los usuarios realizar recorridos virtuales por los principales museos, ver material en alta resolución relevante del mundo del arte y compartir imágenes y comentarios con amigos. En 2020, este proyecto fue renombrado como Google Arts & Culture⁸.

Para este proyecto, Google desarrolló su propia cámara de resolución gigapíxel que denominó Art Camera⁹. Esta cámara (ver Ilustración 5) es capaz de tomar las fotografías de las obras de arte de forma automatizada gracias a sus sensores de infrarrojos y de ultrasonidos. Una vez tomadas las fotografías, la propia cámara las une para crear la imagen final.

⁸ <https://artsandculture.google.com/>

⁹ <https://www.panoramaaudiovisual.com/fr/2016/06/02/la-nueva-google-art-camera-permite-capturar-imagenes-con-resolucion-de-gigapixel/>



Ilustración 5. Google Art Camera

Existen otras empresas especializadas en ese tipo de fotografía que también han desarrollado sus propias soluciones como son la empresa italiana Haltadefinizione o la española Madpixel.

Como hemos visto, el enfoque de una sola instantánea para la toma de fotografías de gigapíxeles tiene sus inconvenientes. El equipo es voluminoso, caro y complicado.

Otra opción más asequible, es la toma de multitud de imágenes de alta resolución en forma de mosaico que luego han de ser procesadas para poder visualizarlas conforme el usuario hace zoom de la imagen inicial. Una vez generada la imagen, nos encontramos con un archivo de tamaño del orden de Gigabytes, por lo que es necesario el tratamiento de este archivo para su correcta visualización.

Lo que se haría en esta fase es trocear la imagen gigapíxel en distintos niveles de profundidad o zoom y cada uno de estos niveles se dividirá a su vez en celdas. A mayor profundidad, mayor resolución de las imágenes. De esta forma, solo cargaremos en memoria y visualizaremos la celda correspondiente en lugar de la imagen gigapíxel completa. La difusión de las imágenes gigapíxel se realiza comúnmente a través de internet, para lo que existen distintos programas que facilitan esta tarea, tanto de pago, como gratuitos, como son OpenSeaDragon¹⁰ y Zoomify¹¹.

¹⁰ <https://openseadragon.github.io/>

¹¹ <http://www.zoomify.com/>

En este proyecto nos hemos decantado por crear un mosaico de imágenes que luego conformaran la imagen gigapíxel definitiva, y para ello se ha desarrollado una solución propia gracias a plataformas de hardware/software libre como Arduino.

Como podemos ver en Cabezos et al. (2022), se propone una nueva técnica de captura para imágenes gigapíxel que consiste en realizar las tomas moviendo la cámara paralela al lienzo. De esta forma, la nitidez de las imágenes será siempre óptima. Sin embargo, con el uso de este método se plantean algunos problemas:

- El punto de vista varía continuamente, por lo que la luz reflejada por la obra cambia entre las tomas, provocando ligeras diferencias de exposición y problemas con los reflejos especulares. Estos problemas se pueden resolver utilizando una fuente de luz controlada estratégicamente ubicada, que se mueve junto con la cámara.
- Cuando se va a componer el mosaico de fotos obtenido, la mayoría de los programas de stitching actuales no son capaces de unir el mosaico cuando el punto de vista no es único. Afortunadamente, este problema se puede solucionar con la ayuda de algunos algoritmos, inicialmente desarrollados por el profesor alemán Helmut Dersch¹², que fueron implementados en Hugin¹³, un software de stitching de código abierto con licencia GPL. Una opción comercial sería la aplicación PTGUI¹⁴. El software de fotogrametría SfM (Structure from Motion), también es capaz de componer mosaicos tomados con esta técnica, ya que su algoritmo de construcción es capaz de generar una malla texturizada a partir de fotografías tomadas desde diferentes puntos de vista. (Cabezos et al., 2022)

Teniendo en cuenta las consideraciones anteriores y como se comenta en Cabezos et al. (2022), disponemos de las siguientes técnicas de captura de imágenes gigapíxel para pinturas o superficies planas:

¹² <https://www.panotools.org/dersch>

¹³ <https://hugin.sourceforge.net>

¹⁴ <https://www.ptgui.com/>

- La técnica de captura de punto de vista único (ver Ilustración 6), que consiste en utilizar la técnica clásica de toma con cabezal panorámico para rotar la cámara alrededor de su centro de perspectiva, que quedaría fijo y centrado frente al lienzo, obteniendo así un conjunto de imágenes superpuestas.

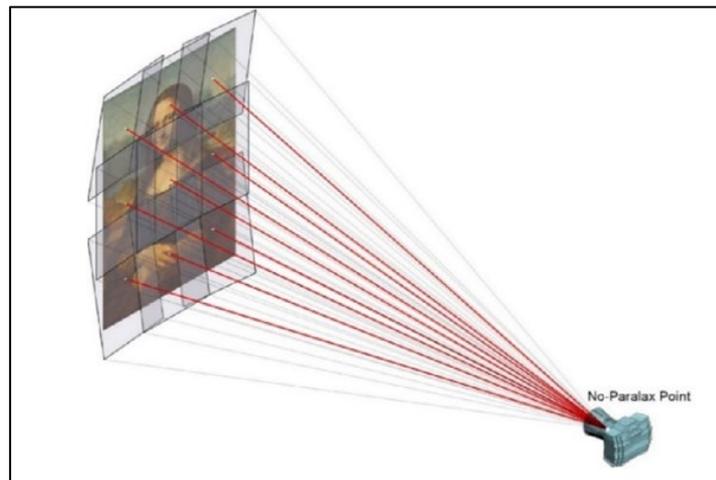


Ilustración 6. Captura de punto de vista único

Imagen obtenida de Cabezos et al. (2022)

- La técnica de captura de punto de vista múltiple paralela (ver Ilustración 7), que consiste en tomar un mosaico de imágenes superpuestas mientras la cámara se mueve paralelamente al lienzo describiendo varias filas o columnas. El sensor digital de la cámara debe estar paralelo a la obra.

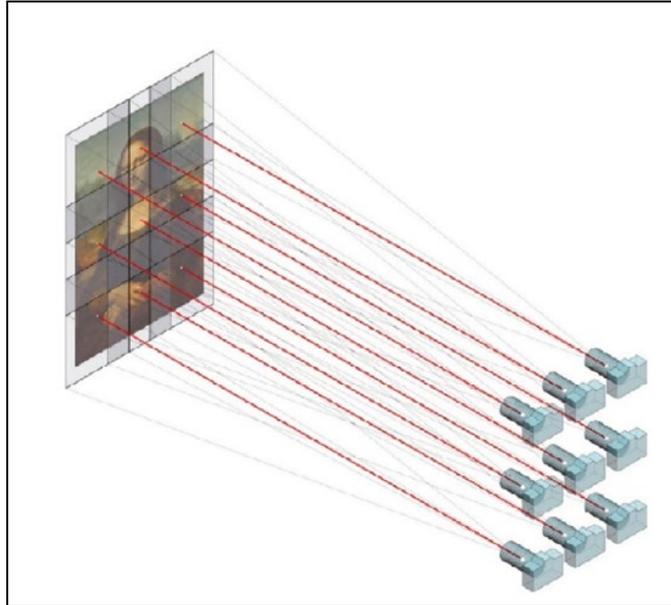


Ilustración 7. Captura de punto de vista múltiple paralela.

Imagen obtenida de Cabezos et al. (2022)

- La técnica de captura de punto de vista múltiple oblicua (ver Ilustración 8) es una variación de la anterior, en la que la cámara se puede inclinar, si es necesario.

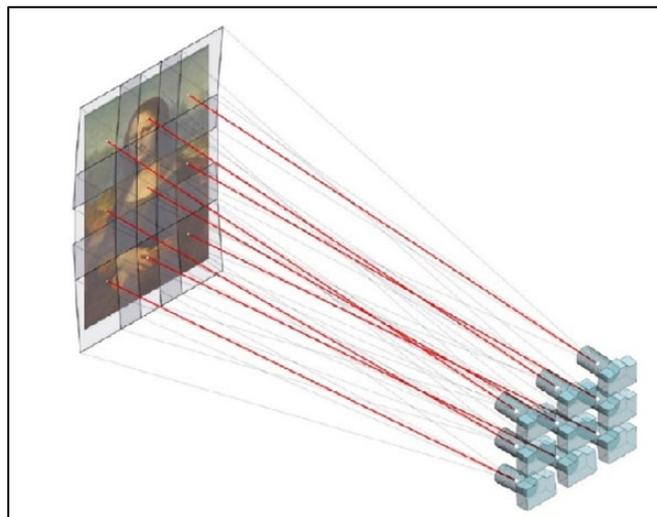


Ilustración 8. Captura de punto de vista múltiple oblicua

Imagen obtenida de Cabezos et al. (2022)

Para el prototipo a desarrollar en este proyecto se ha utilizado una cámara digital Canon EOS 90D¹⁵ de 32,5 MP equipada con un teleobjetivo Canon EF 200mm f2.8 L II USM y un cabezal panorámico Manfrotto 303 SPH junto con el método de captura de punto de vista múltiple paralelo.



Especificaciones más destacadas:

Sensor	CMOS APS-C de 32,5 megapíxeles efectivos
Procesador de imagen:	DIGIC 8
Distancia focal Objetivo	Equivalente a 1,6x la longitud focal del objetivo
Estabilizador de la Imagen	Estabilizador óptico de imagen en los objetivos compatibles Vídeo: IS Digital de vídeo
Sensibilidad	100 a 25.600 ISO (puede ampliarse a 51.200)
Enfoque	Mediante visor óptico: TTL-CT-SIR con un sensor CMOS Mediante visión en directo en pantalla LCD:

¹⁵ <https://www.canon.es/cameras/eos-90d/specifications/>

	Sistema AF Dual Pixel CMOS. Píxeles de detección de fase incorporados en el sensor de imagen 1
Sistema AF	TTL-CT-SIR con un sensor CMOS (visor óptico)/ sistema AF Dual Pixel CMOS. 45 puntos AF
Formatos de archivo foto/ vídeo	JPEG (compatible con Exif 2.31) RAW: RAW (CR3 14 bits), CRAW (RAW compacto) DPOF (Formato de orden de impresión digital) compatible con la versión 1.1
Grabación simultánea de RAW+JPEG.	Es posible cualquier combinación de RAW + JPEG
Tamaño de imagen	RAW: (3:2) 6960 x 4640, (4:3) 6160 x 4640, (16:9) 6960 x 3904, (1:1) 4640 x 4640 JPEG 3:2 (L) 6960 x 4640, (M) 4800 x 3200, (S1) 3472 x 2320, (S2) 2400 x 1600 JPEG 4:3 (L) 6160 x 4640, (M) 4256 x 3200 (S1) 3072 x 2320 (S2) 2112 x 1600 JPEG 16:9 (L) 6960 x 3904, (M) 4800 x 2688 (S1) 3472 x 1952 (S2) 2400 x 1344 JPEG (1:1) (L) 4640 x 4640, (M) 3200 x 3200, (S1) 2300 x 2300 (S2) 1600 x 1600
Almacenamiento	SD, SDHC o SDXC (compatibles con UHS-II)
Flash	flash preparado, sincronización de alta velocidad, bloqueo FE, compensación de la exposición con flash, luz de reducción de ojos rojos
Pantalla Táctil	TFT Clear View II de 7,7 cm (3,0") con formato 3:2 y ángulo variable, aprox. 1 040 000 píxeles
Peso (sólo el cuerpo)	Aprox. 700 g (batería incluida)
Interfaces	USB Micro-B

		Wi-Fi (IEEE802.11b/g/n) (solo 2,4 GHz) Bluetooth v 4.1 Salida mini HDMI
Sistema operativo compatible		Windows 10 / 8.1 / 7 SP1 10 Mac OS X 10.14 / 10.13 / 10.12 Aplicación Camera Connect para iOS y Android
Interruptor/mando distancia	a	Disparador remoto RS-60E3, mando a distancia RC-6 o BR-E1, mando a distancia con temporizador TC-80N3.

Teniendo en cuenta el método de captura, los pasos a seguir para conseguir una imagen gigapíxel serían los siguientes:

- Primero se divide la imagen original en pequeñas porciones y se fotografía cada una de estas porciones en alta resolución.
- A continuación, mediante el software de stitching, se procesan y se unen estas porciones digitalizadas para conformar la imagen digital definitiva.

El proceso de captura de las imágenes puede ser un tanto complejo ya que hay una serie de parámetros que se deben calcular previamente al disparo de la cámara para que el software controlador del disparo haga un barrido por la escena configurada deteniéndose el tiempo necesario para fotografiar cada porción. La variación de estos parámetros y el número de disparos depende de la distancia a la escena fotografiada, la distancia focal del objetivo empleado en la cámara, el valor de la superposición entre imágenes (necesaria para hacer el fusonado) y por supuesto las dimensiones físicas de la escena que se desea retratar.

Según Cabezos et al. (2021) los parámetros que se necesitan para calcular el tamaño del mosaico de imágenes con una superposición mínima del 30% entre capturas adyacentes para asegurarnos de que el proceso de stitching se realiza de forma correcta, serían los siguientes:

El desplazamiento vertical de la cámara, que se calcula con la fórmula:

$$C_v = \frac{D \cdot S_w \cdot (100 - P)}{100 \cdot DF}$$

Donde C_v , es el desplazamiento vertical de la cámara (mm).

D , es la distancia entre la cámara y el lienzo(mm).

S_w , es la anchura del sensor (mm).

P , es el solape entre imágenes (%).

DF , es la distancia focal del objetivo (mm).

El desplazamiento horizontal de la cámara que se calcula como una fracción del desplazamiento vertical cuando se conoce la relación de aspecto del sensor con la siguiente formula:

$$C_h = \frac{C_v}{S_a}$$

Donde C_h , es el desplazamiento horizontal de la cámara (mm).

C_v , es el desplazamiento vertical de la cámara (mm).

S_a , es la relación de aspecto del sensor.

Con estos datos, finalmente se puede componer el mosaico de fotografías a tomar (filas x columnas) con las siguientes fracciones:

$$F = \frac{H}{C_v} \quad C = \frac{A}{C_h}$$

Donde F, es el número de filas del mosaico de imágenes.
 C, es el número de columnas del mosaico de imágenes.
 H, es la altura de la pintura(mm).
 A, es la anchura de la pintura(mm).
 C_h, es el desplazamiento horizontal de la cámara (mm).
 C_v, es el desplazamiento vertical de la cámara (mm).

Para garantizar una alta calidad y detalle de la imagen gigapíxel resultante, debe tenerse en cuenta la resolución relativa o densidad de píxeles. El valor máximo de la densidad de píxel depende de varios parámetros que se relacionan con la siguiente formula:

$$DP = \frac{S_r \cdot DF \cdot U_f}{D \cdot S_w}$$

Donde DP, es la densidad de píxeles (ppc o ppp).
 S_r, es la resolución horizontal del sensor (px).
 DF, es la distancia focal del objetivo (mm).
 U_f, es el factor de conversión de unidades.
 D, es la distancia de la cámara al lienzo (mm).
 S_w, es la anchura del sensor (mm).

Para obtener la mayor densidad de píxeles posible, es necesario utilizar teleobjetivos y acercar la cámara a la pintura el máximo posible respetando siempre la distancia mínima de enfoque del objetivo.

Hay que tener en cuenta que una densidad de píxeles alta no asegura por sí misma una imagen de alta calidad si no se utiliza un objetivo que ofrezca la mejor calidad óptica posible y mantenemos siempre la cámara estable al tomar la fotografía. Enfocar y disparar la cámara remotamente ayuda a evitar pequeños movimientos que producirían imágenes ligeramente borrosas, ya que cualquier pequeño movimiento será amplificado por el sensor de la cámara.

Es posible estimar la resolución de la imagen gigapíxel resultante a partir de la siguiente ecuación:

$$RS = \frac{H \cdot A \cdot DP^2}{1 \cdot 10^9}$$

Donde RS, es la resolución final de la imagen gigapíxel.

H, es la altura de la pintura(cm).

A, es la anchura de la pintura(cm).

DP, es la densidad de píxeles (ppp).

El valor ISO de la cámara debe ser lo más bajo posible para evitar ruido en las imágenes. Además, las imágenes deben almacenarse en formato RAW en lugar de JPG.

El proceso de toma de fotografías puede extenderse a lo largo de varias horas por lo que hay que prestar también atención a otras condiciones que pueden variar con el paso del tiempo tales como la posibilidad de movimiento de objetos en la trayectoria del disparo o las condiciones climáticas y de iluminación que deben ser lo más estables posibles para que el aspecto de las distintas capturas sea el del mismo instante temporal.

Una vez terminado el proceso de captura, se debe hacer un trabajo de procesado de las imágenes individuales para fusionarlas en una única fotografía. El proceso de fusión de las imágenes (*stitching*) consiste en la unión de dos o más imágenes cuyos bordes se solapan, es decir, que tienen puntos gráficos coincidentes. Como se ha mencionado anteriormente, existen aplicaciones de pago y gratuitas que permiten realizar el proceso de *stitching*, todas con un flujo de trabajo similar:

- **Fase de registro.** El programa carga las imágenes y lee los datos EXIF incrustados en las imágenes. Acto seguido, se buscan las características coincidentes entre el conjunto de imágenes en base a diferentes métodos de alineación.
- **Fase de calibración.** Se intenta minimizar las diferencias propias de los factores de la toma de imágenes como la distorsión de la lente, el valor de exposición de la imagen, aberraciones cromáticas, etc.
- **Fase de fusionado.** Se ejecutan los ajustes descubiertos en la fase de calibración y se procede al fusionado de la imagen final.

Es aconsejable guardar el archivo resultante en formato PSB (Adobe Photoshop Big File), desarrollado por Adobe para su paquete de software Adobe Photoshop. El formato de archivo PSB permite almacenar archivos de tamaño superior a 2GB, admite documentos de hasta 300.000 píxeles en cualquier dimensión (9000 Megapíxeles como máximo), así como todas las características de Photoshop, como las capas, efectos y filtros.

1.4 Estructura de la memoria

Esta memoria comienza con una Introducción al uso de las imágenes gigapíxel en el ámbito de las obras de arte, y la tecnología utilizada actualmente en las cámaras digitales. Seguidamente, en el subapartado Motivación, se comenta que este proyecto surge a partir de un proyecto de investigación del profesor Pedro M. Cabezos. El subapartado Objetivos, expone lo que pretendemos conseguir con este proyecto. Para finalizar el apartado de Introducción, se comenta la Metodología que se va a utilizar para la generación de las imágenes gigapíxel, detallando además las distintas opciones disponibles, que están recogidas en Cabezos et al. (2022).

En el segundo apartado, Estado del Arte, se presentan distintas empresas que ofrecen servicios de captura de imágenes gigapíxel en el ámbito de la cultura, así como distintos campos de investigación en los que se utilizan este tipo de imágenes.

En el tercer apartado, se detalla el Desarrollo del Proyecto en sí, donde se describe la plataforma robotizada desarrollada por Pedro M. Cabezos, así como el diagrama eléctrico de la misma, los módulos hardware que utiliza y la programación de la aplicación de control Android y de la placa Arduino.

En el cuarto apartado, Experimentación y pruebas, se analizan los problemas surgidos en el desarrollo del prototipo y se comentan las soluciones que se implementaron.

En el quinto apartado, Conclusiones y Trabajo Futuro, se analizan las posibles mejoras de cara a futuras versiones.

Finalmente, en el sexto y último apartado, Bibliografía, se listan las referencias que se han utilizado como consulta en el desarrollo de la memoria.

2. ESTADO DEL ARTE

Como se ha comentado anteriormente, dada la complejidad del desarrollo de cámaras capaces de tomar fotografías de resolución gigapíxel, no hay disponible a la venta para el usuario final cámaras gigapíxel y son escasas las empresas se dedican a ofrecer los servicios de captura de imágenes en esta resolución.

Google ha fabricado una decena de sus Art Camera, pero no están a la venta, simplemente se ceden de forma gratuita cuando alguien desea iniciar un proyecto de captura de imágenes gigapíxel en colaboración con Google.

Al igual que Google, tanto la empresa española Madpixel, como la italiana Haltadefinizione, han desarrollado su propia solución propietaria y ofrecen sus servicios profesionales de pago a cualquier institución o persona que está interesada.

En el ámbito de las obras de arte, las formas de abarcar la captura de imágenes gigapíxel son similares, por lo que vamos a tomar como ejemplo el caso de uso de esta tecnología en el Museo Reina Sofía que, con el apoyo de Telefónica, ha abierto en su web el espacio Gigapixel¹⁶, donde es posible examinar obras de su colección on-line con un nivel de precisión que va más allá de lo que el ojo humano puede apreciar.

Esto ha sido posible gracias a un sistema robotizado de ejes cartesianos desarrollado a medida para el Museo Reina Sofía de 9 m. de largo y 3,5 m. de ancho que se desplaza con una precisión de 25 micras.

Como se comenta en la nota de prensa del propio proyecto (Museo Reina Sofía, 2020), se ha llevado más allá la fotografía gigapíxel ya que al utilizar distintos filtros de luz (visible, ultravioleta, infrarroja e imagen radiográfica), es posible observar detalles (ver Ilustración 9) que no se pueden ver a simple vista como pueden ser restauraciones, cambios durante la creación de la obra, etc.

Además, la web creada para la consulta de las obras permite comparar de manera simultánea dos tomas del mismo lienzo con distintos filtros.

¹⁶ <https://gigapixel.museoreinasofia.es/>

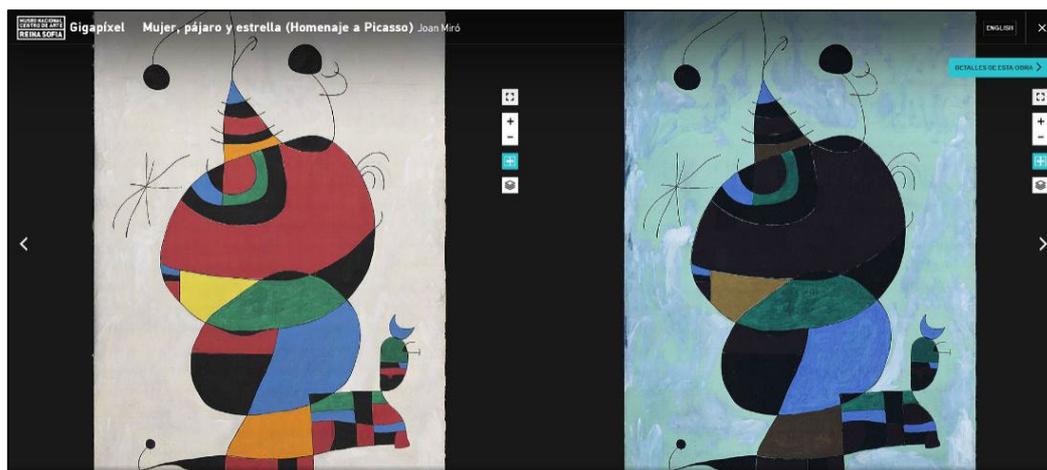


Ilustración 9. Captura de la web del Museo Reina Sofía

El proyecto Gigapixel del Museo Reina Sofía muestra el potencial de las imágenes gigapixel aplicadas a las obras de arte, ya que permite estudiar las obras sin descargarlas así como divulgarlas de forma que cualquier persona con un dispositivo con conexión a internet las tenga al alcance de la mano.

Existen otros casos de uso de la tecnología gigapixel que no están relacionados con el que se ocupa en este proyecto, como sería en el campo de la medicina o su uso en imágenes panorámicas.

La compañía Active Computer Services¹⁷, ofrece como servicio la creación de imágenes gigapixel de 360 grados de objetos o de panorámicas para tours virtuales. El uso de la tecnología gigapixel resulta interesante para mostrar ubicaciones de interés turístico bien sea en el exterior o en el interior de edificios dado el alto grado de detalle de las imágenes.

Otra empresa especializada en la captura de imágenes de resolución gigapixel de objetos o animales de tamaño tanto macroscópico como microscópico es GigamacroTM ¹⁸. Para la captura de imágenes han desarrollado su propio sistema robotizado denominado GIGAmacro Magnify2 (ver Ilustración 10). Esta solución incluye el sistema robotizado, las ópticas, la cámara, el software y la creación y visualización de las imágenes gigapixel.

¹⁷ <http://www.gigapixel.com/>

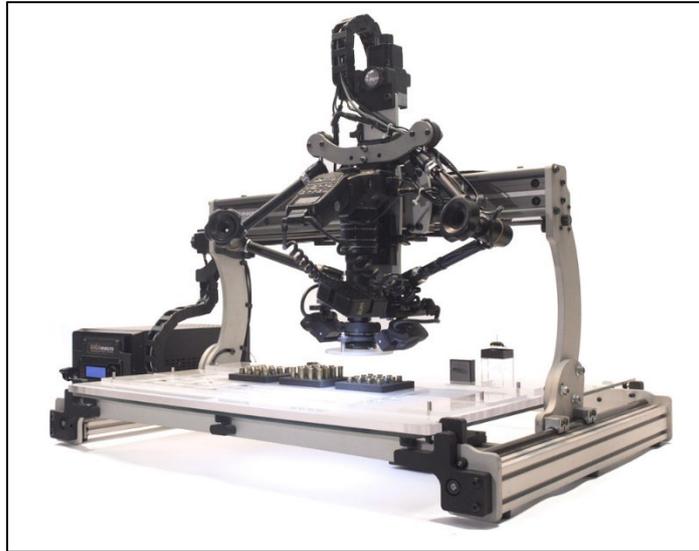


Ilustración 10. Robot GIGAmacro Magnify2

En el campo de la medicina, existen diversos artículos de investigación que utilizan imágenes gigapíxel para ayudar en la detección de cáncer. Las imágenes de alta resolución muestran todo su potencial en este ámbito, como podemos ver en (Liu et al. 2017), donde se propone un método para detectar metástasis de cáncer de mama. En este estudio se aprovecha la alta calidad y resolución de las imágenes gigapíxel combinándolas con Redes Neuronales y Deep Learning, lo que les permite detectar un 92,4% de los tumores en comparación con el 82,7% del mejor enfoque automatizado anterior o con el 73.2% de un patólogo humano entrenado.

Otro campo de investigación relacionada con las imágenes gigapíxel lo podemos encontrar en el repositorio de internet PANDA Dataset¹⁹. Este repositorio online, contiene multitud de imágenes gigapíxel destinadas al análisis y reconocimiento de objetos, centrándose en concreto en el análisis y seguimiento de seres humanos. Realmente este enfoque resulta aterrador desde el punto de vista de la privacidad de las personas, sobre todo después de leer el artículo de investigación que han publicado (Wang et al. 2020). En el apartado 4.2 del artículo, se centran en el seguimiento de peatones durante un largo periodo de tiempo incluso en escenarios muy poblados de

¹⁹ <http://www.panda-dataset.com/>

personas. Al leer este artículo ha venido a mi memoria la película estadounidense de ciencia ficción *Minority Report* dirigida por *Steven Spielberg* en 2002. En esta película cuentan con un sistema de identificación mediante cámaras y drones que permite identificar a los individuos que deambulan por la ciudad pudiendo localizar a cualquier ciudadano en todo momento.

3. DESARROLLO DEL PROYECTO

En este proyecto nos hemos querido sumar a las iniciativas existentes poniendo en práctica una serie de técnicas fotográficas y de equipamiento, desarrollado por un equipo de investigadores de la UPV, para poder alcanzar un altísimo grado de detalle y de fidelidad, tanto métrica como cromática, en la documentación de las obras pictóricas, de modo que los resultados puedan cumplir una doble función; la divulgativa, orientada al público en general, y la documental, dedicada a los conservadores y estudiosos del arte.

3.1 La plataforma Robotizada

La plataforma ha sido diseñada y construida por el profesor Pedro Cabezos Bernal y consiste en una base rectangular metálica que incorpora una tijera vertical capaz de plegarse/desplegarse hasta alcanzar aproximadamente 3 metros de altura. La idea es crear un robot de pequeño tamaño que se pueda transportar de forma sencilla y que no requiera de ningún montaje adicional. En la Ilustración 11 puede observarse las dimensiones de la plataforma tanto cuando esta plegada como cuando está extendida.

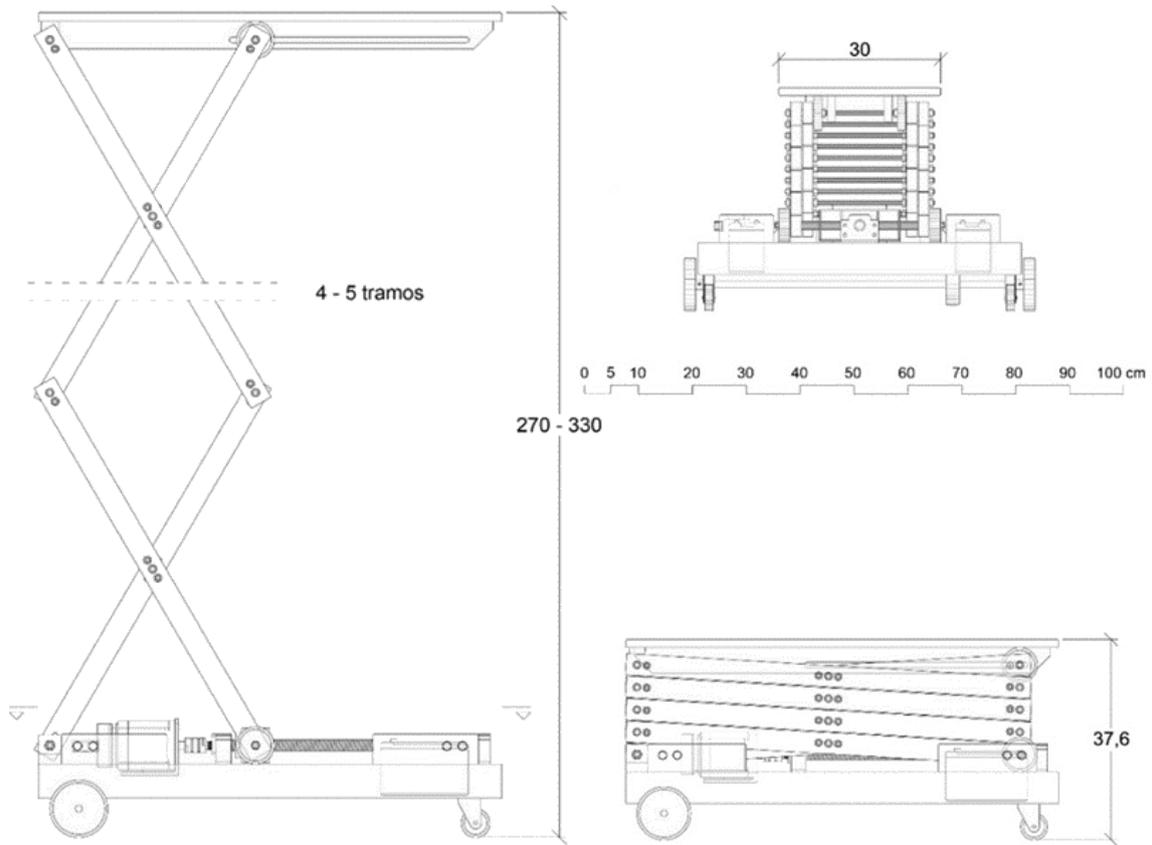


Ilustración 11. Plataforma Robot y sus dimensiones

La base de la plataforma tiene forma rectangular con unas medidas de 80x60 cm y está realizada en acero laminado de 2mm de espesor. La tijera está compuesta por 5 tramos más una plataforma donde descansara la cámara.

El resto de las piezas (ver Ilustración 12) han sido diseñadas e impresas en una impresora 3D. Entre estas piezas podemos encontrar pasa cables para la parte inferior de la plataforma, las cajas que contienen las baterías y los componentes electrónicos y los engranajes de los motores.

Para alimentar todos los componentes electrónicos que gobiernan la plataforma, se incluyen dos baterías AGM (*Absortion Glass Mat*) de 12 voltios y 9 amperios, que se conectan en serie para obtener una corriente de 24 voltios. El interior de estas baterías contiene láminas de fibra de vidrio que absorben el ácido formando una especie de gel. Esto les confiere mayor seguridad ya que, entre otras cosas, no tienen fugas de líquido, no desprenden gases ni precisan mantenimiento.

La plataforma tiene un interruptor de parada de emergencia tipo seta y cuenta con finales de carrera que detienen la plataforma antes de que esta llegue a la posición de plegado, en la que las barras podrían resultar peligrosas si alguien interpusiera alguna parte del cuerpo entre ellas.



Ilustración 12. Cajas impresas en una impresora 3D

Además de la programación del emisor y receptor, el trabajo incluye todo el conexionado y cableado del hardware integrante de la plataforma y la calibración y lectura de los sensores ultrasónicos de medición que permiten situar la plataforma automatizada a la distancia óptima de captura siguiendo el esquema de conexionado de la Ilustración 13.

Los motores se alimentan directamente de la tensión que proporcionan las baterías a través del driver, la placa Arduino se alimenta a 7V con un regulador de voltaje, y el resto de la electrónica se alimenta a 5V procedentes del Arduino.

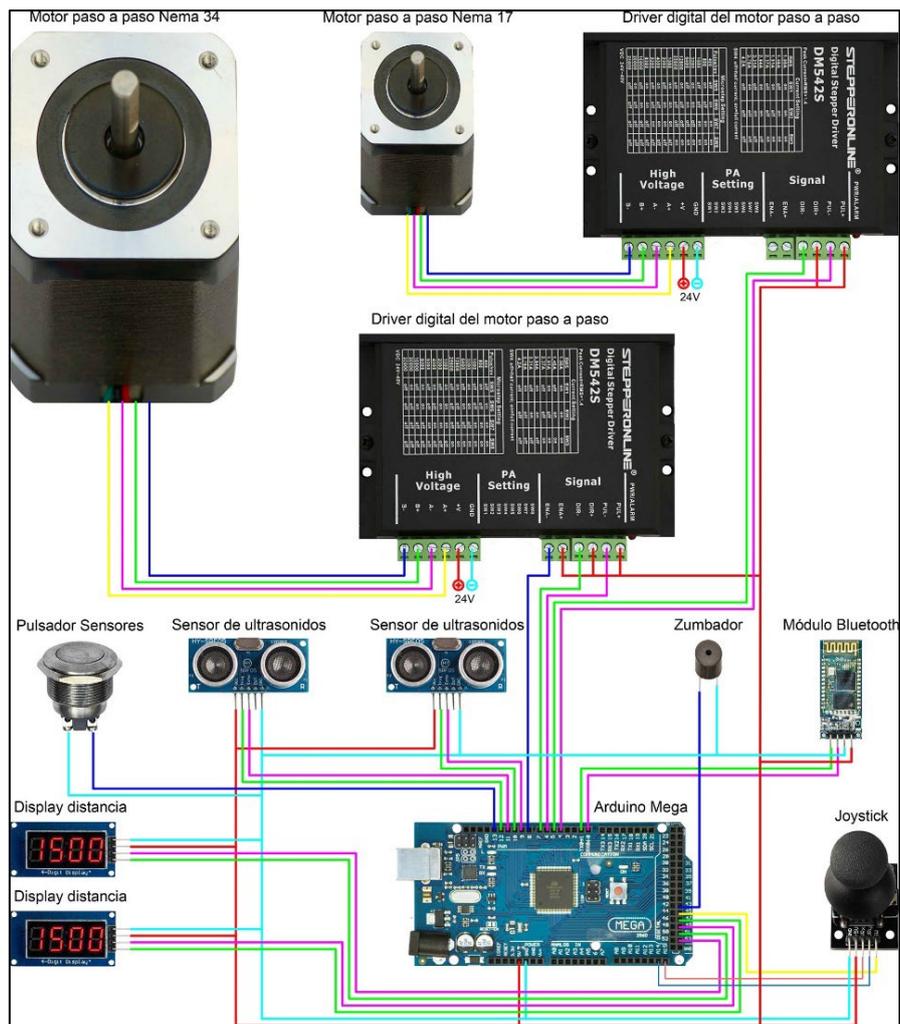


Ilustración 13. Esquema eléctrico del robot

El robot dispone de dos motores paso a paso NEMA (National Electrical Manufacturers Association), un NEMA34 que se encarga del movimiento vertical y un NEMA17 que se encarga del movimiento horizontal. Los motores paso a paso ofrecen una alta fiabilidad y precisión, por lo que se utilizan en impresoras, CNCs, robots, etc.

El motor NEMA17 es un motor bipolar con un ángulo de paso de 1.8 grados (200 pasos por revolución) y que puede desarrollar un par de 3.2kg/cm. El motor NEMA34 también es un motor bipolar con un ángulo de paso de 1.8 grados capaz de desarrollar un par de 42kg/cm, y dispone además de un freno eléctrico para que la tijera quede bloqueada en caso de un fallo en el suministro eléctrico.

Para gobernar estos motores, utilizaremos un driver DM542s para cada uno, este driver nos permite alimentar el motor de forma adecuada (el voltaje que ofrece el Arduino no es suficiente) y controlar el motor de forma más sencilla.

El DMS542s es un driver para motores paso a paso bipolares que implementa la técnica Microstepping (ver Ilustración 14), con la que es posible conseguir que el giro del motor se desarrolle en pasos más cortos, lo que nos permite que el motor gire más fluidamente, con más precisión y menor vibración y ruido.

Tanto la resolución del Microstepping como la alimentación de salida que proporciona el DMS542s es programable por medio de los switches que lleva integrados. Además, el algoritmo de control que implementa nos proporciona un par óptimo del motor en todo momento.

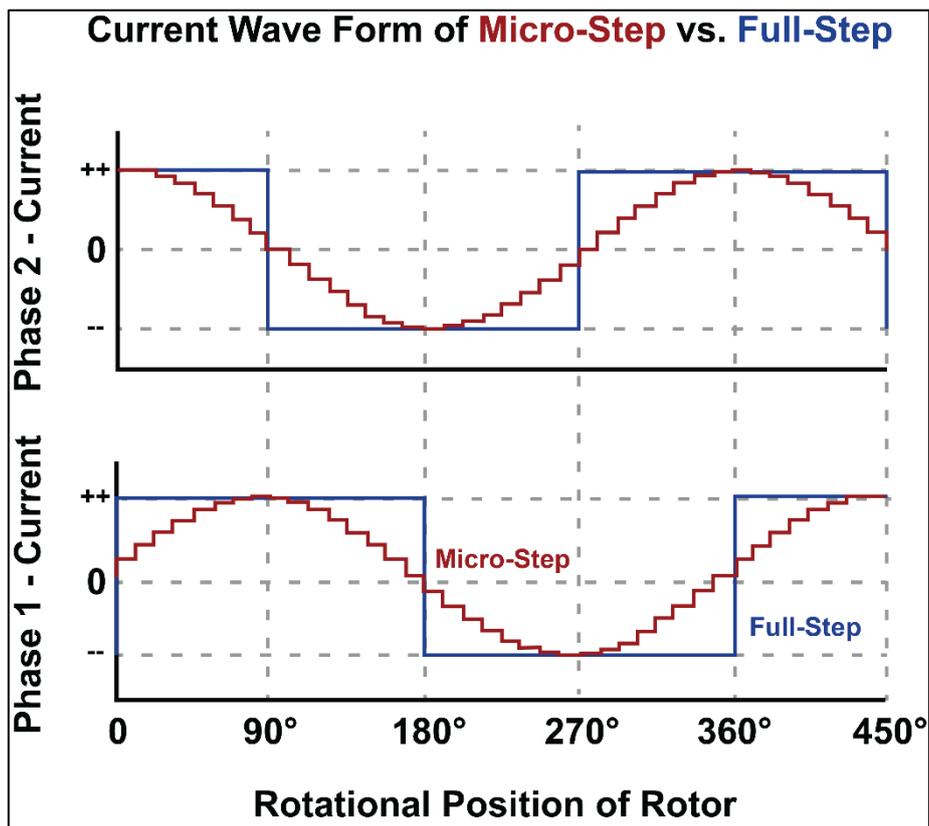


Ilustración 14. Microstepping

Imagen obtenida de <https://www.drdflo.com/>

3.2 El Módulo Receptor Arduino

El módulo receptor está implementado en una placa Arduino Mega. Este módulo atenderá las órdenes enviadas por el módulo de control (Android) y accionará los motores paso a paso de la plataforma, lo que permitirá el ascenso o descenso de la tijera y el avance o retroceso en sentido horizontal. La comunicación entre estos dos módulos se realizará de forma inalámbrica a través de una conexión bluetooth.

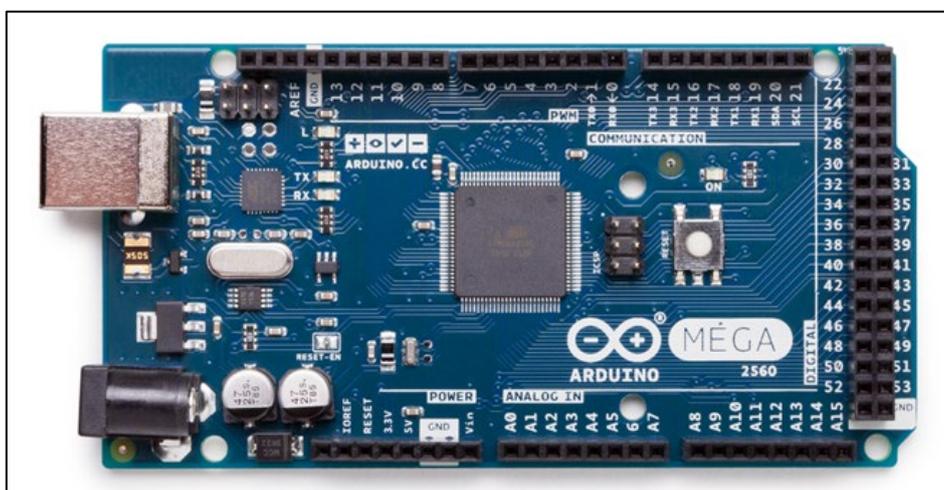


Ilustración 15. Arduino Mega 2560

El Mega 2560 es una placa basada en el microcontrolador ATmega2560²⁰. Dispone de 54 pines digitales de entrada/salida, 16 entradas analógicas, 4 puertos serie, un oscilador de 16 MHz, una conexión USB, un conector de alimentación y una memoria flash de 256K.

Esta placa es la encargada de gobernar y gestionar todos los dispositivos de la plataforma. Dispone de 2 modos de funcionamiento, uno manual que se activa al pulsar el joystick y otro a distancia que funciona desde un dispositivo Android. El modo manual se activa al utilizar el pulsador del joystick y se mantiene hasta que se vuelve a pulsar.

²⁰ <https://www.arduino.cc/en/Guide/ArduinoMega2560>

En este modo la placa Arduino solo responde a las indicaciones que se le dan a través del joystick para desplazar el robot en horizontal o desplegar/plegar la tijera.

En el modo normal, la placa Arduino está pendiente de recibir órdenes vía Bluetooth, una vez recibida la orden, la ejecutará de inmediato según corresponda. Como método de seguridad se ha incluido un botón rojo de emergencia tipo seta que si se pulsa corta la alimentación de todo el sistema, bloqueando además el motor que acciona la tijera.

```
leer_entradas;
si se ha pulsado botón medir
    activar ultrasonidos
    mostrar resultado en pantalla
si se ha pulsado el joystick
    si (control_manual = true)
        control_manual = false
    sino
        control_manual = true
    fsi
fsi
si (control_manual=true)
    pasar a control manual
sino
    leer bluetooth
    si se ha recibido datos
        decodificar comando
        ejecutar comando
    fsi
fin
```

Ilustración 16. Seudocódigo aplicación Arduino

Para el uso de los motores se ha utilizado la librería AccelStepper²¹ que permite gestionar más de un motor y además dispone de un movimiento acelerado lo cual hace que los motores se muevan de forma más suave.

Para la programación de la placa Arduino se ha utilizado el IDE²² gratuito de Arduino v 1.8.19. Han resultado de gran utilidad los ejemplos incluidos²³ en el IDE y en la documentación de Arduino, así como otros tutoriales de internet como por ejemplo el blog de Luis Llamas²⁴. En la Ilustración 16, se puede ver el pseudocódigo del bucle principal de la aplicación para la placa Arduino.

3.3 El Módulo Bluetooth

Los dispositivos bluetooth pueden conectarse como maestro o como esclavo. Cuando un dispositivo se configura como maestro, será el encargado de iniciar el proceso de conexión con otro dispositivo que estará configurado como esclavo. Durante este proceso, el dispositivo esclavo solicitará un código de emparejamiento al dispositivo maestro.

El HC-06 (ver Ilustración 17), es un pequeño módulo (1.7 cm x 4 cm aprox.) de bajo coste que soporta únicamente conexiones en modo esclavo y es compatible con la mayoría de los dispositivos con soporte bluetooth actuales.

Cumple el estándar bluetooth v2.0+EDR, con un rango de alcance de 5m a 10m. Al añadir este módulo bluetooth a una placa Arduino, aumentamos exponencialmente las posibilidades de comunicación e interacción con un ser humano a través de un móvil, Tablet, etc.

²¹ <https://www.airspayce.com/mikem/arduino/AccelStepper/index.html>

²² <https://www.arduino.cc/en/software>

²³ <https://docs.arduino.cc/built-in-examples/>

²⁴ <https://www.luisllamas.es/>

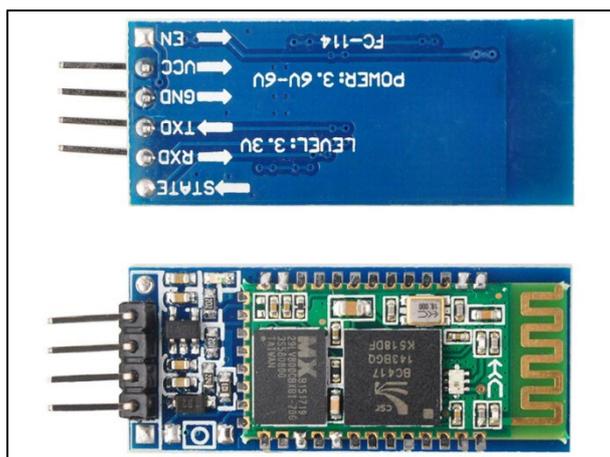


Ilustración 17. Módulo Bluetooth HC-06

El módulo HC-06 dispone de 4 pines de conexión y de un led que nos indica el estado de conexión del dispositivo, si parpadea significa que no está vinculado con ningún dispositivo, cuando permanece fijo, indica que está conectado a un dispositivo bluetooth.

Estos 4 pines son:

- VCC: Es el pin que proporciona la alimentación al dispositivo. Admite 5 o 3.3V
- GND: Es el pin que va a tierra o GND del Arduino
- TX: Es el pin de transmisión de datos. Por este pin se envía al Arduino los datos recibidos del dispositivo al que se está conectado. Se conecta al pin RX del Arduino.
- RX: Es el pin de recepción de datos. Por este pin el dispositivo recibe del Arduino los datos que se desea enviar al dispositivo vinculado. Se conecta al pin TX del Arduino.

3.4 El Sensor Ultrasónico

Los sensores ultrasónicos miden la distancia a un objeto contando el tiempo entre la emisión y la recepción de una onda ultrasónica. El sensor ultrasónico que hemos elegido (ver Ilustración 18) es el HY-SRF05, que utiliza un cabezal para emitir la señal

y otro para recibirla y es capaz de detectar objetos entre 1.7 - 400 cm con una precisión de 2mm.

Básicamente su modo de funcionamiento es el siguiente: el emisor lanza una señal ultrasónica que poco después (mínimo después de 10µs) es recibida por el receptor después de que la señal rebote en algún objeto. En función del tiempo que tarda en volver la señal ultrasónica obtendremos la distancia al objeto. No es posible realizar otra medición antes de 20 ms.

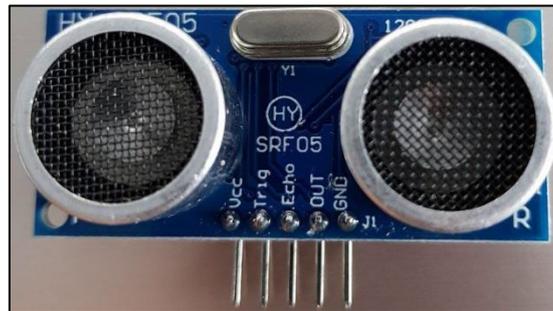


Ilustración 18. Sensor ultrasónico HY-SRF05

La distancia se puede calcular con la siguiente fórmula (el valor se multiplica por 1/2 ya que t es el tiempo recorrido de ida y vuelta).

$$D = \frac{1}{2} * t * C$$

donde D es la distancia,
 t es el tiempo entre la emisión y la recepción,
 C es la velocidad del sonido.

3.5 Módulo Android

Este módulo consiste en una aplicación de control desarrollada para dispositivos Android con soporte de bluetooth (móvil o *Tablet*), que permitirá gobernar de forma remota y establecer el movimiento programado de la plataforma. Esto permitirá el posicionado libre de la cámara que irá instalada sobre la bandeja superior de la plataforma, además del posicionamiento programado de la cámara para la captura automatizada de las distintas fotografías que compondrán el mosaico que sirve para componer la imagen gigapíxel. Esta programación del posicionamiento deberá calcularse en función de diversos parámetros, como la distancia focal, el tamaño de la obra y la distancia entre la plataforma y el cuadro.

3.5.1 Especificación de requisitos

En este apartado, vamos a definir los requisitos que debe cumplir nuestra aplicación de control de la plataforma robotizada. Como requisito general, se ha de tener en cuenta que es necesario tener emparejado el dispositivo móvil con el adaptador bluetooth de la placa Arduino que gobierna la plataforma.

El software a desarrollar debe ser aplicación que se ejecute en un dispositivo ligero y portátil que podamos llevar en la mano y que permita controlar la plataforma robotizada en modo manual o en modo programado semiautomático. Para la especificación de requisitos veremos los casos de uso posibles de la aplicación.

DESCRIPCIÓN DE LOS CASOS DE USO

Caso de Uso	Establecer parámetros
Actores	Usuario
Resumen	El caso de uso se inicia cuando el Usuario desea establecer los parámetros necesarios para operar la plataforma en modo programado. La aplicación debe proporcionar opciones para introducir y almacenar estos parámetros.

Caso de Uso	Mover la plataforma en modo manual
Actores	Usuario
Resumen	El caso de uso se inicia cuando el Usuario quiere mover la plataforma robotizada de modo libre, bien desplazándola en sentido horizontal o elevando/descendiendo el soporte de la cámara. La aplicación debe proporcionar la interfaz necesaria para poder dirigir la plataforma.

Caso de Uso	Mover la plataforma en modo programado
Actores	Usuario
Resumen	El caso de uso se inicia cuando el Usuario quiere mover la plataforma robotizada en modo programado semiautomático. La aplicación debe proporcionar la interfaz necesaria para que la plataforma se desplace de una posición a la siguiente.

La interfaz de usuario debe ser sencilla, en el caso de uso “Mover la plataforma en modo manual”, se mostrará una interfaz con una cruceta de 4 botones en forma de flecha indicando la dirección en la que queremos desplazar la plataforma.

Para el caso de uso “Mover la plataforma en modo semiautomático”, se mostrará una interfaz con 4 botones que permitirán comenzar a ejecutar la programación, avanzar o retroceder una posición en el mosaico de fotos o reenviar el último movimiento en caso de comprobar visualmente que no se ha ejecutado.

La interfaz del caso de uso “Establecer parámetros” permitirá introducir los parámetros necesarios para el funcionamiento en modo semiautomático.

3.5.2 Diseño de la aplicación

En este apartado veremos el prototipado de la aplicación utilizando la técnica de wireframing. Para realizar los prototipos (ver Ilustración 19, Ilustración 20 e Ilustración 21) se ha utilizado la herramienta online gratuita wireframe.cc <https://wireframe.cc/>



Ilustración 19. Boceto Actividad Modo Manual



Ilustración 20. Boceto Actividad Modo Programado



Ilustración 21. Boceto Actividad Ajustes

3.5.3 Implementación de la aplicación

Para llevar a cabo este proyecto, se ha hecho uso del entorno de desarrollo *Android Studio*²⁵. Este entorno de desarrollo integrado (*IDE*) ha sido desarrollado por Google, está basado en IntelliJ IDEA²⁶ y se encuentra disponible bajo licencia Apache 2.0. Dentro de este *IDE*, se integran otras herramientas como un sistema de compilación basado en *Gradle*, un emulador de dispositivos Android, integración con *GitHub*, compatibilidad integrada con *Google Cloud Platform*,

Android Studio, permite el desarrollo de aplicaciones en *JAVA*, *C++* y más recientemente en *Kotlin*²⁷. En este caso se ha utilizado *JAVA* para el desarrollo de la aplicación por ser un lenguaje conocido, aunque Google está apostando muy fuerte por *Kotlin*.

²⁵ Android Developers. [Consultado el 24 de mayo de 2022]. Disponible en: <https://developer.android.com/studio/intro>

²⁶ <https://www.jetbrains.com/idea/>

²⁷ <https://developer.android.com/kotlin/>

Después de realizar varias pruebas tratando de implementar el módulo encargado de la comunicación de la aplicación vía Bluetooth con el dispositivo Arduino se optó por utilizar una librería desarrollada por *Prasad Parshram*²⁸ bajo licencia MIT © 2021. Esta librería está desarrollada en *JAVA* y requiere el uso del SDK de Android v4.4 o superior, y nos provee de una clase que permite que nuestro dispositivo funcione tanto en modo cliente como en modo servidor de comunicaciones. Esta clase facilita las comunicaciones con dispositivos bluetooth permitiendo a demás reutilizar la conexión entre distintas Clases/Actividades/Fragmentos dentro de la misma aplicación.

Cuando un usuario utiliza una aplicación se mueve por las distintas actividades de la misma provocando que estas pasen por distintos estados de su ciclo de vida. Las aplicaciones de Android no se cierran cuando pasan a estar en segundo plano, sino que el sistema las mantiene aletargadas mientras no necesita los recursos que estas utilizan. En caso de que Android necesite los recursos de las actividades de estas aplicaciones, las destruirá.

Si el usuario decide volver a una de estas actividades, el sistema reiniciara el proceso que las contiene, por lo que es el programador el que debe asegurarse de guardar el estado de las actividades antes de que sean destruidas y restaurarlo cuando sean reiniciadas. Es por esto, que es sumamente importante conocer el ciclo de vida de las actividades en Android (ver Ilustración 22).

²⁸ <https://github.com/prasad-pp/Android-Bluetooth-Library>

Como podemos consultar en la documentación de Android developers²⁹, una actividad de Android puede estar en 4 estados diferentes:

- *Running*: la actividad está en primer plano y tiene el foco
- *Paused*: la aplicación que contiene la actividad está activa, pero la actividad es visible pero no tiene el foco.
- *Stopped*: la actividad no es visible. Hay que guardar el estado de la actividad por si el sistema decide pasarla a estado destroyed.
- *Destroyed*: este estado se dispara cuando el sistema decide terminar la actividad o se invoca el método `finish()`.

La clase *Activity* de Android proporciona seis devoluciones de llamada que nos permite saber qué cambio de estado se está produciendo.

Cuando se crea por primera vez una actividad en Android, el sistema lanza el evento `onCreate()` seguido de `onStart()`. Finalmente, justo antes de que una actividad interactúe con el usuario se lanza el evento `onResume()`, pasando a estar visible en el dispositivo.

Cuando el usuario deja la Actividad en segundo plano, esta pasa a estado pausado lanzándose el evento `onPause()`. En este instante es importante que guardemos el estado de la actividad por si el sistema necesita utilizar los recursos de la misma. Así, más adelante podremos recuperar el estado de la actividad cuando se lance el evento `onResume()`.

Cuando la Actividad ya no es visible para el usuario el sistema lanza el evento `onStop()`. Si volvemos a la Actividad antes de que el sistema Android la destruya porque necesita sus recursos se lanzara el evento `onRestart()` seguido de `onStart()` y `onResume()` volviendo la actividad a pasar a primer plano. Justo antes de que la actividad se destruya definitivamente, el sistema lanzará el evento `onDestroy()`. Cuando salimos de una actividad de forma ordenada, se lanza el evento `onDestroy` antes de cerrar del proceso. En nuestra aplicación nos interesa controlar los eventos `onPause()`

²⁹ <https://developer.android.com/>

y `onResume()`, ya que son los estados entre los que va a estar alternando cuando esté operativa (ver Ilustración 23).

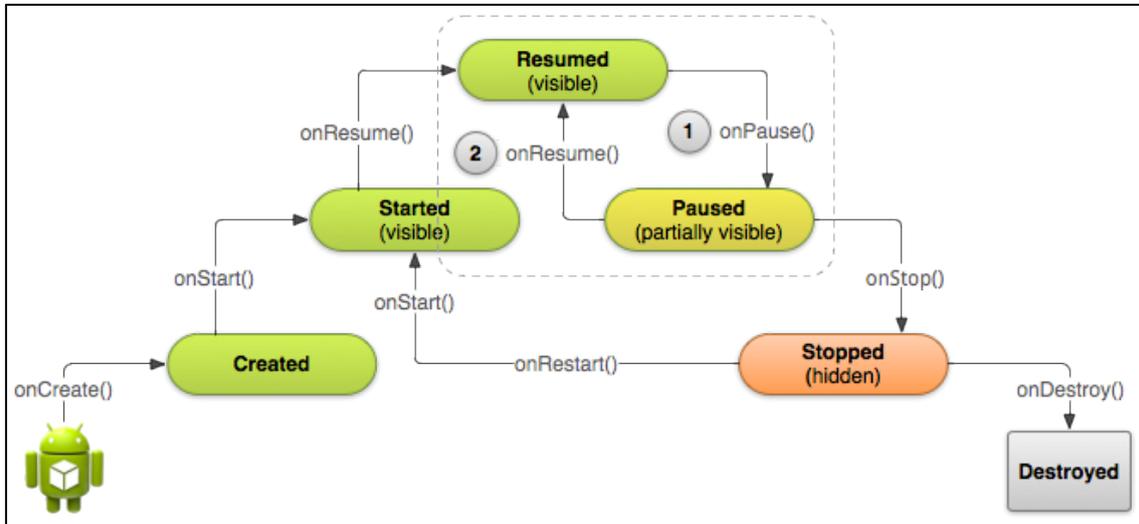


Ilustración 23. Eventos `onPause()` y `onResume()`

Imagen obtenida de <https://stuff.mit.edu/afs/sipb/project/android/docs/training/basics/activity-lifecycle/>

Nuestra aplicación se compone de 3 Actividades distintas, cuya interfaz podemos dividir en 3 bloques:

- La parte superior de la pantalla nos permite cambiar entre las tres Actividades que conforman la aplicación, así como activar/desactivar la conexión bluetooth. La conexión bluetooth con el dispositivo Arduino, requiere que en el dispositivo se active previamente el bluetooth y tengamos vinculado el dispositivo.



Ilustración 24. Menú superior de la aplicación

- El siguiente bloque sería la parte central de la aplicación que contiene todos los elementos de control del robot, de programación o preferencias que se pueden ajustar, según la Actividad en la que nos encontremos
- Finalmente, el último bloque es una pequeña zona con información de estado, que nos indica en todo momento el estado de la aplicación (si estamos conectados o no, si se ha producido un error, etc.)

La primera vez que abrimos la aplicación, los elementos de la interfaz que necesitan conexión previa con el módulo Arduino, están deshabilitados hasta que se establezca la conexión bluetooth con el dispositivo.

La actividad que permite cambiar los ajustes (ver Ilustración 26) está siempre disponible para el usuario. En esta actividad indicaremos la dirección MAC del adaptador bluetooth que incorpora la placa Arduino, además de todos los parámetros necesarios para la operación programada de la plataforma.

Cuando accedemos a esta actividad, se cargan desde el almacenamiento del dispositivo Android los parámetros de la aplicación, en caso de ser la primera vez que se ejecuta la aplicación, se establecen unos parámetros por defecto y se almacenan en el dispositivo para futuras ejecuciones.

Al abandonar la actividad y de forma automática, si se ha producido algún cambio en alguno de los parámetros estos se guardan de forma transparente al usuario. Los campos que contienen los valores aplican un filtro para que no se pueda introducir un valor equivocado, es decir, si debe introducirse un valor numérico, no es posible introducir letras. En el campo de la dirección MAC, se aplica un filtro de forma que solo permite introducir caracteres hexadecimales que son los que se utilizan para este tipo de direcciones hardware.

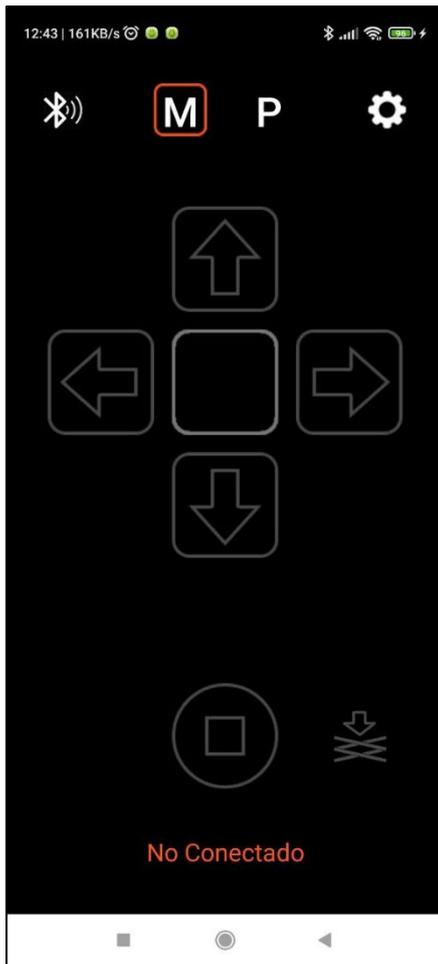


Ilustración 25. Actividad Principal

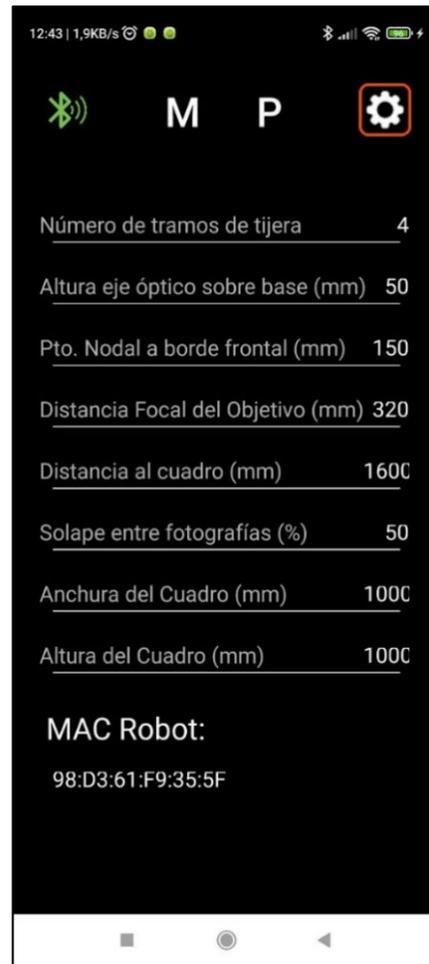


Ilustración 26. Actividad de Preferencias

Una vez se consigue conectar con el dispositivo Arduino, se activan los botones de la interfaz. La Actividad principal (ver Ilustración 25 e Ilustración 27), muestra una botonera que permite manejar el robot de forma manual. En el centro dispone de un cuadro de texto en el que introducir los centímetros que queremos que se desplace el robot, bien en sentido horizontal o en el eje vertical (elevando/bajando la plataforma).

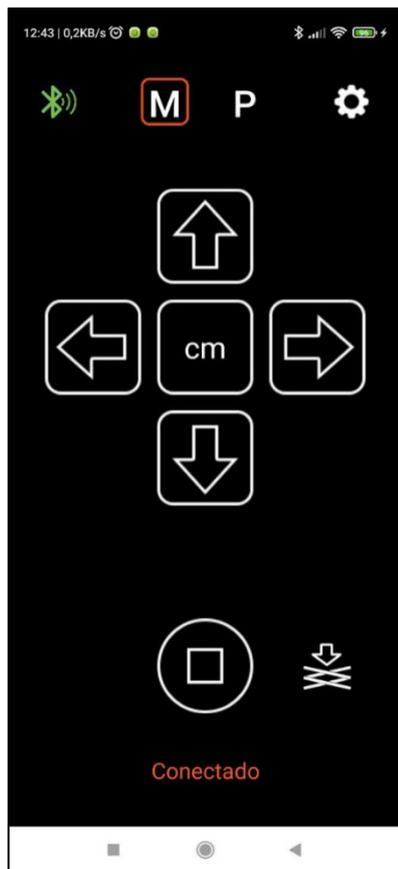


Ilustración 27. Pantalla uso manual



Ilustración 28. Pantalla programación

La actividad de funcionamiento programado (opción P) se encuentra inicialmente con todos sus elementos deshabilitados si no se ha establecido la comunicación con la plataforma robot (ver Ilustración 28). Inicialmente se activa el botón *“INICIAR PROGRAMACIÓN”*. Una vez lo pulsamos se activarán el resto de botones.

Justo debajo de este botón, se encuentran todos los parámetros necesarios para el funcionamiento semiautomático de la plataforma, que son calculados cada vez que se accede a esta actividad por si ha habido algún cambio en los ajustes. En esta parte también se indica la posición actual del robot en el mosaico de imágenes.

El botón *“AVANZAR POSICIÓN”*, avanza una posición en el recorrido del mosaico de fotografías. El botón *“RETROCEDER POSICIÓN”*, nos permite retroceder una posición en el recorrido del mosaico de fotografías.

Dado que el módulo bluetooth empleado en la plataforma solo puede recibir información, por lo que la comprobación de que no se produce ningún error en la

comunicación debe hacerse de forma visual. Para solventar el caso en el que la plataforma no reciba el comando, se ha incluido un botón que nos permite repetir el último movimiento en caso que observemos que no se ha ejecutado (botón “*REPETIR COMANDO*”).

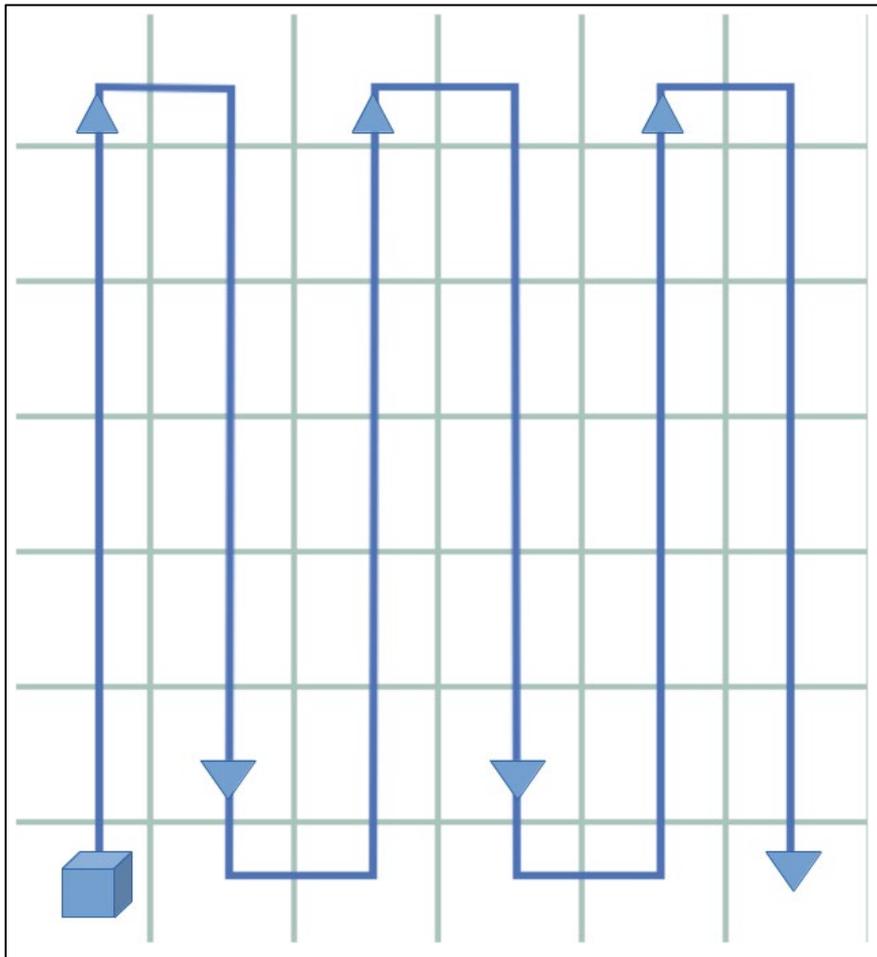


Ilustración 29. Ejemplo de movimientos calculados para el mosaico

Como se puede observar en la Ilustración 11, las ruedas motrices se encuentran en la parte izquierda de la plataforma mirándolo frontalmente. Dado que la parte frontal es la que se orienta hacia la obra que está en la pared, el movimiento natural de la plataforma es hacia la derecha. Como se puede apreciar en la Ilustración 29, para realizar el mosaico de imágenes, situaremos el robot inicialmente en la esquina inferior izquierda.

Realizaremos una primera fotografía y el primer paso que hará el robot subir una celda hacia arriba. Una vez se alcance la última celda de esa columna y se realice la captura correspondiente, el robot se desplazará una celda a la derecha hasta la siguiente columna, tomaremos una fotografía, y los siguientes movimientos serán en sentido descendiente hasta alcanzar la celda más baja de esa columna. Se repetirá este proceso hasta alcanzar la última celda del mosaico. En la Ilustración 29 podemos ver un ejemplo del recorrido que realizaría el robot para un mosaico de 7 filas x 6 columnas.

Como las placas Arduino tienen microprocesadores con poca potencia de cálculo, tenemos que enviarle la distancia que debe recorrer en milímetros convertidas a su equivalente en pasos del motor. En el caso del motor que realiza el desplazamiento en horizontal, la correlación es lineal tal que 1mm se corresponde con 133,3 pasos del motor. En el caso del motor que controla el eje vertical, nos encontramos con que la distancia que debe desplazarse la plataforma donde va situada la cámara y los pasos que da el motor no tienen una relación lineal.

Para calcular la ecuación que nos dará los pasos que debe dar el motor en función de la distancia en milímetros que queremos que se desplace la plataforma, se tomaron más de 60 medidas para ver la correspondencia entre pasos y distancia que se desplaza la plataforma. Con estos parámetros se realizó un ajuste de curvas mediante Excel cuyo resultado puede verse en la gráfica de la Ilustración 30. Esta función nos permite obtener los pasos que se ha de mover el motor para que la plataforma recorra una determinada altura expresada en milímetros:

$$f(x) = -9E-16x^6 + 8E-12x^5 - 3E-08x^4 + 6E-05x^3 - 0,0645x^2 + 33,35x - 6268,2$$

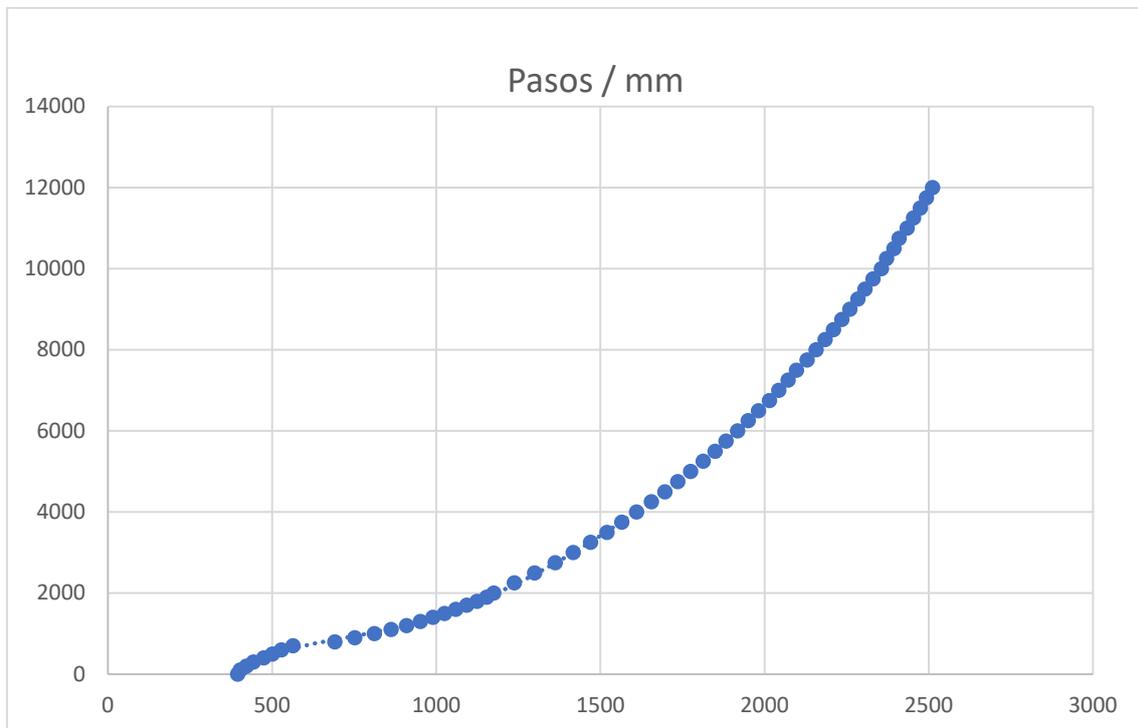


Ilustración 30. Correspondencia entre pasos y milímetros

En lo que respecta a la plataforma hardware utilizada para el desarrollo y depuración de la aplicación, se ha utilizado un teléfono móvil Xiaomi Redmi Note 9 Pro³⁰. Las características más destacables de este dispositivo son:

- Procesador Qualcomm® Snapdragon™ 720G de ocho núcleos con frecuencia hasta 2.3GHz
- Procesador de gráficos Adreno™
- Batería de polímero de iones de litio no extraíble 5020mAh
- Puerto de conector reversible USB tipo C
- Pantalla 6.67" DotDisplay
- Resolución: 2400 x 1080 FHD+
- Cámara trasera Cámara cuádruple IA de 64 MP
- Cámara ultra gran angular de 8 MP

³⁰ <https://www.mi.com/es/redmi-note-9-pro/>

- Cámara frontal 16 MP
- Almacenamiento y RAM 6GB + 128GB*
- Red y conectividad
 - GSM: B2/3/5/8
 - CMDMA: B1/2/4/5/8
 - TDD LTE: B38/40/41 (2535-2655MHz)
 - FDD LTE: B1/2/3/4/5/7/8/20/28
 - Red inalámbrica: 802.11a/b/g/n/ac
 - Compatible con Bluetooth 5.0

4. EXPERIMENTACION Y PRUEBAS

Los trabajos que se han realizado abarcan la instalación del cableado, la soldadura de conectores, el montaje de la electrónica, el testeo de los componentes electrónicos, la programación de la placa Arduino y la programación de la aplicación Android.

Los trabajos de soldadura han sido algo complicados dado mis conocimientos bastante escasos en este ámbito. Tengo que agradecer a mis compañeros de trabajo Salvador y Juanjo que me echara una mano en este apartado enseñándome a hacer una buena soldadura.

En cuanto al desarrollo de la aplicación para Android, desde hace un tiempo, Google que es el desarrollador de Android Studio está apostando muy fuerte por el lenguaje de programación Kotlin, por lo que en ocasiones me ha resultado difícil encontrar ejemplos en Java, ya que es más fácil encontrar ejemplos de nuevas funcionalidades de Android implementadas en Kotlin.

Inicialmente fue difícil entender el ciclo de vida de las actividades y después, surgieron bastantes problemas a la hora de realizar las comunicaciones por bluetooth. Si se incluyen las comunicaciones en el mismo hilo que ejecuta la interfaz gráfica, esta se congela hasta que finaliza la operación. Esta aproximación no es viable por lo que inicialmente se optó por mover todas las comunicaciones a un hilo independiente, lo cual después de distintas pruebas funcionaba bien, pero entonces, apareció el siguiente problema. Cuando hay varias actividades no es posible compartir la conexión si se realiza en un hilo, por lo que hay que implementar un servicio.

Los servicios en Android permiten realizar tareas en segundo plano y ser compartidos entre actividades de la misma aplicación o en el caso de los servicios de Android, se pueden compartir incluso entre distintas aplicaciones. Al igual que las actividades, los servicios tienen su propio ciclo de vida (ver Ilustración 31) y han de ser declarados en el manifiesto de la aplicación.

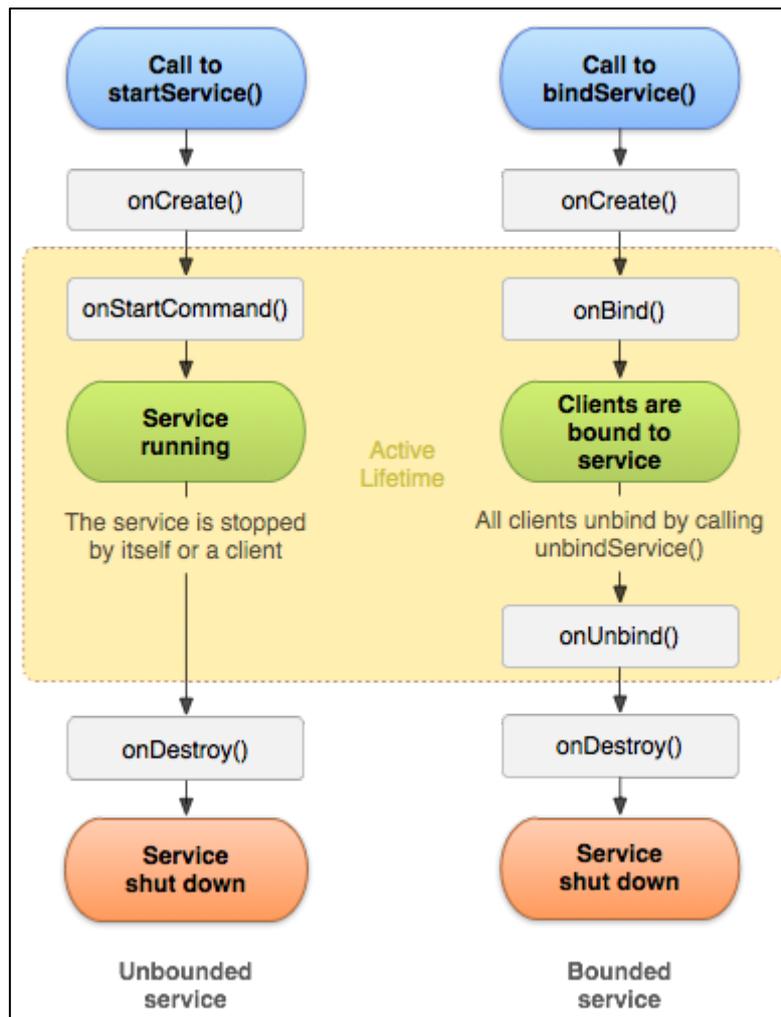


Ilustración 31. Ciclo de vida de un servicio Android

Imagen obtenida de <https://developer.android.com/guide/components/services>

Después de realizar multitud de pruebas, no era posible mantener una conexión bluetooth fiable, por lo que se decidió buscar una librería que ya implementase el servicio. Se comprobó que muchas librerías libres eran antiguas y estaban desarrolladas para versiones de Android que no compilaban con el SDK actual de Android. Finalmente se encontró la librería desarrollada por Prasad Parshram, que facilita mucho el trabajo, ya que permite compartir la conexión bluetooth entre distintas actividades de la aplicación.

A continuación, se pueden ver algunas instantáneas tomadas durante el montaje y las pruebas de la plataforma robotizada.

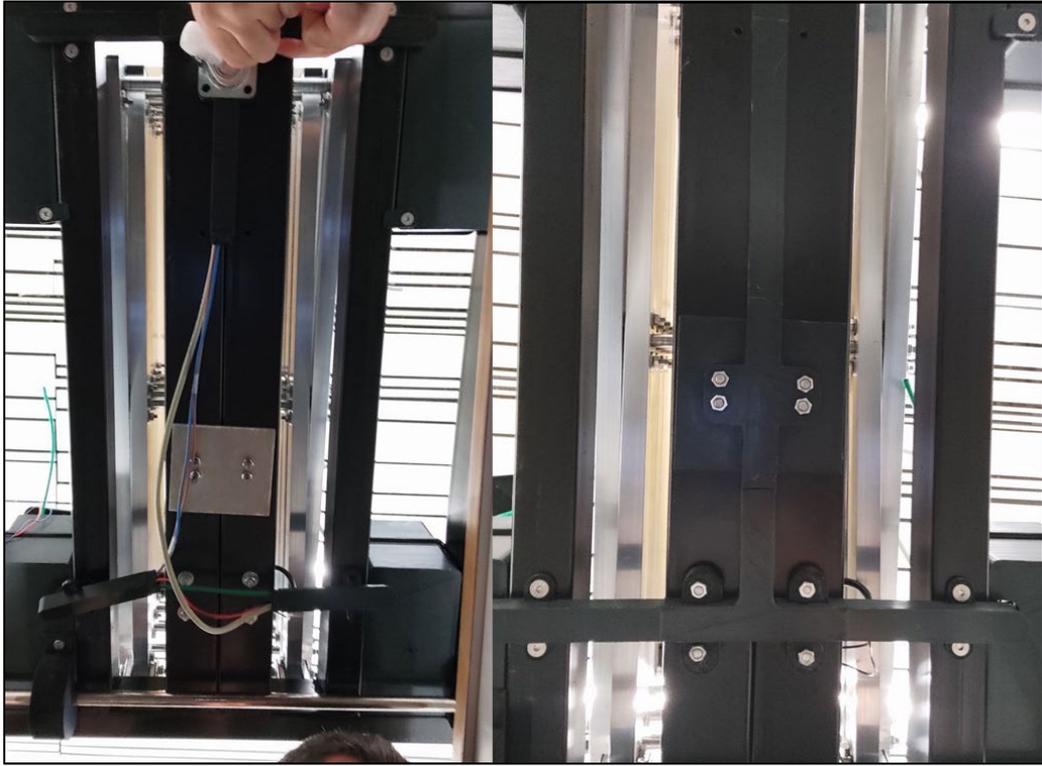


Ilustración 32. Cableado del robot

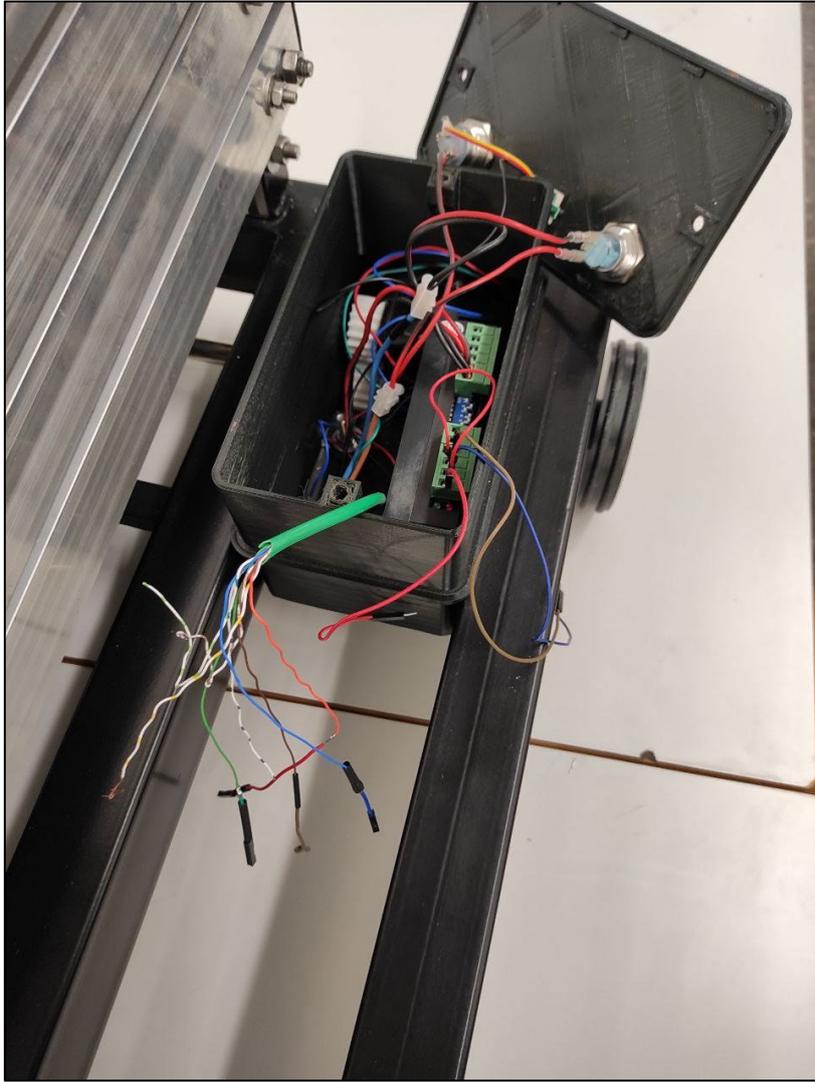


Ilustración 33. Instantánea de parte de la electrónica

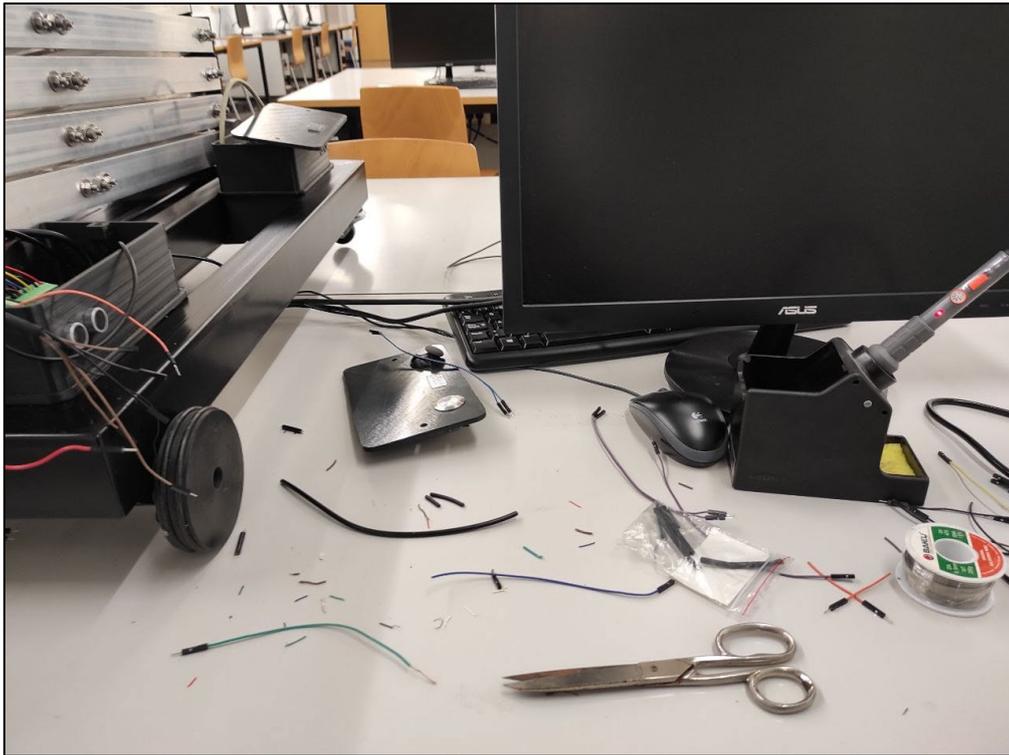


Ilustración 34. Trabajos de soldadura

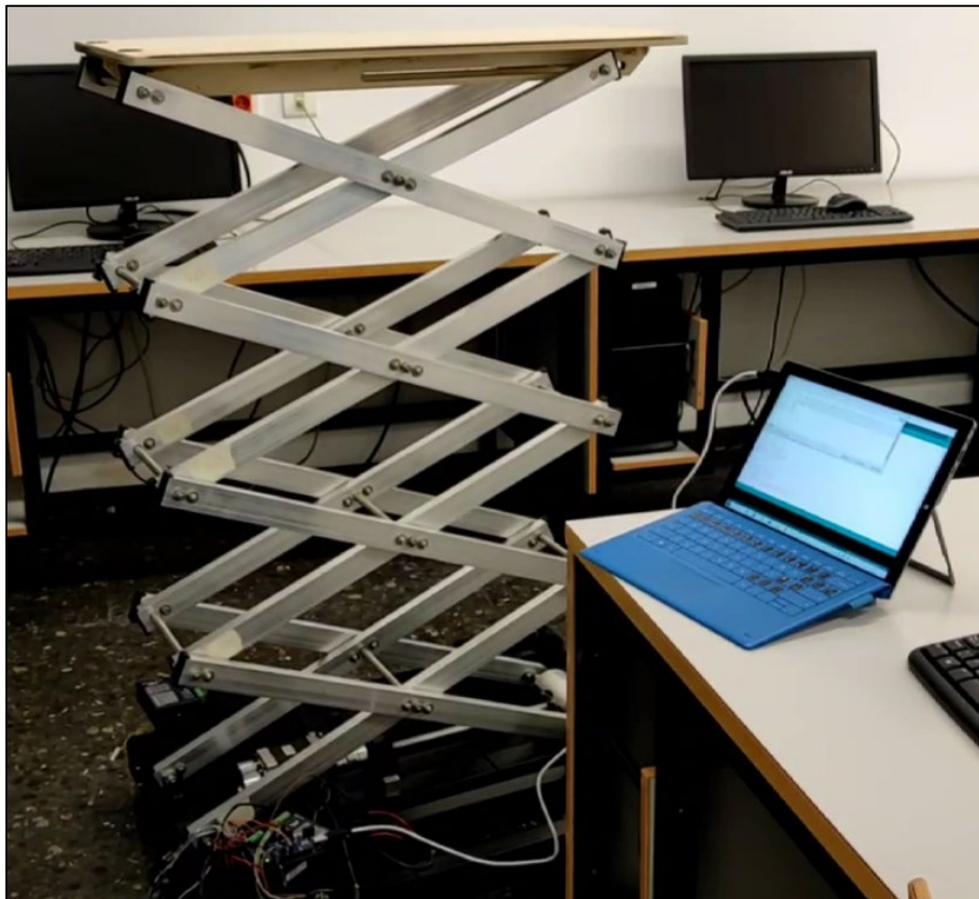


Ilustración 35. Pruebas de los motores

Como cualquier prototipo, esta plataforma ha pasado por una fase de concepto, una fase de diseño y la fase de construcción del prototipo. Ha sido en esta última fase donde se han puesto a prueba todos los aspectos que en la fase de concepto y diseño parecían estar contemplados.

Durante la fase de montaje, nos hemos encontrado con que algunas piezas impresas en 3D se han roto debido a la presión ejercida al apretar los tornillos, o algunas simplemente porque se imprimieron cuando se inició el diseño del prototipo y el material utilizado (PLA) se degrada con el tiempo y pierde parte de sus propiedades. También se ha comprobado que algunas aberturas que en la fase inicial de diseño parecían lo suficientemente grande, han necesitado ser ampliadas para el paso del cableado.

Los motores NEMA han funcionado como se esperaba, excepto el motor NEMA17 que inicialmente no conseguía mover la plataforma. El problema era que el rodamiento impreso en 3D para la transmisión de este motor, no podía desarrollar la suficiente potencia. Este problema se solucionó instalando una reductora de las que se usan en sistemas planetarios, con lo que se consiguió que el motor moviese la plataforma a cambio de reducir su velocidad de desplazamiento.

En el apartado de la electrónica también nos encontramos con algunos problemas, aunque los componentes proceden de Amazon y otras tiendas on-line españolas, la calidad, sobre todo de las soldaduras no han sido lo esperado. En algunas piezas hemos tenido que repasar las soldaduras, como por ejemplo en el relé que controla la alimentación.

Otro problema que surgió es que la plataforma de tijera tiene pequeñas oscilaciones cuando está desplegada, esto provoca que, sin la iluminación adecuada, las imágenes captadas aparezcan borrosas. Esto es problemático, ya que la mayoría de los museos cuando se trata de obras de arte, no permite el uso de flashes.

Una posible solución sería una aproximación similar a lo que se hace en Astronomía para eliminar ruido en las fotografías. El procedimiento consiste en que, para cada instantánea, se toman varias fotografías que posteriormente, se procesan con

una aplicación específica como la aplicación gratuita DeepSkyStacker³¹ que en función de los puntos comunes de las instantáneas tomadas crea una imagen nítida promedio. El problema es que, tratándose de obras de arte, quizás este sistema podría no ser del todo fidedigno y además añade una carga de trabajo adicional a todo el proceso de la generación de las imágenes gigapíxel y su preparación para ser visualizadas que es ya de por sí una tarea costosa.

³¹ <http://deepskystacker.free.fr/english/index.html>

5. CONCLUSIONES Y TRABAJO FUTURO

El desarrollo de este proyecto ha sido muy enriquecedor por mi parte. He aprendido mucho sobre las imágenes gigapíxel y la toma de fotografías gracias a la inestimable ayuda de uno de mis tutores, Pedro M. Cabezos, que tiene amplia experiencia en este ámbito y del cual partió el diseño de la plataforma robot.

En el apartado de la programación también he tenido que ponerme al día. Aunque había realizado alguna incursión en la programación en C para placas Arduino, siempre habían sido proyectos muy sencillos.

A pesar de los problemas que han ido surgiendo, se han solucionado y ha sido posible alcanzar los objetivos del proyecto.

Respecto al trabajo futuro de este proyecto, se podría realizar un repositorio web de obras de arte de los museos más importantes, similar al que ya ha realizado el profesor Pedro M. Cabezos para su proyecto incluyendo obras de arte de más museos.

Otra vía de trabajo podría ser la migración de las comunicaciones entre la placa Arduino y la aplicación Android a un canal de comunicaciones WIFI en lugar de bluetooth, lo que le conferiría mayor rango de alcance y podríamos añadir fácilmente la comunicación bidireccional entre plataforma y aplicación de control. Es posible añadir la conectividad WIFI a Arduino con un shield específico o simplemente añadiendo un chip

ESP8266³² (ver Ilustración 36). El chip ESP8266 se lanzó en 2014 e inicialmente era bastante difícil de programar.

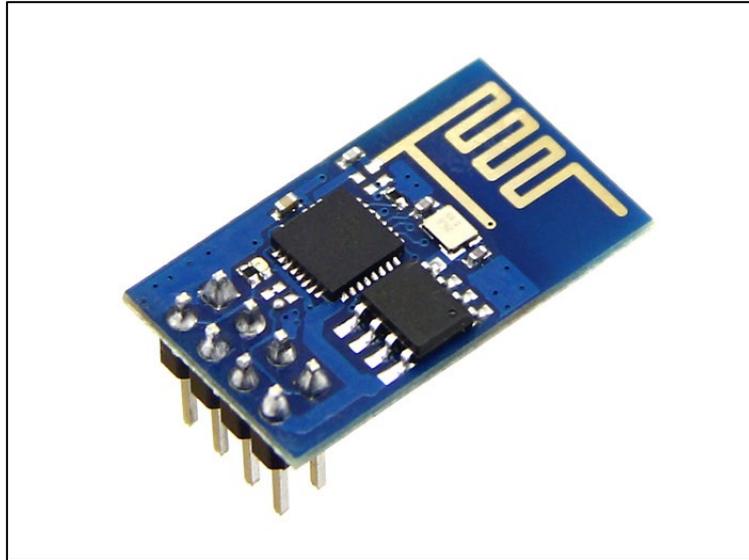


Ilustración 36. Chip esp8266

Imagen obtenida de <https://descubrearduino.com/esp8266/>

Una opción que puede resultar más interesante sería utilizar el chip ESP32³³ en lugar de una placa Arduino. Este chip, al igual que Arduino, es un microcontrolador que además tiene conectividad Wifi y Bluetooth integrada. En la Ilustración 37 podemos ver una imagen de un chip ESP32 junto su correspondiente asignación de pines.

³² <https://esphome.io/>

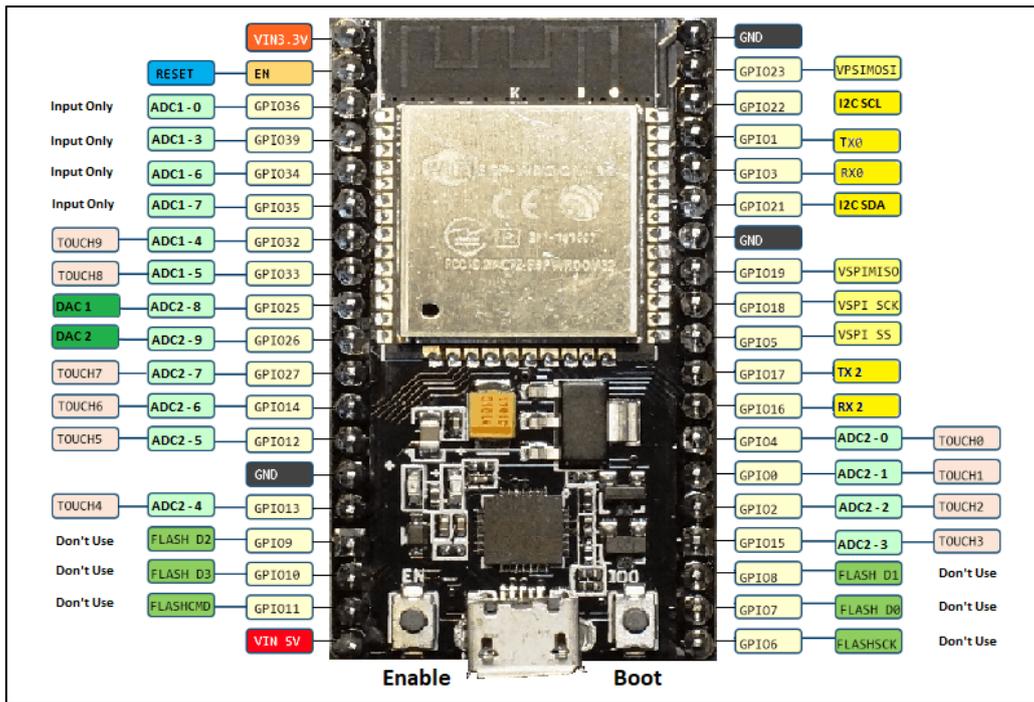


Ilustración 37. Chip ESP32

Imagen obtenida de <https://www.circuitos-electricos.com/esp32-especificaciones-y-disenos/>

En la Ilustración 38, se podemos una comparativa de las características técnicas más importantes y su diferencia entre una placa Arduino UNO y el ESP32.

	Arduino UNO	ESP32
Dimensiones	6.8 x 5.3 cm	5 x 2.7cm
Pines E/S	14	36
Pines PWM	6	16
Pines analógicos	6	Hasta 18
Procesador	ATMega328P	Xtensa Dual Core 32-bit LX6
Memoria Flash	32 kB	4 MB
SRAM	2 kB	520 kB
Velocidad de Reloj	16 MHz	Hasta 240 MHz
Voltaje	5V	3.3V
Puertos serie HW	1	3
Wifi	No	802.11 b/g/n
Bluetooth	No	V4.2 + EDR y BLE
Sensores táctiles	No	10

Ilustración 38. Arduino Uno comparado con ESP32

Otra posible vía de investigación sería buscar una opción de robot que, manteniendo las premisas de ser ligero y compacto, permitiera eliminar las pequeñas oscilaciones que se producen en la plataforma que sostiene la cámara debido a la tensión que soporta la tijera.

Una posible propuesta sería cambiar el sistema de elevación del soporte de la cámara por uno similar al que utilizan las impresoras 3D. La idea sería una plataforma rectangular como la actual, con 2 soportes desmontables verticales (o 4 para dar mayor estabilidad) para que el soporte de cámara se eleve al igual que lo hace el cabezal de impresión de algunas las impresoras 3D como se puede ver en la Ilustración 39 .

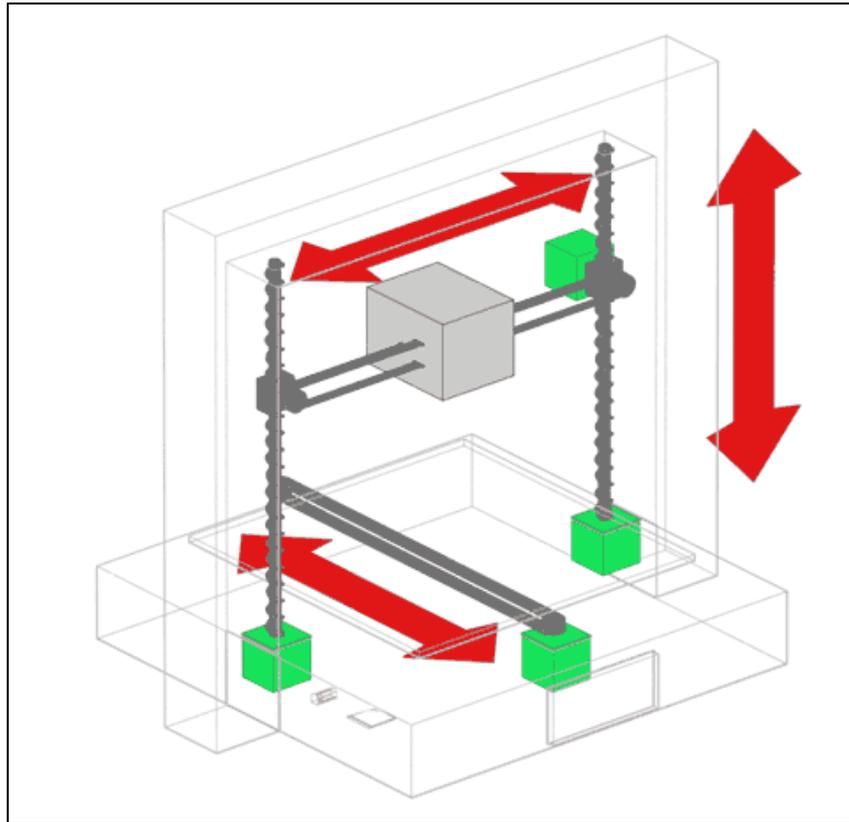


Ilustración 39. Cinemática impresora 3D

Imagen obtenida de <https://www.luisllamas.es/como-funciona-la-cinematica-de-una-impresora-3d-fff/>

El problema de esta aproximación vendría dado por la limitación para transportar estas barras que limitarían la altura que podría alcanzar el robot. Esto se podría solventar utilizando soportes telescópicos en caso de que esta hipótesis fuese viable.

6. REFERENCIAS

- [1] Flint, G. (2000). *The gigapixel Project*. Consultado en: <https://web.archive.org/web/20060719063806/http://www.gigapxl.org> (recuperado: 01/03/22).
- [2] Brady, D. J., Gehm, M. E., Stack, R. A., Marks, D. L., Kittle, D. S., Golish, D. R., Vera, E. M. y Feller, S. D., (2012). *Multiscale gigapixel photography*. *Nature* [en línea]. 486(7403), 386–389. [Consultado el 14 de mayo de 2022]. Disponible en: doi: <https://doi.org/10.1038/nature11150>
- [3] Cabezos Bernal, P. M., (2021). Proyecto – Gpix [en línea]. Gpix. [Consultado el 2 de mayo de 2022]. Disponible en: <https://gpix.webs.upv.es/index.php/proyecto/>
- [4] Cabezos Bernal, Pedro M.; Rodríguez Navarro, P.; Gil Piqueras, T.; Cisneros Vivó J. y Gil Gil C. (2022). *Captura fotográfica gigapíxel de obras de arte*. Valencia: edUPV
- [5] Liu, Y.; Gadepalli, K.; Norouzi M.; Dahl G.E.; Kohlberger T.; Boyko A.; Venugopalan S.; Timofeev A.; Nelson P.Q.; Corrado G.S.; Hipp J.D.; Peng L.; Stumpe M.C. (2017). *Detecting Cancer Metastases on Gigapixel Pathology Images*. arXiv. <https://doi.org/10.48550/arXiv.1703.02442>
- [6] Kopf, J., Uyttendaele, M., Deussen, O., Cohen, M. (2007) *Capturing and Viewing Gigapixel Images*. *ACM Trans. Graph*, 26, 10. <https://doi.org/10.1145/1276377.1276494>.
- [7] Wang, X.; Zhang, X.; Zhu, Y.; Guo, Y.; Yuan, X.; Xiang, L.; Wang, Z.; Ding, G.; Brady, D.; Dai, Q.; Fang, L. (2020). *PANDA: A Gigapixel-level Human-centric Video Dataset*. <https://arxiv.org/abs/2003.04852>

[8] Gálvez Gabarda, J., (2014). *Mosaic | La imagen gigapixel* [en línea]. Mosaic. Disponible en: <https://mosaic.uoc.edu/2014/12/05/la-imagen-gigapixel/>

[Consultado el 4 de mayo de 2022]

[9] Son, Hui & Marks, Daniel & Brady, David & Kim, Jungsang. (2013). *Oversampled triangulation of AWARE-10 monocentric ball lens using an auto-stigmatic microscope*. Optics express. 21. 22206-22214. 10.1364/OE.21.022206.

[10] *PANDA: A Gigapixel-level Human-centric Video Dataset* (Xueyang Wang et al. 2020). Disponible en:

https://openaccess.thecvf.com/content_CVPR_2020/papers/Wang_PANDA_A_Gigapixel-Level_Human-Centric_Video_Dataset_CVPR_2020_paper.pdf

[11] *Comunicado del gabinete de prensa del Museo Reina Sofía*, publicado el 4 de mayo de 2020. Disponible en: <https://www.telefonica.com/es/sala-comunicacion/el-museo-reina-sofia-lanza-gigapixel-un-espacio-web-que-permite-explorar-obras-de-su-coleccion-con-resolucion-ultra-hd/>

[Consultado el 2 de mayo de 2022]



ANEXO I. OBJETIVOS DE DESARROLLO SOSTENIBLE

Grado de relación del trabajo con los Objetivos de Desarrollo Sostenible (ODS).

Objetivos de Desarrollo Sostenibles	Alto	Medio	Bajo	No Procede
ODS 1. Fin de la pobreza.				X
ODS 2. Hambre cero.				X
ODS 3. Salud y bienestar.				X
ODS 4. Educación de calidad.		X		
ODS 5. Igualdad de género.				X
ODS 6. Agua limpia y saneamiento.				X
ODS 7. Energía asequible y no contaminante.				X
ODS 8. Trabajo decente y crecimiento económico.				X
ODS 9. Industria, innovación e infraestructuras.				X
ODS 10. Reducción de las desigualdades.		X		
ODS 11. Ciudades y comunidades sostenibles.				X
ODS 12. Producción y consumo responsables.				X
ODS 13. Acción por el clima.				X
ODS 14. Vida submarina.				X
ODS 15. Vida de ecosistemas terrestres.				X
ODS 16. Paz, justicia e instituciones sólidas.				X
ODS 17. Alianzas para lograr objetivos.				X



Reflexión sobre la relación del TFG/TFM con los ODS y con el/los ODS más relacionados.

En 2015, la ONU aprobó la Agenda 2030³⁴ sobre el Desarrollo Sostenible, una oportunidad para que los países y sus sociedades emprendan un nuevo camino con el que mejorar la vida de todos, sin dejar a nadie atrás.

La Agenda cuenta con 17 Objetivos de Desarrollo Sostenible, que incluyen desde la eliminación de la pobreza hasta el combate al cambio climático, la educación, la igualdad de la mujer, la defensa del medio ambiente o el diseño de nuestras ciudades. Respecto a la relación de este TFG con la Agenda 2030, podemos ver un vínculo directo importante con los ODS 4 y 10.

ODS 4: Garantizar una educación inclusiva y equitativa de calidad.

El desarrollo de este Trabajo de Fin de Grado (TFG) junto con el proyecto³⁵ del que deriva, ayudará a acercar un gran catálogo cultural a cualquier persona independientemente de su formación (estudiantes, profesorado, profesionales, etc.). Esto permitirá reducir las diferencias entre las personas puesto que no todas tienen las mismas posibilidades de acceso a la educación debido a desigualdades económicas, culturales, etc. Además, el desarrollo de las TICs resulta extremadamente importante para conseguir una educación de calidad en todo el mundo como hemos podido comprobar durante la pandemia del COVID-19 durante la cual se ha podido seguir con la docencia en numerosos países de forma telemática. De hecho, este proyecto permite acercar la cultura a cualquier persona del mundo simplemente con el requisito de tener una conexión a internet y un dispositivo como puede ser un *smartphone*, una *Tablet*, etc.

³⁴<https://www.un.org/sustainabledevelopment/es/>

³⁵Proyecto de Investigación denominado Captura Fotográfica de resolución gigapíxel para la documentación y divulgación del patrimonio pictórico, financiado con ayuda a Primeros Proyectos de Investigación (PAID-06-18)



ODS 10: Reducción de las desigualdades.

Este trabajo de fin de grado (TFG) elimina las barreras físicas o sociales que se puedan presentar para acceder a la cultura. Centrándonos en las desigualdades físicas, este proyecto permitirá que las personas con problemas de movilidad puedan contemplar y estudiar sin tener que desplazarse a un museo, las obras de arte que allí se exponen. También ayudará a personas con problemas visuales, ya que en el museo no es posible acercarse a las obras tanto como permiten las imágenes gigapíxel. Este proyecto también elimina las desigualdades económicas, ya que no todo el mundo puede costearse el desplazamiento y la entrada a los museos. Por último, considero que este proyecto ayudaría a eliminar las desigualdades sociales que se dan en algunos países donde el acceso a la cultura está vetado por cuestión de raza, religión o sexo).