Tesina de Master

Titulación:

Máster en Inteligencia Artificial, Reconocimiento de Formas e Imagen Digital

Titulo del Proyecto:

"Estudio Comparativo de Técnicas de Reconocimiento de Imágenes"

Autor:

Emilio Chapí Verdú

Tutor:

Roberto Paredes Palacios

Valencia. Febrero 2012

Abstract

Los sistemas de reconocimiento de imágenes han sido un foco de interés en los últimos años; el auge de internet, y de aplicaciones como google imágenes o facebook, ha hecho que la búsqueda de imágenes, el reconocimiento, y la verificación sea un tema a solventar.

En la presente Tesina realizamos un breve repaso por las diferentes tecnologías existentes en la actualidad, analizando las diferentes tareas a las que se enfrenta el reconocimiento de formas y algunos de los métodos usados para afrontarlas. También realizamos un estudio sobre algunos de los métodos más ampliamente usados y aportamos una nueva aproximación a la tarea de clasificación, que compararemos con sistemas existentes en el estado de la investigación.

Índice

1. Introducción **1**

1.1. Sistemas de reconocimientos de imágenes 1

1.1.1. Clasificación 1

1.1.2. Etiquetado 2

1.1.3. Reconocimiento de objetos 2

1.1.4. Búsqueda de objetos 2

1.1.5. Búsqueda de imágenes 2

1.2. Aplicaciones 3

2. Estado de la investigación **4**

2.1. Extracción de características 4

2.2. Representaciones usadas 8

2.3. Cálculo de la semejanza y clasificación 8

3. Metodología propuesta **13**

3.1. Características utilizadas 13

3.1.1. Sift Denso 13

3.1.2. Histograma de color Denso 17

3.2. Cuantificación vectorial 18

3.2.1. K-means 18

3.2.2. Proyecciones aleatorias 19

3.3. Combinación de vectores de características 20

3.3.1. Correlación espacial 21

3.4. Clasificación 22

4. Resultados **24**

4.1. Evaluación de los resultados 24

4.2. Bases de datos 24

4.2.1. Corel 24

4.2.2. Msrcorid 24

4.3. Características utilizadas 25

4.3.1. Función de distancia 25
4.3.2. Tamaño de rejilla 27
4.4. Método de cuantificación vectorial 28
4.5. Escala fija frente a Escala detectada 29
4.6. Tamaño de rejilla dependiente de la escala 29
4.7. Combinación de características 30
4.7.1. Concatenación de los vectores de características 31
4.7.2. Concatenación de la cuantificación vectorial 32
4.7.3. Correlación entre los grupos de pertenencia 32
4.8. Correlación espacial 33

5. Conclusiones y Trabajo Futuro **35**
5.1. Conclusiones 35
5.2. Trabajo Futuro 35

6. Bibliografía **37**

Índice de figuras

1. Esquema de un sistema “ Bag of Words” 9
2. Pirámide de Gaussianas y DOG 14
3. Cálculo de las características SIFT 16
4. Obtención de los puntos de interés de manera densa 17
5. Cálculo de la correlación espacial 21
6. Ejemplo de imágenes en la base de datos Corel 25
7. Ejemplo de imágenes en la base de datos Msrcorid 26
8. CER en función del tamaño de rejilla 27
9. Tamaño de rejilla dependiente de la escala 30
10. Concatenación de los vectores de características en función del tamaño de rejilla 31
11. Correlación entre los grupos de pertenencia 32
12. Resultados utilizando correlación espacial 33

Índice de cuadros

1. Ejemplo de cálculos de distancia 11
2. CER En función del cálculo de distancia para las bases de datos Corel y Microsoft 26
3. K-medias frente a Proyecciones aleatorias 28
4. Escala fija frente a Escala detectable 29
5. CER usando tamaño de rejilla dependiente de la escala 31
6. Resumen de resultados para la base de datos Corel 34
7. Resumen de resultados para la base de datos Msrcorid 34

1. Introducción

La necesidad inherente al ser humano de organizar, clasificar y recuperar el conocimiento de forma clara y rápida puede considerarse una de las claves para entender el progreso. Esta necesidad nos ha llevado a formas estructuradas de almacenar y disponer de la información. Ahora bien, como sabemos, disponemos de métodos informatizados para organizar y recuperar la información textual contenida en: libros, páginas web, manuales,... ¿Pero qué ocurre con la información almacenada en otro tipo de medios, ya sea sonido, video o imágenes?.

En la presente tesina de máster nos vamos a centrar en el reconocimiento de imágenes; concretamente en la tarea de clasificación de dichas imágenes. Entendiendo como clasificación la pertenencia a un grupo concreto de imágenes, con las cuales comparte una serie de atributos.

Primero, y como parte de esta introducción, revisaremos algunas de las formas en las que aparece el reconocimiento de imágenes en diversas aplicaciones. También se hará un breve repaso a los distintos tipos de reconocimiento que existen.

1.1. Sistemas de reconocimientos de imágenes

Dentro del ámbito del reconocimiento de imágenes se pueden observar una serie de subtipos que englobarían, en mayor o menor medida, distintas de las posibles aplicaciones. A continuación se detallan algunos de los subtipos más comunes.

1.1.1. Clasificación

Este tipo de sistemas asignan la imagen a una clase concreta. Cada clase está compuesta por una serie de imágenes que comparten una serie de características y representan conceptos semánticos que van desde simples objetos como ventanas o coches, hasta conceptos más generales como podían ser imágenes de naturaleza o de ciudades. Este tipo de sistemas se puede usar para la ayuda al diagnóstico como en [BMOM].

La presente tesina se encuentra englobada dentro de este tipo de sistemas de reconocimiento de imágenes ya que las pruebas que vamos a realizar consistirán en asignar una imagen a un grupo.

1.1.2. Etiquetado

En estos sistemas se busca asignar etiquetas a las imágenes. Dichas etiquetas podrían englobar a toda la imagen, en cuyo caso estaríamos hablando, en realidad, de sistemas de clasificación; también hay sistemas que asignan etiquetas a partes concretas de la imagen. De esta forma, cada objeto de la imagen iría acompañado de una etiqueta que le definiría. Algunos ejemplos de etiquetas podrían ser: mar, tigre, césped... Ejemplos de estos tipos de sistemas los podemos encontrar en: [BDF02, dFB01, DWK+05]

1.1.3. Reconocimiento de objetos

Dentro de este tipo se encuentran aquellos sistemas en los que lo que se pretende es reconocer objetos concretos como caras [AH+06] o huellas dactilares [JG05]. Existe una clasificación dentro de estos sistemas:

- Sistemas de reconocimiento.
  Este tipo de sistemas lo que intentan es reconocer el objeto como uno de los posibles objetos dentro de la base de datos. Ejemplo de este tipo de sistemas sería un escáner de huella dactilar que, al pasar el dedo por el lector, indicase quien es el sujeto.

- Sistemas de verificación.
  Estos sistemas verifican que un objeto, que previamente se ha identificado, es el objeto que se dice que es. Un ejemplo sería un sistema de entrada a un edificio en el que el usuario, tras haberse identificado, es verificado mediante la imagen de su rostro.

1.1.4. Búsqueda de objetos

Este tipo de sistemas permiten buscar objetos concretos dentro de una imagen. En este tipo de sistemas el usuario introduciría una imagen del objeto que desea encontrar y el sistema lo buscaría dentro de todas las imágenes de la base de datos. Ejemplo de este tipo de sistema son: [Low04, JZL96]

1.1.5. Búsqueda de imágenes

Este tipo de sistemas buscan, de entre todas las imágenes de una base de datos, aquellas imágenes que son más similares a la consulta. Existen distintas formas

de realizar la consulta, ya sea mediante texto introducido por teclado o con otra imagen con respecto a la cual se quieren buscar imágenes similares [b31, Deb04].

1.2. Aplicaciones

No podemos concluir nuestro repaso al estado de la investigación acerca de los sistemas de reconocimiento de imágenes sin incluir un breve comentario sobre los diversos campos donde estos sistemas se utilizarían. El ámbito más general en el que se pueden aplicar estos sistemas serían los buscadores genéricos (tipo Google o Yahoo!) donde los usuarios buscarían cualquier tipo de imagen en una base de datos amplia y genérica. Por otra parte existen otros campos donde la utilidad de estos sistemas se hace más patente como, por ejemplo, en aplicaciones médicas como [AXLT04] o en aplicaciones de cartografía o aplicaciones que utilicen fotografías aéreas o tomadas por satélite. En estos tipos de entorno los sistemas de reconocimiento de imágenes demostrarían su utilidad buscando imágenes similares a una tomografía o una radiografía, facilitando de esta manera el diagnóstico. También encuentran aplicación en museos, colecciones de arte o de imágenes de objetos robados donde facilitan la búsqueda de obras concretas. Existen otros ámbitos como en la producción donde un sistema de reconocimiento de imágenes puede detectar las piezas defectuosas para retirarlas de la cadena de montaje. O como sistemas de verificación y acceso a edificios o sistemas.

2. Estado de la investigación

A continuación se detalla el estado actual de la investigación en lo que se refiere a reconocimiento de imágenes. Para ello repasaremos las distintas fases que componen un sistema de reconocimiento de imágenes. Empezaremos por analizar algunos de los métodos existentes de extracción de características, después veremos los métodos de cuantificación vectorial más utilizados, y por último analizaremos distintas formas de clasificación y cálculo de semejanza entre imágenes.

2.1. Extracción de características

El problema de describir matemáticamente una imagen se da porque los ordenadores sólo disponen de la matriz de valores que representan el color de cada pixel de la imagen, y no del significado de dicha imagen (gap semántico). Para intentar suplir esta deficiencia, se extraen de dicha matriz las características visuales más relevantes como el color, la textura, la forma, etc. Las características son útiles porque tienen la propiedad de capturar cierta propiedad visual de la imagen [DJLW07]. La mayoría de sistemas extraen previamente las características de las imágenes de que disponen en la base de datos y las utilizan como entrada para los algoritmos de cálculo de la semejanza.

Una buena práctica a la hora de extraer las características de una imagen es segmentarla. Si decidimos dividir una imagen existen diversas formas de hacerlo. La más simple consiste en dividir la imagen en un número determinado de porciones; por ejemplo en cinco: las cuatro esquinas y el centro como vemos en [ADBP02, Joh00]. Existen otros modos de segmentación que toman como particiones las formas presentes en la imagen, como podemos observar en [BDF02] y en [dFBB$^+$02], aplicados en este caso a la auto anotación. También se puede segmentar una imagen y utilizar tan solo aquellos segmentos que sean relevantes para la consulta, como en *Blobworld*. En otros caso la segmentación se produce eligiendo los puntos de interés y se calculan características locales en dichos puntos como en [Low04] o en [USS09]. Este hecho produce una distinción añadida sobre las características; la contraposición entre características globales y locales. Las características globales tratarían de definir la imagen como un todo, mientras que las locales definirían regiones u objetos de la imagen. Las características locales surgen de la imposibilidad de reducir el gap semántico basándose tan solo en la representación global de la imagen. Estas características locales se calculan de igual forma que las globales, pero tomando tan solo una región de la imagen. Los avances indican que en

el futuro se fusionaran las características tanto globales como locales para proporcionar nuevas representaciones, así mismo la disponibilidad de imágenes en tres dimensiones y estereoscópicas se utilizaran para obtener características más significativas. "Reducir el tándem gap sensorial y gap semántico debe seguir siendo el objetivo principal"[DJLW07].

Aunque en la lista que se muestra a continuación no se incluyen todas las características que se pueden extraer de una imagen se hace un repaso sobre las más frecuentes y relevantes:

- Color: De entre todas las características que se pueden extraer, una de las más frecuentes es el color, que se mide utilizando su histograma, es decir: la frecuencia de aparición de cada valor de color en la imagen, ya sea en niveles de gris o en color. Está demostrado que el espacio en el que midamos el color influye en su posterior procesado, y no todos coinciden por igual con la visión humana. Ejemplos de diferentes tipos de espacios, extraídos de [Joh00, DJLW07, JFS95] serían los siguientes:

  - RGB: Es el espacio más común, y está compuesto de rojo verde y azul (red, green, blue).
  - YIQ y YUV: Se utiliza en los estándares PAL y NTSC. Son transformaciones lineales de RGB; YIQ se calcularía como:

$$\begin{aligned} Y &= 0{,}299R + 0{,}587G + 0{,}114B \\ I &= 0{,}5957161349R - 0{,}2744528378G - 0{,}3212632971B \\ Q &= 0{,}2114564021R - 0{,}5225910452G + 0{,}3111346432B \end{aligned}$$

    y YUV como:

$$\begin{aligned} Y &= 0{,}299R + 0{,}587G + 0{,}114B \\ U &= 0{,}492(B - Y) = -0{,}147R - 0{,}289G + 0{,}436B \\ V &= 0{,}877(R - Y) = 0{,}615R - 0{,}515G - 0{,}100B \end{aligned}$$

  - HSV (tinte, saturación y valor) y HSI (tinte, saturación e intensidad): Los experimentos de evaluación con usuarios muestran que este espacio de color

se ajusta más a la percepción humana, aunque no es perfecto. Por otra parte este espacio presenta la ventaja de ser menos sensible a la iluminación.

- LUV: También conocido como CIELUV, parece coincidir mejor con la visión humana.

La principal ventaja de esta característica radica en que es muy sencilla, rápida y eficiente de calcular, con lo que siempre se incluye a la hora de extraer las características.

- Textura: Las características referidas a las texturas están diseñadas para capturar la granularidad y los patrones repetitivos en las superficies de una imagen. Por ejemplo los ladrillos, el papel pintado, la arena... Su utilización es vital para el reconocimiento de imágenes médicas o aéreas, ya que, en estos casos, la textura está íntimamente relacionada con la semántica de la imagen. Una manera común de extraer esta característica es utilizar los coeficientes de alguna transformada como: la transformada discreta del coseno o la transformada wavelet. Esta última se usa los descriptores de textura basados en cadenas de Markov, que son ampliamente utilizados como características de la imagen.

- Esquinas y su orientación: Pueden ser extraídos de diversas formas: filtros gabor y laplacianos, detectores de esquinas Canny, operadores Sobel o el gradiente de la imagen. Un ejemplo de la extracción de esta característica lo podemos ver en [IA02], donde utilizan las uniones en "U", en "L" y las líneas paralelas para reconocer objetos fabricados por el hombre; ya que la mayoría de estos objetos son rígidos y están compuestos de líneas paralelas y esquinas.

- Forma: Otra de las características más comunes es la forma. Su representación es robusta y eficiente, y juega un papel fundamental en la recuperación de la información. A la hora de utilizar este tipo de características tan importante es la representación de la forma como la comparación de formas entre sí. Las formas y regiones se pueden representar por su frontera (código-cadena). A partir de esta representación se pueden calcular una serie de descriptores:

  - Características simples de la forma: Área, circularidad, diámetro, excentricidad, orientación de eje mayor,...

- Características complejas de la forma: Por ejemplo; Descriptores de Fourier, momentos de la forma, espectro de la forma de la imagen, etc.

También se puede utilizar la forma directamente para el cálculo de la semejanza, por ejemplo calculando la mínima energía necesaria para que los contornos sean iguales. Además de la forma podemos encontrarnos una serie de características que se derivan directamente de ella, como podrían ser la curvatura o las plantillas. Las plantillas toman una región u objeto de la imagen y lo utilizan como filtro a la hora de buscar la imagen consulta. También se puede construir un grafo a partir de las formas presentes en la imagen. En [dFBB+02] podemos ver un sistema que reconoce las formas de los objetos presentes en la imagen, además guarda la información referente a la localización espacial de los objetos entre sí (característica que veremos más adelante). Cabe destacar que uno de los inconvenientes de usar las formas es la variación que sufren al cambiar el punto de vista.

- Posición: Localización espacial de las características, las regiones y los objetos.
- Relaciones espaciales: Localización relativa de los objetos y regiones entre sí; "el objeto 1 se encuentra debajo y a la derecha del objeto 2".
- Áreas de interés marcadas a mano: El usuario puede marcar áreas, no necesariamente objetos, como zonas de interés para las que serán calculadas las características.
- Puntos de interés. Este tipo de características se calculan en los puntos que el algoritmo considera relevantes para extraer las características; de esta forma se obtiene una representación de aquellos puntos que mejor definen a la imagen. Algunos de los más destacados son los descriptores SIFT [Low04] que definiremos de manera extensa en 3.1.1 ya que son una de las características que utilizaremos en la metodología propuesta y SURF [BTVG06].
- Texto: Puede ser creada a mano o producto de un proceso de auto-anotación como hemos comentado con anterioridad; este tipo de características suele producir clasificación en categorías y subcategorías, y no podemos dejar de tomarlo en cuenta ya que es una de las características más usadas en la actualidad.

Cada uno de los tipos de características es especialmente útil para un tipo

de tarea distinto; por ello es necesario analizar el sistema y sus necesidades a la hora de elegir las características que vamos a utilizar.

2.2. Representaciones usadas

Dado que lo más común es segmentar la imagen, hallar los puntos de interés, o incluso calcular las características para una serie de puntos definidos en una rejilla (como veremos en 3.1); el resultado de la extracción de características será una serie de vectores, cada uno de los cuales representa una posición concreta de la imagen. Este conjunto de vectores es la denominada firma de la imagen. Para tratar con dichos vectores existen diversas posibilidades. La más simple sería comparar directamente las regiones que representa cada uno de los vectores de las dos imágenes. También se podría considerar el vector final que represente a la imagen como una combinación lineal de los vectores presentes en la imagen.

Una de las soluciones más utilizadas en los últimos años es el método denominado "Bag of words" [USS09, MM09, KKLK09]. Este método representa a la imagen como un histograma de los diferentes vectores de características presentes en la imagen. Para ello asigna un grupo a cada uno de los vectores de características y a partir del número de grupo calcula el histograma como vemos en la imagen 2.2.

Para poder generar los distintos grupos partimos de una serie de imágenes denominadas conjunto de entrenamiento. A partir de los vectores presentes en dichas imágenes generaremos los grupos de manera que representen conjuntos aislados con significado propio.

Existen diversos métodos de cuantificación vectorial para calcular los distintos grupos; el más conocido es el algoritmo k-means que explicaremos en 3.2.1. También existen otros métodos como el "random forest" utilizado en [USS09].

2.3. Cálculo de la semejanza y clasificación

Una vez hemos construido la firma de una imagen, es decir el vector de valores que representa a la imagen, el siguiente paso es el cálculo de la semejanza. Se han propuesto numerosas soluciones en los últimos años, algunos de los factores fundamentales a la hora de diseñar un cálculo de la semejanza se pueden resumir como sigue:

- Correspondencia con la semántica.

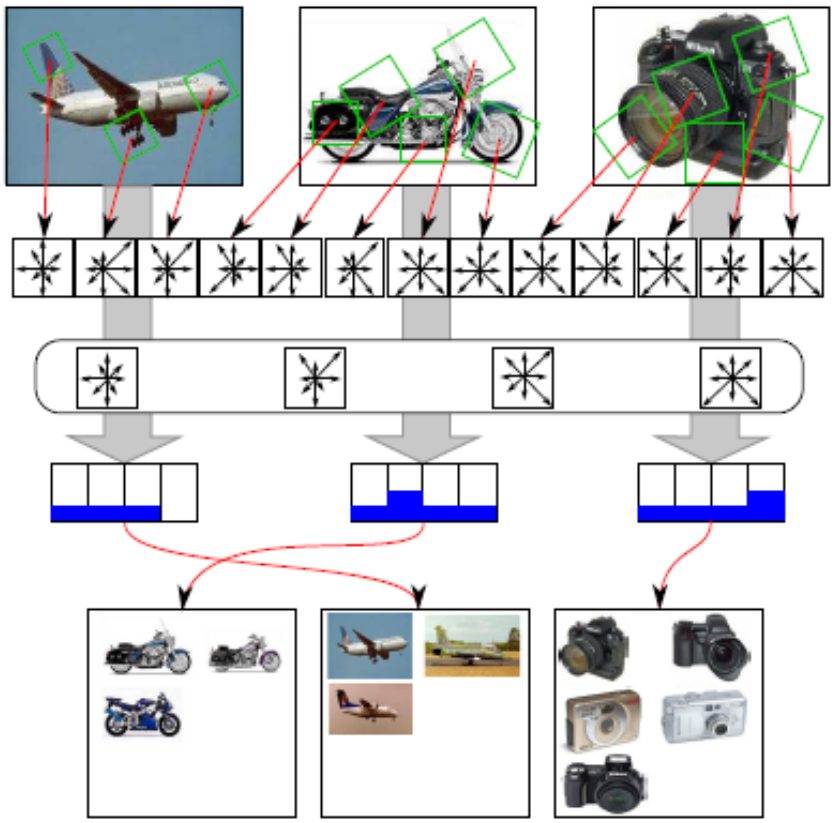

Figura 1: Esquema de un sistema " Bag of Words"

- Robustez frente al ruido e invariancia a las perturbaciones.
- Eficiencia computacional; que incluiría escalabilidad y posibilidad de funcionar en tiempo real.
- Invariancia frente al fondo de la imagen, permitiendo así las consultas basadas en región.
- Linealidad local; por ejemplo siguiendo la desigualdad triangular en la vecindad. De forma que no se produzcan discontinuidades: cortes, saltos, asíntotas verticales.

Además, las distintas técnicas se pueden agrupar de acuerdo con sus filosofías de diseño en la siguiente clasificación:

- Según la forma en que tratan las características, como vectores, no vectores...
- Según si utilizan semejanza global, local, o una mezcla de ambas.
- Según si el cálculo se realiza sobre un espacio lineal o no lineal.
- Dependiendo del rol que juegan los segmentos de la imagen en el cálculo de la semejanza.
- Si utiliza medidas de semejanza estocásticas, difusas o deterministas.
- Dependiendo de si utiliza aprendizaje supervisado, semi-supervisado o no supervisado.

Como hemos comentado con anterioridad, el cálculo de la semejanza depende en gran medida del tipo de firma que hemos construido para la imagen. Aunque en general, para cada tipo de sistema hay que utilizar el cálculo de semejanza que más le convenga, a continuación se resumen algunos de los usados.

Ejemplos de distancias serían la distancia euclídea, la distancia euclídea ponderada, la distancia Hausdorff, etc. La distancia euclídea sería la más simple de todas y se calcularía como la distancia entre los vectores de características; siendo la distancia euclídea ponderada una variación de esta, que se construye añadiendo un vector de pesos que pondera la influencia de cada región de la imagen a la hora de calcular la distancia. La distancia Hausdorff trata de encontrar para cada vector de características de la imagen I_1 el vector más semejante de la imagen I_2 , y la distancia se calcularía como la

Distancia	Entrada	Cálculo	Complejidad	Métrica	Comentarios
Euclídea	$\vec{X}_a, \vec{X}_b \in \mathbb{R}^n$ (vectores)	$\vec{X}_a \cdot \vec{X}_b$	$\Theta(n)$	Si	Popular, rápida, L1 también usada
Euclídea con pesos	$\vec{X}_a, \vec{X}_b \in \mathbb{R}^n$ $W \in \mathbb{R}^n$ (vector más pesos)	$\vec{X}_a^T[W]\vec{X}_b$ [.] $\leftarrow$ diagonalizar	$\Theta(n)$	Si	Permite que los vectores sean ponderados
Hausdorff	Conjuntos de vectores $\{\vec{X}_a^{(1)}, \ldots, \vec{X}_a^{(p)}\}$ $\{\vec{X}_b^{(1)}, \ldots, \vec{X}_b^{(q)}\}$	Ver ecuación 1	$\Theta(pqn)$ ($d(\cdot,\cdot) \leftarrow L^2$ norm)	Si	
Mallows	Conjuntos de vectores $\{\vec{X}_a^{(1)}, \ldots, \vec{X}_a^{(p)}\}$ $\{\vec{X}_b^{(1)}, \ldots, \vec{X}_b^{(q)}\}$ Signific.:S	ver ecuación 2	$\Theta(pqn)$+parte variable	Si	El EMD es un caso especial
IRM	Conjuntos de vectores $\{\vec{X}_a^{(1)}, \ldots, \vec{X}_a^{(p)}\}$ $\{\vec{X}_b^{(1)}, \ldots, \vec{X}_b^{(q)}\}$ Signific.:S	ver ecuación 2	$\Theta(pqn)$+parte variable	No	Mucho más rápido que el Mallows en la práctica
Divergencia K-L	$\vec{F}, \vec{F} \in \mathbb{R}^m$ (Histogramas)	$\sum xF(x)log\frac{F(x)}{G(x)}$	$\Theta(m)$	No	Asimétrico, compara distribuciones

Cuadro 1: Ejemplo de cálculos de distancia

máxima distancia de entre todos los pares de vectores. A la hora de informatizar este cálculo se simetriza invirtiendo el papel de las imágenes y escogiendo la distancia mayor; cómo podemos ver en la ecuación 1.

$$D_H(I_1, I_2) = \max\left(\max_i \min_j d(z_i^{(1)}, z_j^{(2)}), \max_j \min_i d(z_j^{(2)}, z_i^{(1)})\right) \tag{1}$$

$$D(I_1, I_2) = \min_{s_{i,j}} \sum_{i=1}^{n_1} \sum_{j=1}^{n_2} s_{i,j} d(z_i^{(1)}, z_j^{(2)}) \tag{2}$$

Este cálculo de semejanza puede utilizarse para realizar la clasificación de una imagen. También existen otras formas de clasificación que han sido utilizadas en otros

ámbitos del reconocimiento de formas como serían: las redes neuronales, las máquinas de soporte vectorial, mixturas de gaussianas,...

En el cuadro 1 se pueden observar distintos cálculos de distancia entre vectores. Para la complejidad consideramos que p es el número de vectores en uno de los conjunto, q es el número de vectores en el otro conjunto, y n es el número de elementos en cada vector.

Estos sistemas de clasificación tomarían los vectores de características, junto a la clase a la que pertenecen, para entrenar los modelos que luego servirán para clasificar las nuevas imágenes.

3. Metodología propuesta

Para la realización de la tesina se adopto el esquema de representación denominado "Bag of words". A continuación se explica; primero los algoritmos para extraer las características de dichas imágenes. Después los métodos de cuantificación vectorial usados para generar las palabras propias del sistema y el método usado para generar el histograma. Por último se explicara el método de clasificación utilizado.

3.1. Características utilizadas

Las imágenes de la base de datos no se pueden usar para el reconocimiento ya que para el ordenador no son más que matrices en las que, por cada punto, se indican las tres componentes de color. Por ello, de cada imagen, se obtiene una serie de vectores de características que representan a dichas imágenes. A continuación se detallan los distintos algoritmos utilizados para la obtención de los vectores de características, explicando primero los algoritmos básicos con los que se realizaron las pruebas para después ahondar en las combinaciones de dichos algoritmos. Con la combinación de ambos algoritmos lo que se pretende es intentar obtener unos vectores de características capaces de representar aquellas partes de la imagen que quedan fuera de alguno de los dos algoritmos básicos.

3.1.1. Sift Denso

Uno de los tipos de características obtenidos fueron las de tipo SIFT (Scale Invariant Feature Transform) [Low04]. Estas características han sido ampliamente utilizadas en tareas de reconocimiento de imágenes como en: [USS09] [BMOM] y [SGSL05]. En nuestro caso utilizamos una versión densa del algoritmo SIFT. Indicar que en lo que se refiere a SIFT hay dos partes claramente diferenciadas, por un lado está la detección de puntos de interés y por otro los vectores de características que se extraen de las posiciones seleccionadas por el proceso anterior.

- **Algoritmo SIFT.**
  El primer paso en el algoritmo SIFT es la obtención del espacio de escalas. Dicho espacio se forma convolucionando la imagen original con gaussianas y almacenándolo en la denominada pirámide de gaussianas. Los distintos niveles, compuestos por imágenes emborronadas, se restan por pares la pirámide de diferencias de gaussianas

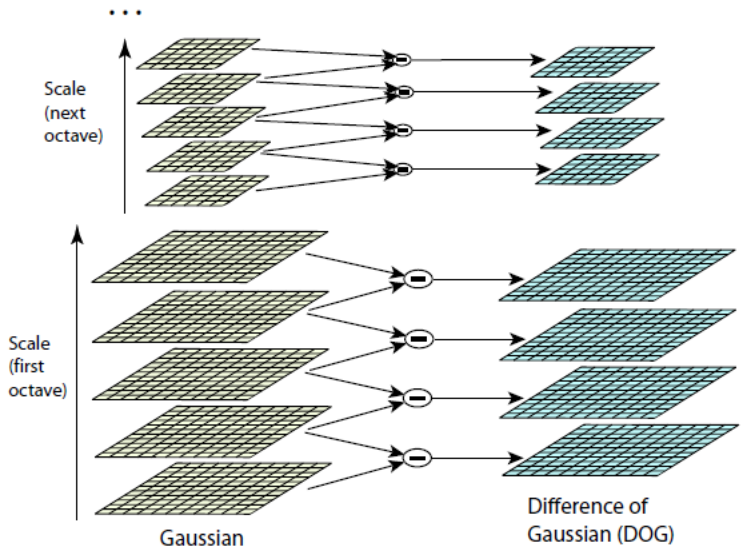

Figura 2: Pirámide de Gaussianas y DOG

de la siguiente manera:

$$D(x, y, \sigma) = (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) = L(x, y, k\sigma) - L(x, y, \sigma) \qquad (3)$$

El proceso se muestra en la imagen 3.1.1

Luego se obtienen los extremos locales. Para ello se cada punto se compara con sus ocho vecinos en la imagen actual de la pila de diferencia de gaussianas y con los nueve vecinos en de las escalas superior e inferior. Si el punto es mayor o menor que todos sus vecinos entonces se elije como extremo local.

A estos puntos se les realiza un ajuste detallado con respecto a los datos cercanos para la localización, la escala y el ratio de las curvaturas principales. De esta forma se pueden rechazar los puntos sensibles al ruido y los que están localizados a lo largo de un borde.

Primero se ajusta una función cuadrática a los puntos locales para determinar la localización interpolada del máximo. Para ello se usa la expansión de Taylor:

$$D(x) = D + \frac{\partial D^T}{\partial x} + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \qquad (4)$$

Donde D y sus derivadas son evaluadas en el punto y x= $(x, y, \sigma)^T$ es el desplazamiento con respecto a este punto. La localización del extremo $\hat{x}$ se determina toman-

do la derivada de la función con respecto a x e igualando a cero:

$$\hat{x} = \frac{\partial^2 D^{-1}}{\partial x^2} \frac{\partial D}{\partial x} \tag{5}$$

Si el desplazamiento es mayor que 0,5 en cualquier dimensión eso significa que el extremum está más cerca de otro el punto se cambia. El valor de esta función es útil para rechazar los extremos inestables con bajo contraste al sustituir la ecuación (5) en (4):

$$D(\hat{x}) = D + \frac{1}{2} \frac{\partial D^T}{\partial x} \hat{x} \tag{6}$$

Los valores menores de 0.03 son descartados. También se eliminan las respuestas de las aristas. Para ello se utiliza las curvaturas principales, ya que, una arista tendrá una curvatura muy grande a lo largo de la arista y una pequeña en la dirección perpendicular. Las curvaturas principales se calculan con una matriz Hessiana en el punto y la escala del punto clave.

$$H = \begin{vmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{vmatrix} \tag{7}$$

Podemos evitar calcular los eigenvalores ya que solo nos interesa su ratio. Dejemos que α sea el eigenvalor con la mayor magnitud, y β el de menor. Podemos calcular la suma de los eigenvalores a partir de la traza H y su producto a partir del determinante:

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta \tag{8}$$

$$Det(H) = D_{xx} D_{yy} - (D_{xy})^2 = \alpha\beta \tag{9}$$

Si r es el ratio entre las dos magnitudes entonces $\alpha = r\beta$:

$$\frac{Tr(H^2)}{Det(H)} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r+1)^2}{r} \tag{10}$$

Esta función depende solo del ratio. Ahora solo tenemos que comprobar que:

$$\frac{Tr(H^2)}{Det(H)} < \frac{(r+1)^2}{r} \tag{11}$$

Está por debajo de un umbral r. Luego se calcula la orientación y el gradiente de cada punto de la siguiente forma:

$$m(x,y) = \sqrt{(L(x+1,y) - L(x-1,y))^2 + (L(x,y+1) - L(x,y-1))^2} \tag{12}$$

$$\theta(x,y) = tan^{-1}((L(x,y+1) - L(x,y-1))/((L(x+1,y) - L(x-1,y))) \quad (13)$$

Aquellos puntos en los que haya más de una dirección dominante se extraen otros puntos con las siguientes direcciones dominantes. De estos puntos será de los que se obtengan los descriptores locales de la imagen. Para ello se toman los dieciséis vecinos del punto con sus gradientes y sus magnitudes y se crea un histograma por cuadrantes como se puede ver en la imagen.

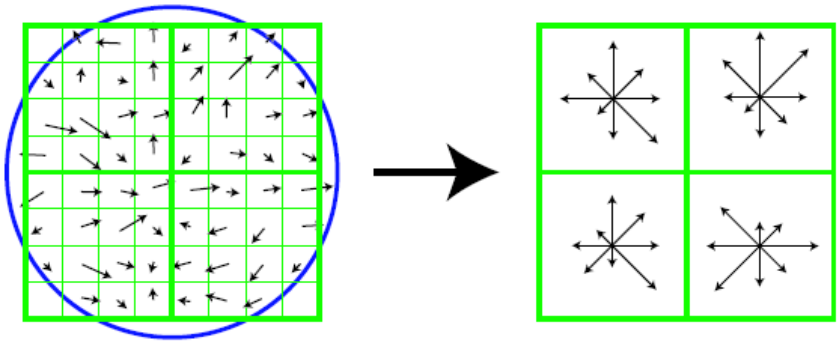

Figura 3: Cálculo de las características SIFT

Los vecinos se dividen en cuadrantes y en ocho posibles direcciones obteniéndose un vector de características de $4x4x8 = 128$ elementos. Estos vectores se normalizan para paliar los efectos de la iluminación; los valores obtenidos de la normalización mayores que 0.2 se igualan a 0.2 para reducir el efecto de las grandes magnitudes de gradientes para reducir el efecto de los cambios de iluminación no lineares. Luego se vuelven a normalizar.

- **SIFT denso.**

  Para obtener las características SIFT de manera densa se sustituyo la extracción de los puntos clave por un algoritmo que obtiene puntos igualmente espaciados entre si y calcula los vectores de características en la localización de dichos puntos. Para cada uno de los puntos de la rejilla se calculaba en que escala del espacio de escalas era máximo y en cual mínimo. Después se elegía de entre ambos aquel que estaba más alejado de la media y está escala se usaba para calcular el vector de características. De esta forma se pretendía obtener invariancia con respecto a la escala.

  Como podemos observar en la imagen 4 los puntos sobre los que se extraerían los vectores de características serían las intersecciones entre las líneas verticales

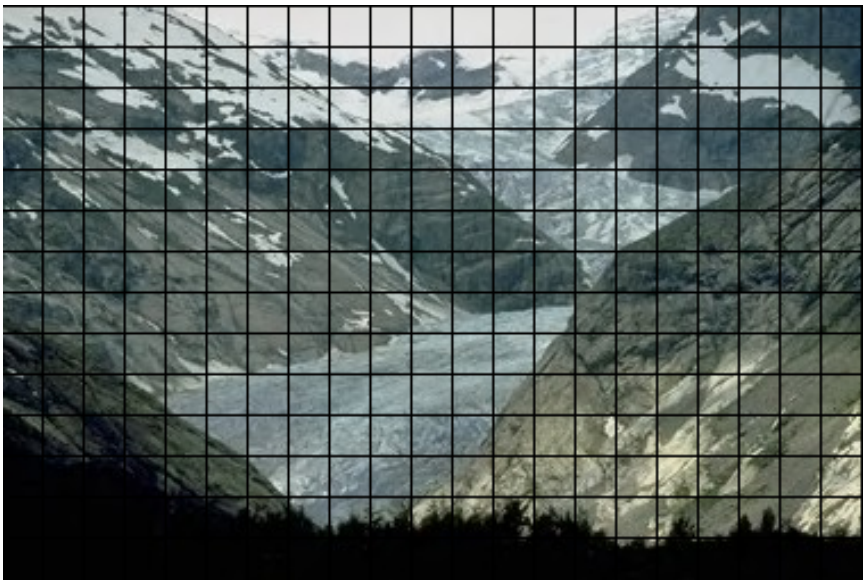

Figura 4: Obtención de los puntos de interés de manera densa

y horizontales; produciendo así una rejilla de puntos de los que se extraerían las características.

Por otra parte el algoritmo de [Low04] trata sólo con imágenes en escalas de grises, para adaptarlo a imágenes de color se calculó para cada punto y para cada capa de color la escala correspondiente. De esta forma la escala de cada color puede ser distinta para cada color en un mismo punto.
Una vez obtenida la escala se calculaba en cada punto, para cada color, el vector de características. El vector final se compone de la concatenación de los vectores de características de los tres colores; por lo que el tamaño del vector final es 4x4x8x3 = 384

3.1.2. Histograma de color Denso

Otra de las características más ampliamente usadas son los histogramas de color. Estas características son fácilmente extraíbles y obtienen resultados aceptables [HM02, SC95, JFS95]

- **Histograma de color.**
  En nuestro caso el histograma de color se obtiene separando cada espacio de color en 8 valores, de esta forma tenemos 8x8x8=512 posibles valores de color distintos.

Luego solo hay que recorrer la imagen y sumar uno a la posición del vector de 512 valores que corresponde al color del punto en el que nos encontramos en ese momento.

- **Histograma de color denso.**
  Se calculó el histograma de color denso utilizando el mismo método que en el apartado 3.1.1. Es decir, se obtuvo el espacio de escala de la imagen, luego la pirámide de diferencias de gaussianas y a partir de esta, y utilizando una rejilla, se obtuvieron los puntos donde se calcularían los vectores de características.

  Los vectores de características se obtuvieron calculando el histograma de color para una región de 16x16 en torno al punto escogido.

3.2. Cuantificación vectorial

Una vez tenemos los vectores de características se realiza un proceso de cuantificación vectorial. Mediante este proceso convertimos una imagen, representada por un conjunto de vectores de características, en un solo vector que contiene un histograma que representa a la imagen y que será usado para realizar comparaciones entre las imágenes.

Para hallar el grupo al que pertenece cada uno de los vectores de características de la imagen existen varios procesos, dos de los cuales se describen a continuación. Una vez tenemos todos los grupos a los que pertenece solo tenemos que calcular el histograma que representara a la imagen.

3.2.1. K-means

El algoritmo k-medias intenta hallar una serie de grupos, con su valor central, de manera que cada grupo represente una porción significativa del conjunto de puntos con el que se entrene.

Se trata de un proceso iterativo en el que se parte de un número de vectores aleatorios de posibles centros que se va refinando en cada iteración. El número inicial de grupos tiene que ser fijado a mano previamente, en nuestro caso se fijaron 2000 grupos.

Con cada iteración el valor de los centros se actualiza en dos pasos. Un primer paso calcula a que centro pertenecen cada uno de los puntos del conjunto de entrenamiento de acuerdo a la siguiente formula, que asigna a cada punto el centro más cercano.

$$S_i^{(t)} = \{x_j : ||x_j - m_i(t)|| \leq ||x_j - m_{i^*}^{(t)}|| \ \forall i^* = 1, \ldots, k\} \tag{14}$$

Una vez tenemos los puntos que corresponden a cada centro, actualizamos el valor de los centros de acuerdo a la siguiente fórmula:

$$m_i^{(t+1)} = \frac{1}{|S_i^t|} \sum_{x_j \in S_i^{(t)}} X_j \tag{15}$$

El proceso se repite hasta que los puntos asignados a cada centro no cambian, o hasta que se cumple un umbral de convergencia o hasta un número determinado de iteraciones.

Ya que el resultado obtenido depende en gran medida de los puntos iniciales que son aleatorios se suele realizar el proceso entero con distintos valores quedándonos con el resultado que obtiene mejor índice de compacidad.

Indicar que se puede considerar el algoritmo k-means como un caso concreto del algoritmo E-M generalizado ya que; el paso de asignación es el paso de estimación y el paso de actualización es el paso de maximización.

3.2.2. Proyecciones aleatorias

Otra de las formas de realizar la cuantificación vectorial es utilizando la técnica de proyecciones aleatorias. En este algoritmo se parte de un número de vectores tal que dicho número n sea $2^n = c$. Donde c es el número de clases deseadas. Cada uno de dichos vectores tiene el mismo tamaño que los vectores de características de la imagen. Y está formado por valores reales comprendidos entre el -1 y el 1.

Para hallar a qué grupo pertenece cada uno de los vectores, y así calcular el histograma, se formará un nuevo vector, denominado B a partir de la colección de proyecciones aleatorias A y el vector de características C para el que estemos calculando el grupo. Indicar que $|B| = n$ y $A(x)$ representa al vector en la posición x. El vector se formara a partir de la siguiente fórmula:

$$B(x) = \begin{cases} 1 \text{ si } A(x)C \geq 0 \\ 0 \text{ en otro caso} \end{cases} \tag{16}$$

De esta forma tenemos que, para cada vector de características de la imagen, se producen n valores binarios, donde n es el número de vectores del codebook. Si consideramos los n valores binarios como la representación binaria de un solo número, tenemos

un valor comprendido entre 0 y 2^n que será el grupo al que pertenece dicho vector de características.

Luego solo nos queda calcular el histograma de todos los valores para hallar el histograma que representa a la imagen.

3.3. Combinación de vectores de características

También se probó a realizar una combinación de los vectores de características obtenidos por los dos métodos anteriores, es decir, mediante SIFT denso e Histograma de color denso. Se utilizaron tres métodos de combinación que se detallan a continuación, y con ellos se pretende aprovechar las ventajas de ambos métodos.

- **Concatenación de los vectores de características.**
  Para esta combinación se concatenaron los vectores de características; para cada punto el nuevo vector de características estaba compuesto por el vector producido por el SIFT denso, y a continuación el producido por el Histograma de color denso; este método producía vectores de una longitud de $384 + 512 = 896$ valores.

  Una vez concatenados los vectores estos se almacenaban en nuevos ficheros de características a los que eran tratados de la misma forma que los obtenidos por los dos métodos sin mezclar.

- **Concatenación de los vectores codificados.**
  Otra de las formas de combinar es concatenar los vectores producidos por la cuantificación vectorial. De esta forma, una vez hemos aplicado el algoritmo de cuantificación vectorial a los ficheros que contienen todos los vectores de características de una imagen y hemos obtenido un solo vector que la representa, concatenamos los vectores producidos por el SIFT Denso y por el histograma de color denos para producir el nuevo fichero.

- **Combinación usando correlación.**
  La última forma de combinación que se probó fue utilizando correlaciones. Para cada punto se obtenía el número del grupo al que pertenecía tanto en SIFT como en histograma de color. Una vez disponemos de ambos números; se calcula la correlación como la probabilidad de que: dado un valor de grupo para SIFT se obtuviese un valor de grupo para histograma de color.

Para ello se analizaban para cada punto y se obtenía su grupo con los dos algoritmos, lo que se almacenaba era la cuenta del número de veces que se habían producido dos números de grupo en el mismo punto.

3.3.1. Correlación espacial

Por último se realizaron pruebas utilizando la correlación espacial. Mediante este método se pretendía capturar la relación espacial entre las posiciones de los diferentes vectores de características. Para ello se almacenaba la probabilidad de que un valor de cuantificación vectorial dado fuese seguido en las posiciones derecha, abajo y diagonal, por otro valor dado; como se ve en la imagen 5.

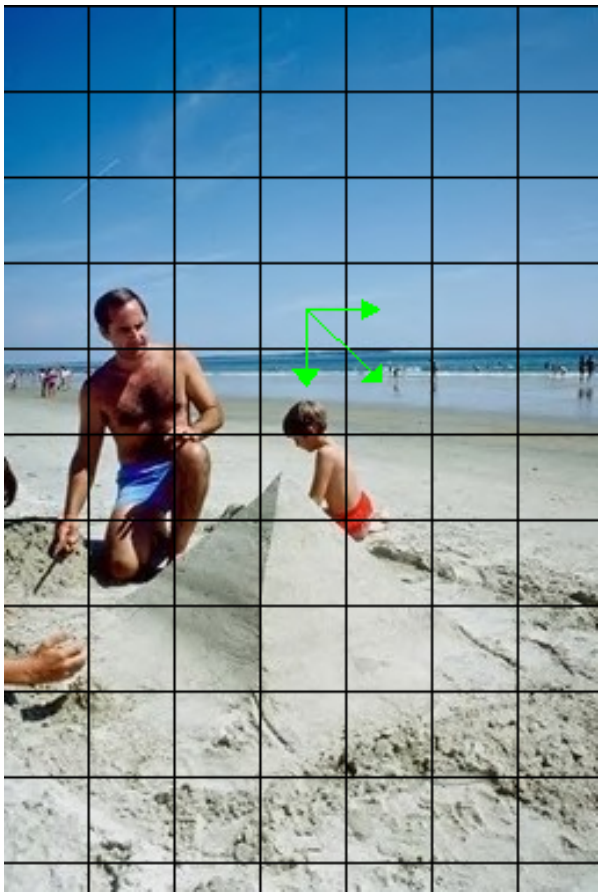

Figura 5: Cálculo de la correlación espacial

De esta forma se conseguía representar la relación espacial. El resultado de este proceso es un fichero donde se almacenan el número de ocurrencias de cada par de

grupos. Indicar que los valores están normalizados antes de almacenarlos con lo que se encuentran en un rango de 0 a 1.

3.4. Clasificación

Una vez tenemos ya un método para generar los vectores de características que definirán una imagen podemos usar dichos vectores para realizar un sistema de clasificación. Para realizar la clasificación partimos de un conjunto de imágenes distribuidas en grupos, en nuestro caso se muestran en 4.2; dichas imágenes con sus respectivos grupos servirán para clasificar imágenes nuevas que llegan al sistema.

En la presente tesina se implemento el sistema de clasificación conocido como k-vecinos. En este tipo de aproximaciones la clase de la imagen nueva viene determinada por la clase de las imágenes más próximas a la imagen a clasificar de acuerdo con un tipo de distancia. En nuestro caso se probaron tres tipos distintos de cálculo de distancia, o semejanza, entre imágenes. Con esto pretendíamos encontrar aquella distancia que ofreciese una menor tasa de error de clasificación:

- Distancia euclídea.
  La distancia entre dos vectores se calcula para dos puntos $P = (p_1, p_2, \ldots, p_n)$ y $Q = (q_1, q_2, \ldots, q_n)$ como:
  $$D(P, Q) = \sqrt{(p_1 - q_1)^2 + (p_2 - q_2)^2 + \ldots + (p_n - q_n)^2} \tag{17}$$

- Divergencia Kullback Leibler.
  En este caso se tratan los vectores como conjuntos de probabilidad, están normalizados, y la distancia, en este caso divergencia, se calcula como:
  $$D_{KL}(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \tag{18}$$
  En el caso en el que $Q(i)$ sea 0 se realiza un suavizado asignando a $Q(i)$ el valor 0,00001.

- Distancia Jensen Shannon.
  La divergencia Jensen Shannon se calcula a partir de la divergencia Kullback Leibler como:
  $$JSD(P||Q) = \frac{1}{2} D(P||M) + \frac{1}{2} D(Q||M) \tag{19}$$
  donde $M = \frac{1}{2}(P + Q)$ y $D(Q||M)$ es la divergencia kullback Leibler.

Una vez tenemos la forma de calcular la distancia entre dos imágenes, para hallar a qué grupo pertenece una imagen solo hay que calcular la distancia de dicha imagen con respecto a todas las demás y ordenarlas; de manera que primero estén las más semejantes a la imagen consulta. La clase de la imagen será aquella que se encuentre un mayor número de veces entre las k primeras imágenes recuperadas por el sistema. En caso de igual número de ocurrencias de dos o más clases nos quedaremos con aquella que haya aparecido antes en la ordenación.

4. Resultados

4.1. Evaluación de los resultados

Para evaluar las distintas pruebas realizadas se obtuvo el Classiffication Error Rate (CER), es decir: el porcentaje de fallos de clasificación del corpus de test. El CER se calcula como:

$$CER = \frac{\text{Número de imágenes mal clasificadas}}{\text{Número de imágenes}} \tag{20}$$

El corpus de test está formado por 100 imágenes de cada una de las bases de datos. Dicho corpus es escogido aleatoriamente intentando que las imágenes se distribuyan de manera equiprobable entre todas las clases existentes en la base de datos.

4.2. Bases de datos

Para realizar las pruebas de los algoritmos de reconocimiento de imágenes se usaron dos bases de datos de imágenes que han sido ampliamente utilizadas en la literatura. A continuación se explican ambas, definiendo los grupos existentes en las dos. La razón por la que se utilizan dos bases de datos es intentar evitar el sesgo existente al tratar solo con una base de datos, ya que los algoritmos usados pueden funcionar muy bien para una base de datos y para otra no debido a la composición de los grupos o al contenido de las imágenes.

4.2.1. Corel

La base de datos Corel está compuesta por mil imágenes distribuidas en 10 grupos de cien imágenes cada uno. Estos diez grupos contemplan un amplio rango de categorías diferentes. Esta bases de datos ha sido utilizada en [FPZ03] [CFB04] y [BDG+03].

Pueblos de África. Playas. Monumentos. Autobuses. Dinosaurios. Elefantes. Flores. Caballos. Alpinismo. Comidas

4.2.2. Msrcorid

La base de datos Microsoft Research Cambridge Object Recognition Image Database (Msrcorid) [Dat] está compuesta por 4500 imágenes distribuidas de manera desigual entre las treinta y tres categorías que se muestran a continuación. Esta base de datos ha sido utilizada en [MM09] y [EZWVG06].

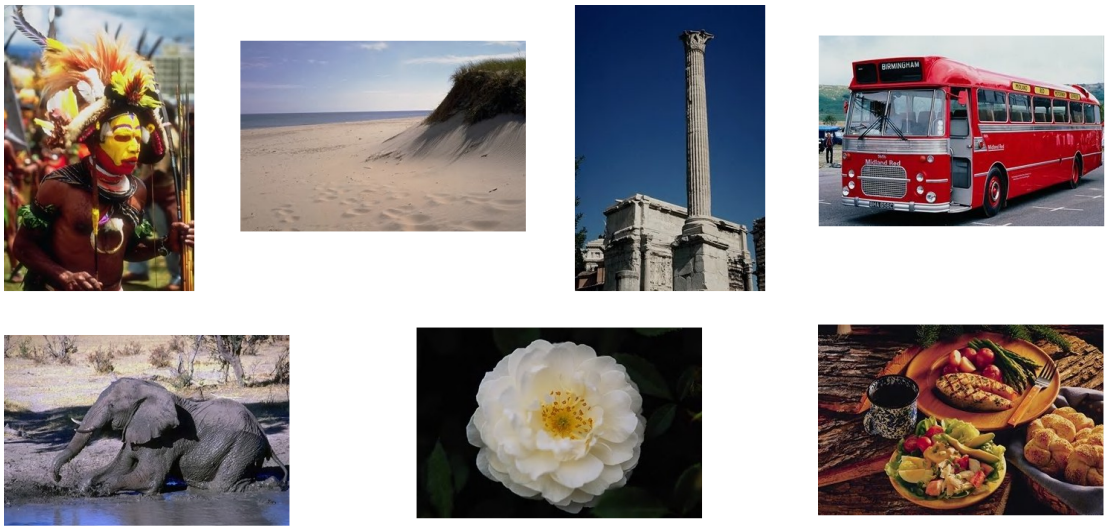

Figura 6: Ejemplo de imágenes en la base de datos Corel

Aeroplanos en general. Aeroplanos solos. Bancos y sillas. Bicicletas en general. Bicicletas vistas desde el lateral. Pájaros en general. Pájaros solos. Edificios . Coches en general. Coches vistos desde delante. Coches vistos desde un lado. Coches vistos desde detrás. Chimeneas. Nubes. Vacas en general. Vacas solas. Puertas. Flores en general. Flores solas. Tenedores. Cuchillos. Hojas. Miscelánea. Escenas de campo. Escenas de oficina. Escenas de ciudad. Ovejas solas. Ovejas en general. Signos. Cucharas. Arboles en general. Arboles solos. Ventanas

4.3. Características utilizadas

El primer paso realizado se realizo utilizando las características SIFT y de histograma de color. Para ambas se utilizo su variante densa tal y como se explica en los apartados 3.1.1 y 3.1.2.

A continuación se detallan los resultados obtenidos para las dos características en función del cálculo de la distancia empleado y del tamaño de rejilla aplicado en las características densas.

4.3.1. Función de distancia

Como hemos comentado en el apartado de metodología se realizaron pruebas utilizando los diversos algoritmos de cálculo de distancias. En la tabla 2 se pueden obser-

Figura 7: Ejemplo de imágenes en la base de datos Msrcorid

var los resultados de CER dependiendo del cálculo de distancia utilizado y del algoritmo de generación de características utilizado.

Algoritmo / Distancia	Euclidean	Kullback Leibler	Jensen Shannon
SIFT denso	56 %	49 %	47 %
Histograma denso	25 %	18 %	14 %

Algoritmo / Distancia	Euclidean	Kullback Leibler	Jensen Shannon
SIFT denso	53 %	48 %	47 %
Histograma denso	44 %	39 %	36 %

Cuadro 2: CER En función del cálculo de distancia para las bases de datos Corel y Microsoft

Como se puede observar el cálculo de distancia Jensen Shanon es el que mejores resultados aporta. Por otra parte también observamos que el histograma de color denso es claramente superior al método SIFT denso y que la tarea de clasificación de la base de datos Microsoft es más compleja que el caso de la base de datos Corel.

4.3.2. Tamaño de rejilla

La distancia existente entre los diversos puntos tomados para calcular las características puede influir de una manera decisiva en los resultados obtenidos ya que si tomamos un tamaño de rejilla demasiado amplio podemos perder parte de la información presente en la imagen. Por otro lado, un tamaño excesivamente pequeño puede suponer un exceso de solapamientos en las zonas de los puntos que conduciría a una redundancia en la información extraída de la imagen. Por ello se realizaron pruebas dependientes del tamaño de la rejilla, de la base de datos y del algoritmo utilizado para averiguar que tamaño es el óptimo, los resultados se pueden observar en la grafica 8.

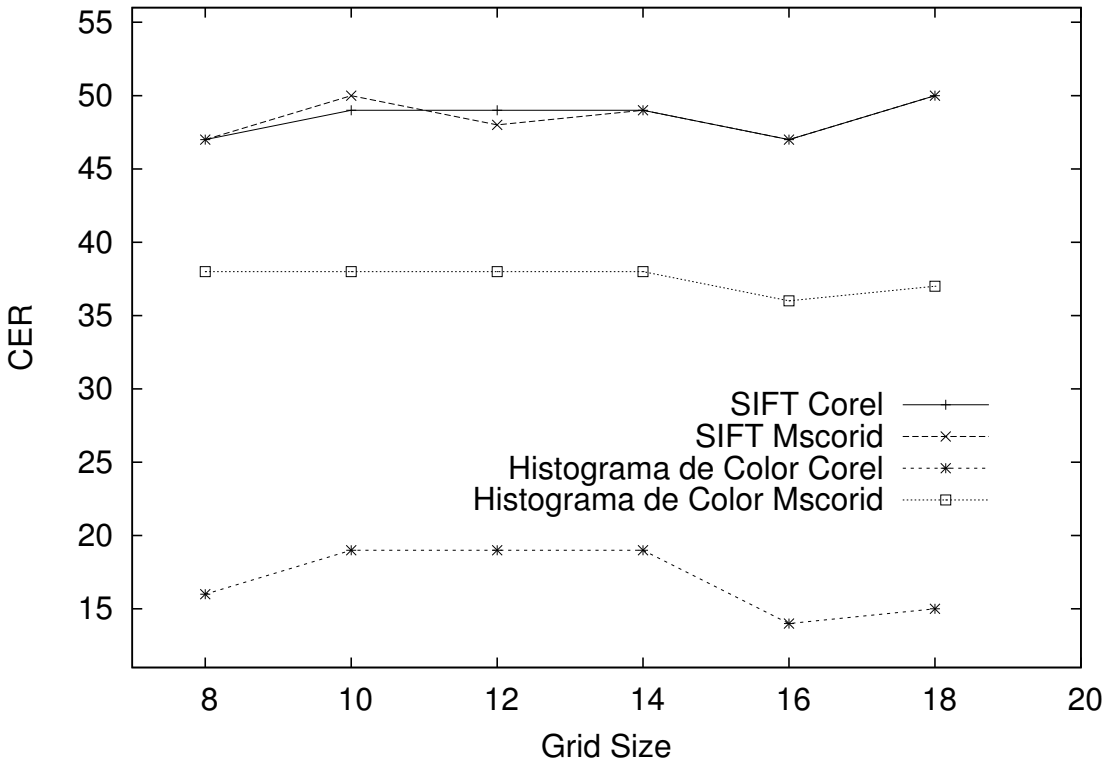

Figura 8: CER en función del tamaño de rejilla

El tamaño óptimo de rejilla se encuentra en dieciséis puntos de separación entre puntos de la rejilla. Como se puede observar a partir de esa distancia el número de errores vuelve a crecer. También se puede apreciar como las características basadas en histograma de color obtienen resultados superiores a los obtenidos por las características SIFT y como la base de datos Msrcorid es más compleja de clasificar que la base de datos

Corel. Vemos también que dicha diferencia de complejidad entre las dos bases de datos es menos patente al utilizar las características SIFT que las basadas en histograma de color.

4.4. Método de cuantificación vectorial

Existen diversos factores que influyen en gran medida en los resultados obtenidos por los diversos algoritmos de clasificación. A continuación vamos a analizar la influencia del algoritmo de cuantificación vectorial.

Como hemos comentado anteriormente el algoritmo K-means es uno de los más utilizados a la hora de generar cuantificaciones vectoriales y ofrece unos buenos resultados en muchos de los ámbitos. Por otra parte el algoritmo de cuantificación vectorial basado en proyecciones aleatorias no ofrece unos resultados tan buenos pero a cambio es un algoritmo extremadamente rápido, lo que cuando tratamos con bases de datos con un gran número de imágenes es un factor que no puede ser obviado.

El coste temporal del algoritmo K-means es de: $O(x^{dk+1} \log x)$ donde x es el número de vectores a cuantificar, k es el número de clases y d es la longitud del vector. El coste computacional del algoritmo de proyecciones aleatorias es de: $O(dnx)$ donde d es el tamaño del vector de características, n es el número de proyecciones aleatorias y x el número de vectores a cuantificar.

A continuación, en la tabla 3 se muestran los resultados para ambos métodos para las dos bases de datos fijando el cálculo de la distancia al método Jensen-Shannon por ser el que mejores resultados ofrece:

Método de cuantificación vectorial / Base de datos	Corel	Msrcorid
K-medias-SIFT	45 %	46 %
Proyecciones aleatorias -SIFT	47 %	47 %
K-medias-Histograma de color	13 %	34 %
Proyecciones aleatorias -Histograma de color	14 %	36 %

Cuadro 3: K-medias frente a Proyecciones aleatorias

Como podemos observar el método de k-means mejora los resultados aproximadamente entre un 2 % y un 7 %. Aun así, al tratarse de un método mucho más costoso temporalmente optamos por utilizar el método de cuantificación vectorial de proyecciones aleatorias .

4.5. Escala fija frente a Escala detectada

Otro de los aspectos relevantes es la escala a la que se obtienen los vectores de características en cada uno de los puntos de la rejilla. En esta tesina se plantean dos métodos de obtención de los vectores de características.

Por un lado es posible fijar el tamaño de la escala con lo que los espacios sobre los que se calculan los vectores de características serán iguales para todos los puntos de la imagen.

Por otro lado es posible calcular en cada uno de los puntos de dicha rejilla que escala es la más apropiada mediante el método explicado en el apartado 3.1.1. De esta forma se consigue que el tamaño del espacio sobre el que se calculan los vectores pueda variar de un punto a otro, captando de esta forma las partes homogéneas y las partes con detalles pequeños.

En la siguiente tabla, 4 podemos observar los resultados obtenidos para ambos métodos fijando el cálculo de la distancia como Jensen-Shannon y el algoritmo de cuantificación vectorial de proyecciones aleatorias .

Tipo de escala / Base de datos	Corel	Msrcorid
Escala fija-SIFT	54 %	58 %
Escala detectada-SIFT	47 %	47 %
Escala fija-Histograma de color	26 %	45 %
Escala detectada-Histograma de color	14 %	36 %

Cuadro 4: Escala fija frente a Escala detectable

Como podemos observar la escala detectable obtiene resultados superiores a los obtenidos por la escala fija. Esto revela que la importancia de captar la diferencia entre los espacios homogéneos y las zonas con alta presencia de detalles.

4.6. Tamaño de rejilla dependiente de la escala

Otra posibilidad para avanzar un paso más en la diferenciación entre las zonas homogéneas como podría ser el cielo o el mar y las zonas más detalladas como las personas es la variación del tamaño de rejilla en función de la escala en un punto dado. Para este método el siguiente punto en la rejilla se situaría en los bordes del recuadro formado por la elección de la escala anterior, como se observa en la imagen 9.

Este método de extracción es extremadamente costoso, y aunque ofrece unos

Figura 9: Tamaño de rejilla dependiente de la escala

resultados superiores a las anteriores solo se han realizado pruebas usando la cuantificación vectorial de proyecciones aleatorias y la distancia Jensen-Shannon. Los resultados se muestran en la tabla 5.

4.7. Combinación de características

Como hemos observado en los apartados anteriores las características basadas en histograma de color son superiores a las características SIFT. Aun así el lógico plantearse algún método que consiga reunir las ventajas de ambas características.

Para ello existen distintas posibles formas de unión como pueden ser la concatenación de los vectores de características, la concatenación de los vectores codificados o la correlación entre los dos tipos de características. A continuación discutiremos todos estos métodos.

Característica / Base de datos	Corel	Msrcorid
Tamaño fijo - SIFT	47 %	47 %
Tamaño dependiente - SIFT	42 %	43 %
Tamaño fijo - Histograma de color	14 %	36 %
Tamaño dependiente - Histograma de color	13 %	33 %

Cuadro 5: CER usando tamaño de rejilla dependiente de la escala

4.7.1. Concatenación de los vectores de características

La primera forma de combinar ambas características es mediante la simple concatenación de sus vectores, es decir, formando un vector de tamaño $n+m$ donde n es el tamaño de un vector de características SIFT y m el de un vector basado en histograma de color. La gráfica 10 muestra los resultados obtenidos mediante este método.

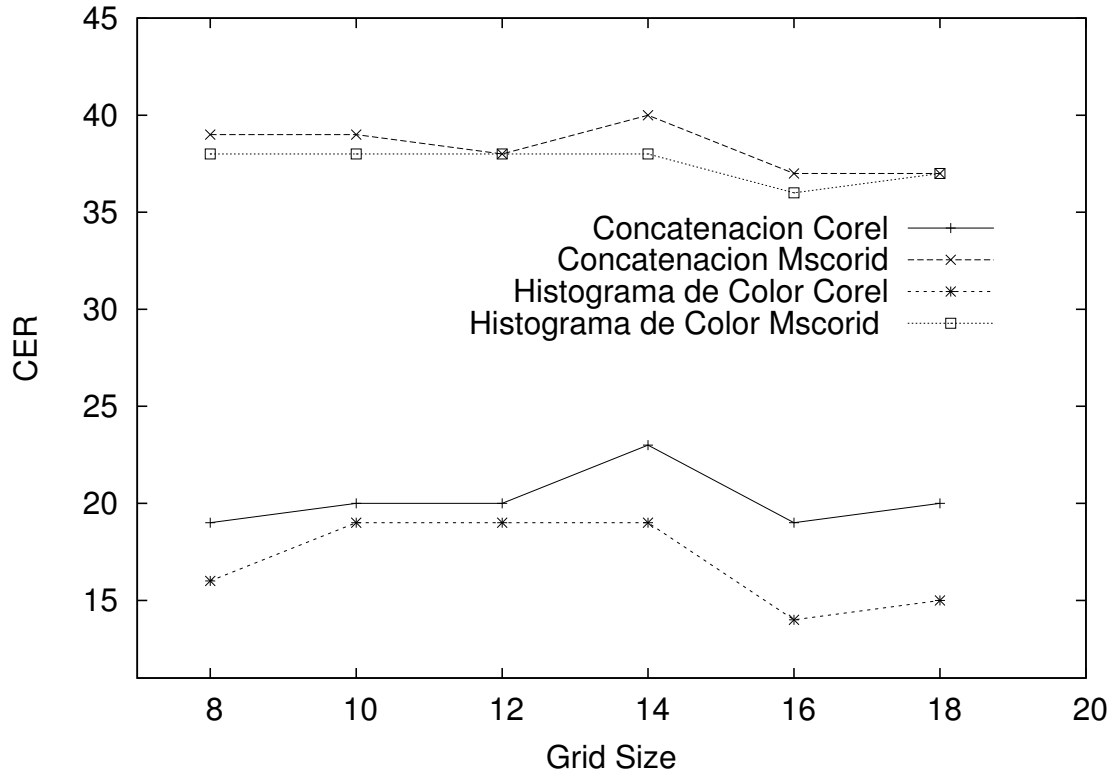

Figura 10: Concatenación de los vectores de características en función del tamaño de rejilla

Como podemos observar los resultados se encuentran en un punto intermedio

entre los obtenidos por las características SIFT y las del histograma de color, no siendo por lo tanto un método valido de unión de ambas características.

4.7.2. Concatenación de la cuantificación vectorial

El segundo método de combinación de ambos vectores de características fue la concatenación de los vectores producidos por la cuantificación vectorial de cada uno de los vectores de características. Este método producía unos resultados extremadamente malos de CER por lo que no es necesario comentarlos más allá de la indicación aquí realizada.

4.7.3. Correlación entre los grupos de pertenencia

El último método de combinación de características utilizado fue la correlación entre las distintas características para un mismo punto.

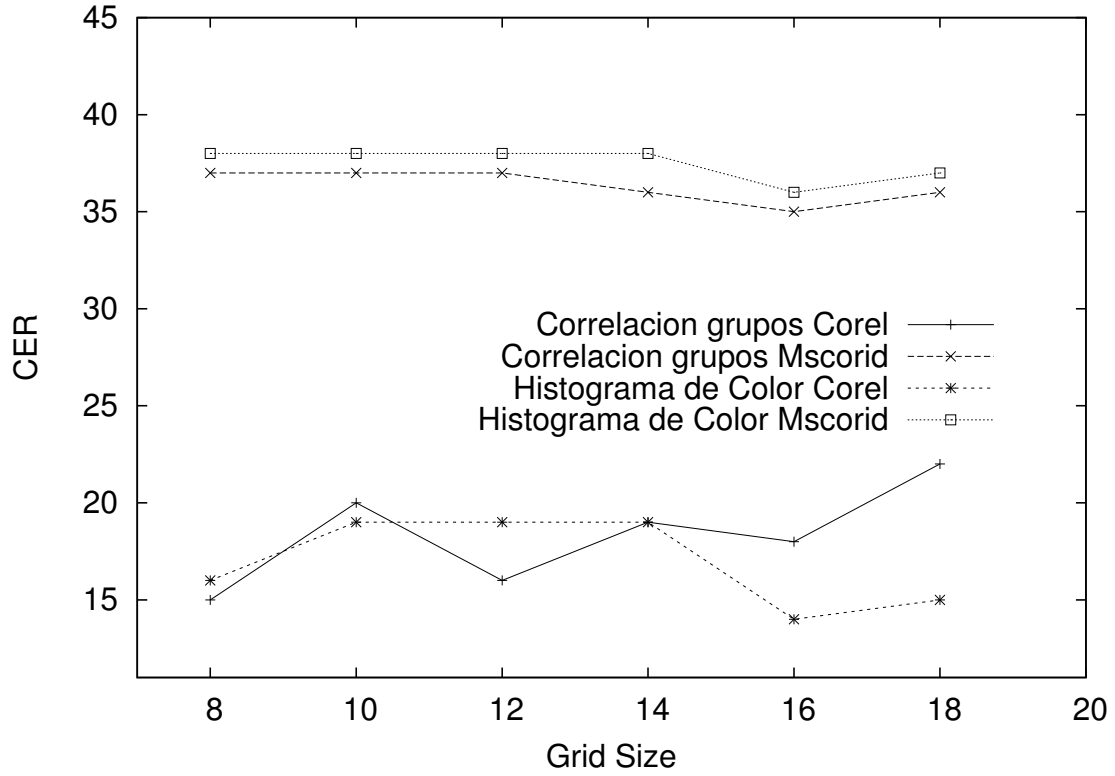

Figura 11: Correlación entre los grupos de pertenencia

Como se puede observar en la gráfica 11 los resultados son muy similares a los obtenidos con las características basadas en histograma de color, llegando a superarlos

en algún momento.

4.8. Correlación espacial

Por último se introdujo la correlación espacial entre las distintas celdas de la imagen que intenta captar las relaciones espaciales existentes entre las distintas zonas de la imagen y de esta forma aportar mayor información a los vectores de características. En la gráfica 12 se muestran los resultados obtenidos.

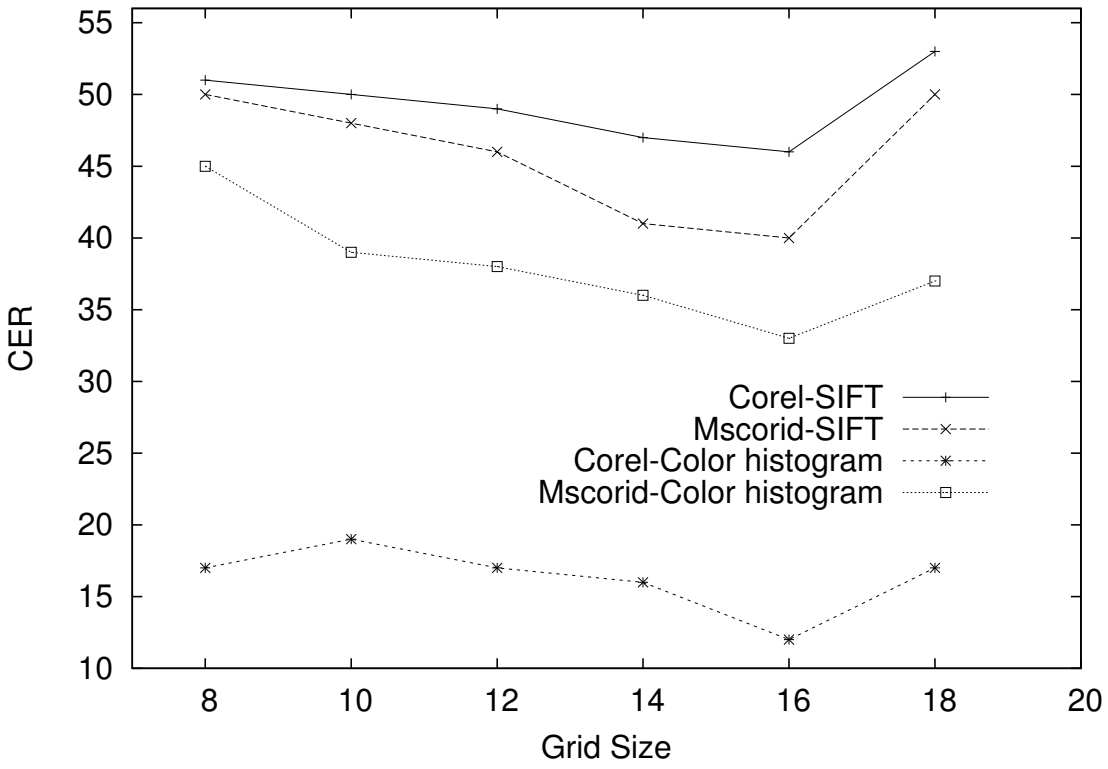

Figura 12: Resultados utilizando correlación espacial

Como podemos ver los resultados obtenidos por la correlación espacial de las características son superiores al resto de métodos. Para cada uno de los dos tipos de características usadas, la correlación espacial obtiene aproximadamente una mejora del 15 % sobre los resultados obtenidos. Esto se debe a que con este método conseguimos tener en cuenta la relación espacial de las distintas celdas de la rejilla. Al tratarse de imágenes, dicha relación espacial es muy relevante a la hora de realizar una correcta clasificación o un buen reconocimiento de imágenes.

A modo resumen podemos ver como la aplicación de las diversas técnicas presentadas consigue mejorar los resultados de las características básicas. Las tabla 6 y 7 resumen los mejores resultados obtenidos en cada una de las pruebas.

Método / Base de datos	Corel
Histograma de color	25 %
Histograma de color +J-S	14 %
Histograma de color +J-S + Correlación de características	15 %
Histograma de color +J-S + Tamaño de rejilla dependiente de la escala	13 %
Histograma de color +J-S + Correlación espacial	12 %

Cuadro 6: Resumen de resultados para la base de datos Corel

Método / Base de datos	Msrcorid
Histograma de color	44 %
Histograma de color +J-S	36 %
Histograma de color +J-S + Correlación de características	35 %
Histograma de color +J-S + Tamaño de rejilla dependiente de la escala	33 %
Histograma de color +J-S +Correlación espacial	33 %

Cuadro 7: Resumen de resultados para la base de datos Msrcorid

5. Conclusiones y Trabajo Futuro

5.1. Conclusiones

La correlación espacial mejora los resultados de las características densas tradicionales. Esto demuestra el hecho de que la información espacial es vital en el reconocimiento de imágenes. Por ello es necesario añadirla al proceso de reconocimiento de imágenes.

El presente trabajo demuestra también que las características basadas en SIFT están demasiado orientadas a la obtención y el procesado de puntos clave de la imagen y por ello fallan a la hora de reflejar la información en áreas extensas y homogéneas donde el algoritmo normal no encontraría ningún punto clave. Las texturas que ocupan gran parte de una imagen, como son la arena, el cielo o la vegetación no quedarían bien representadas, y por ello una aproximación más simple, como el histograma de color, obtiene mejores resultados.

5.2. Trabajo Futuro

Como trabajo futuro resta probar otros métodos de clasificación más avanzados como las redes neuronales o las máquinas de soporte vectorial.

En cuanto a las características usadas, existen muchas que son capaces de capturar mejor la información espacial del entorno de cada punto de la rejilla. Aproximaciones como [MM02] son capaces de extraer características de textura que quizás sean útiles a la hora de identificar las zonas amplias del fondo de la imagen. De esta forma se podría realizar una mejor clasificación si tenemos en cuenta que la mayoría de los subgrupos están más determinados por el fondo que por los pequeños detalles de la imagen; este es el caso de grupos como playas o montañas, donde la presencia de personas o edificio en la imagen no aportan información relevante para su clasificación.

También queda trabajar en la forma en la que las características se mezclan y en la manera en la que introducimos la correlación espacial. De esta forma, buscaríamos la manera de juntar amabas aproximaciones; tanto la mezcla de características como la correlación espacial, para producir sistemas que aprovechen las ventajas de las diferentes características extraídas así como la información espacial.

Otra de las posibilidades reside en la combinación de la aproximación basada en tamaño de rejilla dependiente de la escala y la correlación espacial ya que ambos méto-

dos han demostrado obtener unos buenos resultados. Se estudiaría entonces la relación espacial entre los distintos puntos de la rejilla teniendo en cuenta que las posiciones donde se extraerán las características es distinto en cada caso, por lo que la relación espacial no será siempre de una posición con otra. Puede darse el caso que dos puntos se encuentren al mismo tiempo colindantes con un tercero ya que los tres compartan bordes del recuadro de escala, o que un punto se encuentre incluido en el ambito de otro con una escala mayor.

6. Bibliografía

Referencias

[ADBP02] J. Assfalg, A. Del Bimbo, and P. Pala. Three-Dimensional Interfaces for Querying by Example in Content-Based Image Retrieval. 2002.

[AH+06] T. Ahonen, A. Hadid, et al. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 2037–2041, 2006.

[AXLT04] S.K. Antani, X. Xu, L.R. Long, and G.R. Thoma. Partial shape matching for CBIR of spine x-ray images. *Proceedings of SPIE*, 5307:1–8, 2004.

[b31] http://alipr.com.

[BDF02] K. Barnard, P. Duygulu, and D. Forsyth. Modeling the statistics of image features and associated text. *Document Recognition and Retrieval IX, Electronic Imaging*, 2002.

[BDG+03] K. Barnard, P. Duygulu, R. Guru, P. Gabbur, and D. Forsyth. The effects of segmentation and feature choice in a translation model of object recognition. 2003.

[BMOM] A. Bosch, X. Munoz, A. Oliver, and J. Martı. Modeling and classifying breast tissue density in mammograms.

[BTVG06] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. *Computer Vision–ECCV 2006*, pages 404–417, 2006.

[CFB04] P. Carbonetto, N. Freitas, and K. Barnard. A statistical model for general contextual object recognition. *Computer Vision-ECCV 2004*, pages 350–362, 2004.

[Dat] Microsoft Research Cambridge Object Recognition Image Database. http://research.microsoft.com/en-us/downloads/b94de342-60dc-45d0-830b-9f6eff91b301/default.aspx.

[Deb04] S. Deb. *Multimedia systems and content-based image retrieval*. Hershey, PA: Idea Group Pub., 2004.

[dFB01] N. de Freitas and K. Barnard. Bayesian latent semantic analysis of multimedia databases. 2001.

[dFBB+02] N. de Freitas, E. Brochu, K. Barnard, P. Duygulu, and D. Forsyth. Bayesian models for massive multimedia databases: a new frontier. *7th Valencia International Meeting on Bayesian Statistics/2002 ISBA International Meeting, June*, pages 2–6, 2002.

[DJLW07] R. Datta, D. Joshi, J. Li, and J.Z. Wang. Image retrieval: Ideas, influences, and trends of the new age. *ACM Computing Surveys*, 39:65, 2007.

[DWK+05] T. Deselaers, T. Weyand, D. Keysers, W. Macherey, and H. Ney. FIRE in ImageCLEF 2005: Combining Content-based Image Retrieval with Textual Information Retrieval. *CLEF*, pages 688–698, 2005.

[EZWVG06] M. Everingham, A. Zisserman, C. Williams, and L. Van Gool. The PASCAL visual object classes challenge 2006 (VOC2006) results, 2006.

[FPZ03] R. Fergus, P. Perona, and A. Zisserman. Object class recognition by unsupervised scale-invariant learning. 2003.

[HM02] J. Han and K.K. Ma. Fuzzy color histogram and its use in color image retrieval. *IEEE Transactions on Image Processing*, 11(8), 2002.

[IA02] Q. Iqbal and JK Aggarwal. Retrieval by classification of images containing large manmade objects using perceptual grouping. *Pattern Recognition*, 35(7):1463–1479, 2002.

[JFS95] C.E. Jacobs, A. Finkelstein, and D.H. Salesin. Fast multiresolution image querying. *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 277–286, 1995.

[JG05] T.Y. Jea and V. Govindaraju. A minutia-based partial fingerprint recognition system. *Pattern Recognition*, 38(10):1672–1684, 2005.

[Joh00] B. Johansson. A survey on: Contents based search in image databases. *Report LiTH-ISY*, 2000.

[JZL96] A.K. Jain, Y. Zhong, and S. Lakshmanan. Object matching using deformable templates. *IEEE Transactions on pattern analysis and machine intelligence*, 18(3):267–278, 1996.

[KKLK09] T. Kinnunen, J.K. Kamarainen, L. Lensu, and H. K
.[a]lvi
.[a]inen. Bag-of-Features Codebook Generation by Self-Organisation. *Advances in Self-Organizing Maps*, pages 124–132, 2009.

[Low04] D.G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[MM02] BS Manjunath and WY Ma. Texture features for browsing and retrieval of image data. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(8):837–842, 2002.

[MM09] M. Mirza-Mohammadi. Contextual Bag-Of-Visual-Words and ECOC-Rank for Retrieval and Multi-class Object Recognition. 2009.

[SC95] J.R. Smith and S.F. Chang. Single color extraction and image query. In *Proc. IEEE Int. Conf. on Image Proc*, pages 528–531. Citeseer, 1995.

[SGSL05] R. Sim, M. Griffin, A. Shyr, and J.J. Little. Scalable real-time vision-based SLAM for planetary rovers. In *IEEE IROS Workshop on Robot Vision for Space Applications*, pages 16–21. Citeseer, 2005.

[USS09] J.R.R. Uijlings, A.W.M. Smeulders, and RJH Scha. Real-time bag of words, approximately. In *Proceeding of the ACM International Conference on Image and Video Retrieval*, pages 1–8. ACM, 2009.