



ESCUELA SUPERIOR DE INGENIERÍA INFORMÁTICA

INGENIERÍA INFORMÁTICA SUPERIOR

CURSO ACADÉMICO 2012/2013

Proyecto de Fin de Carrera

**Una metodología para la clasificación de imágenes  
binarias basada en Redes Complejas**

Autor : David Jiménez Pérez

Tutores: Ángel Sánchez Calle

Miguel Romance del Río

# Agradecimientos

En primer lugar, quiero dar las gracias a mis tutores, ya que sin ellos este proyecto no hubiera sido posible. Ya que me han dedicado mucho tiempo y esfuerzo, en las incontables reuniones, donde me han resuelto todas las dudas.

También quiero agradecer el apoyo que me han ofrecido mi familia y amigos durante todo el tiempo en el que he realizado este proyecto.

Tampoco debo olvidar en mencionar a todos los compañeros que he conocido, con los que he pasado tantos momentos de diversión y de fatigas durante tantos años.

A todos os doy las gracias, ya que sin vosotros no hubiera llegado tan lejos.

# Resumen

En el campo de la Visión Artificial y en concreto en el Reconocimiento de Patrones existe el problema de clasificación de objetos a partir de su silueta, el cual tiene aplicaciones en diversos campos como, por ejemplo, en la Robótica y en el Procesamiento de Imágenes Médicas.

El objetivo de este proyecto consiste en aportar otra manera de clasificar los objetos de una base de datos de imágenes binarias según a la categoría que pertenecen, basándose en la teoría de Redes Complejas.

La solución consta de varias etapas:

1. Construir la red compleja a partir de una imagen de la base de datos.
2. Obtener los histogramas normalizados de los resultados de cada una de las siguientes características:
  - a) Distribución de los valores de grado: Indican el número de píxeles a los que está conectado cada píxel de la imagen.
  - b) Distribución de una medida de circularidad de la imagen: Utilizando para ello el *algoritmo de Bresenham para circunferencias*.
  - c) Obtención para todos los grados existentes en la imagen del valor de distribución de probabilidades *joint degree*.
3. Calcular la matriz para cada característica con la comparativa de la similitud entre los histogramas de pares de imágenes.
4. A partir de las matrices anteriores, sintetizar en una matriz global, toda la información aportada por las características calculadas. En cada una de estas matrices el valor de cada posición corresponde a la distancia o diferencia entre cada par de imágenes.

5. Con la matriz obtenida previamente, evaluar las posibilidades de reconocimiento de formas, variando el rango de valores de la distancia a los que se asume que dos imágenes pertenecen a la misma categoría.
6. El estudio evalúa los datos obtenidos por las diferentes versiones de la matriz global, conseguidas aplicando diversas ecuaciones, buscando cual de ellas finalmente aporta una mejor clasificación frente al resto.

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Planteamiento . . . . .	2
1.2. Hipótesis en el desarrollo del proyecto . . . . .	5
1.3. Solución propuesta . . . . .	5
1.4. Objetivos . . . . .	7
1.5. Estructura del proyecto . . . . .	7
<b>2. Estado del arte</b>	<b>9</b>
2.1. Clasificación binaria de imágenes . . . . .	9
2.1.1. Métodos estadísticos . . . . .	9
2.1.2. Métodos basados en transformaciones . . . . .	11
2.1.3. Métodos basados en redes neuronales artificiales . . . . .	11
2.1.4. Método de programación dinámica . . . . .	13
2.2. Redes complejas . . . . .	13
2.3. Métricas en las redes complejas . . . . .	19
2.3.1. Grado . . . . .	19
2.3.2. Joint degree . . . . .	19
2.3.3. Conectividad . . . . .	20
2.3.4. Grado Jerárquico . . . . .	21
<b>3. Descripción Informática</b>	<b>22</b>
3.1. Metodología . . . . .	22
3.2. Herramientas utilizadas . . . . .	25
3.2.1. Matlab . . . . .	25
3.2.2. L <sup>A</sup> T <sub>E</sub> X . . . . .	26
3.2.3. Hardware y entorno de desarrollo . . . . .	27

<b>4. Métodos de clasificación</b>	<b>29</b>
4.1. Crear la lista de adyacencia . . . . .	29
4.2. Cálculo de la característica de grado . . . . .	32
4.3. Cálculo de la circularidad . . . . .	35
4.4. Cálculo de la propiedad Joint degree probability . . . . .	44
4.5. Cálculo del ratio de Fisher . . . . .	48
<b>5. Análisis de la información</b>	<b>51</b>
5.1. La base de datos del proyecto . . . . .	51
5.2. Análisis inicial . . . . .	55
5.2.1. Análisis de la característica de grado . . . . .	57
5.2.2. Análisis de la característica de la circularidad . . . . .	59
5.2.3. Analisis de la característica de la distribución joint degree	61
5.2.4. Matrices de confusión de las características estudiadas	62
5.3. Funciones de combinación . . . . .	66
5.3.1. Experimentación con la combinación basada en la media	67
5.3.2. Experimentación con la combinación basada en la me- dia cuadrática . . . . .	69
5.3.3. Experimentación con la combinación basada en el máxi- mo . . . . .	70
5.3.4. Experimentación con la combinación basada en la dis- tribución de pesos primera . . . . .	72
5.3.5. Experimentación con la combinación basada en la dis- tribución de pesos segunda . . . . .	73
5.3.6. Experimentación con la combinación basada en la dis- tribución de pesos tercera . . . . .	75
5.3.7. Experimentación con la combinación basada en la dis- tribución de pesos cuarta . . . . .	76
5.3.8. Experimentación con la combinación basada en la dis- tribución de pesos quinta . . . . .	77
5.3.9. Experimentación con la combinación basada en la dis- tribución de pesos sexta . . . . .	78
5.3.10. Experimentación con la combinación basada en la dis- tribución de pesos según el ratio de Fisher . . . . .	79
5.3.11. Conclusiones de los análisis iniciales . . . . .	80
5.4. Estudiando la topología . . . . .	81
5.4.1. Errores generales en la clasificación . . . . .	84
5.4.2. Falsos positivos en la clasificación . . . . .	87

5.4.3.	Falsos negativos en la clasificación . . . . .	89
5.5.	Análisis cuantitativo de la clasificación . . . . .	93
5.5.1.	Datos cuantitativos de la combinación basada en los máximos . . . . .	95
5.5.2.	Datos cuantitativos de la combinación basada en la media	96
5.5.3.	Datos cuantitativos de la combinación basada en la media cuadrática . . . . .	97
5.5.4.	Datos cuantitativos de la combinación basada en la distribución de pesos primera . . . . .	98
5.5.5.	Datos cuantitativos de la combinación basada en la distribución de pesos segunda . . . . .	100
5.5.6.	Datos cuantitativos de la combinación basada en la distribución de pesos tercera . . . . .	101
5.5.7.	Datos cuantitativos de la combinación basada en la distribución de pesos cuarta . . . . .	102
5.5.8.	Datos cuantitativos de la combinación basada en la distribución de pesos quinta . . . . .	103
5.5.9.	Datos cuantitativos de la combinación basada en la distribución de pesos sexta . . . . .	105
5.5.10.	Datos cuantitativos de la combinación basada en la distribución de pesos dada por el ratio de Fisher . . . .	106
5.5.11.	Matrices de confusión de las 10 combinaciones estudiadas	107
5.5.12.	Conclusión del análisis cuantitativo . . . . .	118
<b>6.</b>	<b>Conclusión</b>	<b>121</b>
6.1.	Conclusiones . . . . .	121
6.2.	Líneas futuras de trabajo . . . . .	124
<b>Apéndices</b>		<b>III</b>
<b>Figuras de la BBDD</b>		<b>IV</b>

# Capítulo 1

## Introducción

El reconocimiento de objetos ocupa una parte importante de los trabajos realizados en el campo de la Inteligencia Artificial. Para el desarrollo de clasificadores cuyo aprendizaje sea sin supervisión humana, es un aspecto clave en el ámbito de la navegación dentro de la robótica e incluso en el análisis de imágenes para fines médicos.

Los resultados de clasificación de los diferentes estudios e investigaciones realizados no han llegado a obtener una clasificación perfecta. El principal factor que influye en la calidad de la clasificación es la robustez con la que se comporten los algoritmos respecto a una serie de características de los objetos en la imagen como son el escalado o dimensión de estos dentro de una clase, la posición y la rotación en la imagen del objeto. Ante las cuales, de manera ideal, los resultados del clasificador no deben verse afectados.

Aparte de esas características, también hay que tener en cuenta que dentro de una misma clase de objetos, pueden existir algunas muestras con un contorno que se distancia del más habitual de esa clase. Como por ejemplo, dentro de una categoría de “animales” como pueden ser las “aves”, no tienen una proporción similar una “gaviota” frente a un “pavo”, como una “gaviota” y una “paloma”, como se aprecian en las siguientes siluetas.





Figura 1.1: Comparativa de las similitudes y diferencias de 3 siluetas de aves

En este proyecto se propone un nuevo enfoque para la tarea de clasificación de objetos, basándose en la teoría de las Redes Complejas y usando una serie de características que resulten lo más invariantes posibles frente a factores como el escalado y la posición que ocupan los objetos en las imágenes.

Con los resultados obtenidos se realizan diversas comparativas para poder determinar de entre todas, cuál es la métrica que aporta un mejor resultado.

## 1.1. Planteamiento del problema

La base del trabajo consiste en agrupar el mayor número de imágenes posibles dentro de la categoría correcta a la que pertenecen. Para ello en un primer lugar se construye la estructura correspondiente para cada imagen para poder, de esta forma, obtener las características necesarias para diferenciar las clases de objetos.

Una vez obtenidas esas características se aplican una serie de ecuaciones, para crear diferentes escenarios en los que se van restringiendo de forma progresiva el valor de la distancia en el que se considera que dos imágenes pertenecen a la misma categoría hasta el caso extremo en el que cada imagen

pertenece a su propia clase.

Es decir, como paso final se estudia la topología de la red y el ritmo al que se reduce el número de conexiones según disminuye el valor permitido para las conexiones entre cada par de imágenes.

Para este proyecto se ha tomado como fuente para realizar las pruebas, una base de datos (BBDD) con imágenes de siluetas, o imágenes binarias, disponibles en la página web del departamento de ingeniería de la Universidad de Brown [11]. La BBDD está estructurada en 17 categorías con 12 imágenes por categoría.

En primer lugar, se debe transformar la imagen recibida en un conjunto de datos manejables por el código implementado. El modelo seleccionado han sido las Redes Complejas. Esta rama de la teoría de redes estudia la dinámica o el cambio topológico de grafos, aplicando diversos criterios sobre las conexiones entre nodos de la red. Las Redes Complejas se completan con los datos estadísticos extraídos del análisis de su estructura.

La estructura seleccionada para almacenar esa información en el programa ha sido la de una lista de adyacencia, ya que debido a las restricciones de memoria para esta etapa del trabajo no se ha podido trabajar con matrices, porque en una imagen de dimensiones, por ejemplo de 115x90, se tendría que trabajar con una matriz de 10350x10350.

El peso de las conexiones entre dos píxeles en una imagen se obtiene como la diferencia en valor absoluto de las intensidades de ambos píxeles, tal como se explica en el artículo [4]. A continuación, se calculan una serie de características, cuyos valores van a quedar registrados en un histograma.

La primera es el *grado* que consiste en contabilizar el número de píxeles a los que está conectado cada nodo o píxel de la imagen. Esta propiedad es similar a la conectividad, con la diferencia en que no se tiene en cuenta los pesos de las conexiones de las aristas y como en este proyecto se trabaja con imágenes binarias, en las cuales el peso de las conexiones va a ser sólo de valor 1, no tiene sentido aplicar esta otra característica.

La siguiente característica es la de la *circularidad*, la cual se apoya en el

algoritmo de *Bresenham* aplicado a circunferencias, donde a un radio creciente se calcula el porcentaje de píxeles de cada circunferencia que pertenecen al objeto.

La última característica a calcular, es la denominada *joint degree probability* que aparece en el artículo [5]. Su objetivo consiste en obtener el valor de probabilidad de que dos píxeles con un determinado valor de grado cada uno ellos, estén conectados mediante una arista, es decir que sean nodos adyacentes.

Una vez que se ha obtenido la estructura y sus características para cada imagen es necesario obtener una tabla donde se indique la distancia o grado de diferencia entre imágenes para cada una de estas características. Así pues, para ello a cada característica de una imagen se compara con la misma característica del resto de las imágenes, usando la siguiente ecuación:

$$\frac{\sum_{i=1}^n |h_i - p_i|}{\sum_{i=1}^n \max\{h_i, p_i\}} \quad (1.1)$$

donde  $h_i$  y  $p_i$  son el elemento en la posición  $i$  del histograma de la imagen  $h$  y de la imagen  $p$  respectivamente, para la misma característica. Esta ecuación permite obtener resultados normalizados entre 0 y 1

Una vez que se han calculado las distancias entre todas las imágenes para las 3 características consideradas, es necesario reducir esa información a un único valor para cada par de imágenes que indique lo similar o diferente que es respecto al resto de imágenes de la base de datos.

Para ello se experimenta con distintas formas de combinación de resultados, usando como punto de partida las tres matrices calculadas, obteniendo para cada posición la media, la media cuadrática, el máximo y varias posibilidades de reparto de pesos entre cada característica.

A partir de cada uno de los resultados anteriores se compara la calidad de la clasificación de cada uno de ellos, analizando cada matriz dentro de un rango de valores que actúan como umbral donde se consideran objetos de la misma clase aquellos para los que su distancia obtenida sea inferior a ese umbral.

## 1.2. Hipótesis en el desarrollo del proyecto

Durante la realización del proyecto, se ha supuesto una serie de condiciones respecto a los datos de entrada con los que se trabaja, o en otras palabras, las imágenes de la base de datos, como se indica, a continuación:

- Todas las imágenes se componen de un objeto con un contorno o silueta de color uniforme sobre un fondo blanco.
- Las imágenes procesadas son en blanco y negro, aunque si se da el caso en el que se introduce una imagen en escala de grises o en color, el programa se encarga de transformarla a blanco y negro.
- Las dimensiones de las imágenes deben ser similares para el cálculo correcto de la característica de la circularidad.

## 1.3. Solución propuesta

La siguiente imagen muestra una visión global de los pasos de los que se compone el proyecto:

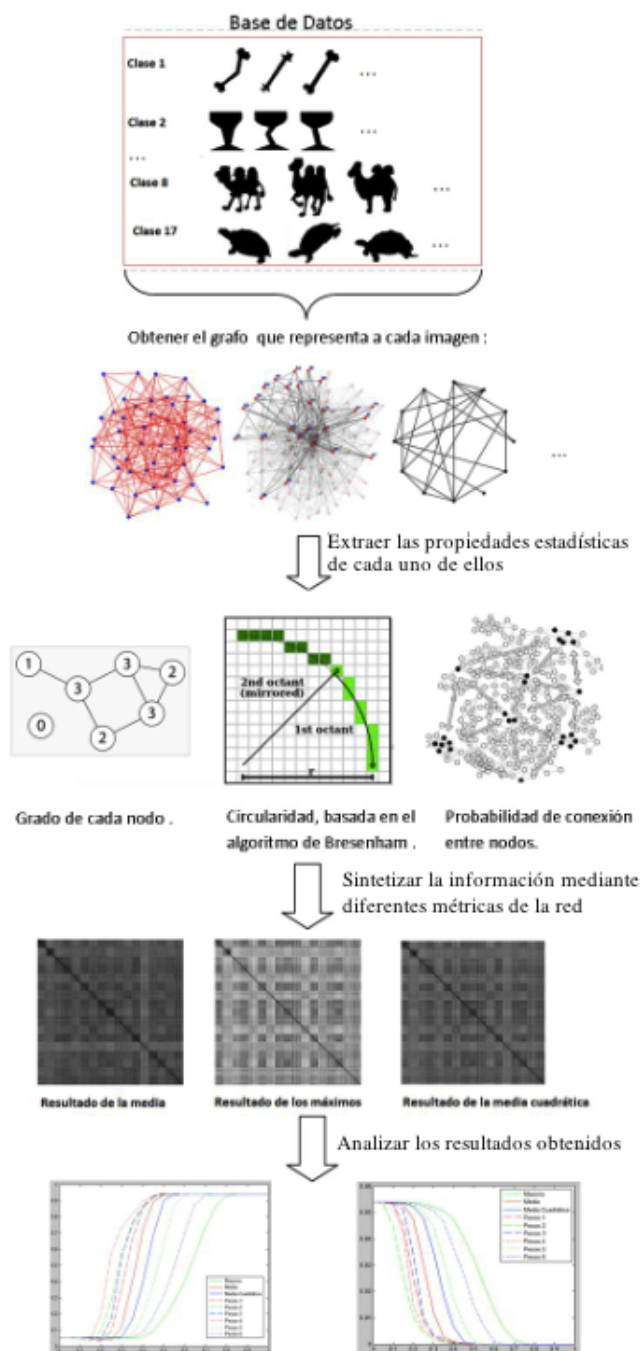


Figura 1.2: Esquema de las principales etapas de la solución propuesta

## 1.4. Objetivos

El principal objetivo del proyecto es aportar un clasificador de formas, basado en técnicas de redes complejas, para ello se consideran las siguientes etapas en el método de clasificación:

- Obtener la red compleja equivalente a cada imagen, para poder trabajar con cada una de ellas.
- Aportar los datos estadísticos, compuestos por los histogramas que recogen los datos de las siguientes características: *grado*, *circularidad* y *joint degree probability*.
- Reunir y generalizar los resultados obtenidos en un sólo valor, para comparar o determinar el grado de similitud entre cada par de imágenes.
- Analizar todos los resultados dados por el grafo que "relaciona" las imágenes, siguiendo como criterio el descartar todas aquellas conexiones cuyo valor supera un umbral dado.
- Presentar esas conclusiones a partir de un conjunto de pruebas visuales que permitan interpretar de forma visual los resultados obtenidos.
- Por último, realizar un análisis en más profundidad de las matrices de confusión intra-clases y así conocer los errores cometidos por cada resultado. Se trata de determinar finalmente cuáles son las métricas que aportan mejores resultados en el reconocimiento.

## 1.5. Estructura del proyecto

La memoria del proyecto se ha organizado en cinco capítulos. El primero, es el presente capítulo, en el que se intenta aportar una visión general de los objetivos propuestos para este trabajo y una breve descripción de la realización del mismo.

El segundo capítulo reúne una pequeña explicación de los principales clasificadores que son utilizados en el campo de reconocimiento de patrones y concretamente en catalogar clases a partir de siluetas de imágenes. También

se han incluido los conceptos más importantes de las Redes Complejas y algunas de sus propiedades estadísticas más relevantes.

El tercer capítulo describe las herramientas informáticas utilizadas a lo largo del proyecto, así como la metodología que se ha seguido para realizar el trabajo.

El cuarto capítulo, es uno de los más importante de toda la memoria. En él se describe todo el proceso de desarrollo del proyecto para clasificar imágenes utilizando la teoría de las Redes Complejas y como, de forma general, se ha implementado el código que calcula las diversas propiedades estadísticas de la red.

Una vez descrito todo el proceso de creación de la red y sus datos asociados, en el capítulo quinto se analizan todos los resultados obtenidos para determinar cuál de todas las variantes del método produce mejores resultados.

Por último en el sexto capítulo se hace un análisis global del proyecto y los resultados señalándose las conclusiones finales. También se proponen las posibles líneas futuras de trabajo en este campo.

# Capítulo 2

## Estado del Arte

### 2.1. Clasificación binaria de imágenes

El problema de la clasificación binaria de objetos juega un papel importante en el reconocimiento, emparejamiento y registro de objetos en los sistemas de la Visión Artificial.

Algunas categorías de clasificadores comprenden los métodos de emparejamiento basados en el uso de las características extraídas de los contornos de las imágenes, frente a otras como la medición de intensidad de la iluminación en la imagen o la comparación entre la región externa al objeto y su área interna. A continuación, se comentarán algunos de los métodos utilizados para afrontar este problema.

#### 2.1.1. Métodos estadísticos

El principal objetivo de estos métodos es construir el clasificador en dos fases: la fase de aprendizaje y la fase de clasificación. En la fase de aprendizaje se obtiene la distribución de probabilidades para varias características extraídas de las imágenes.

En esta primera fase se estiman diferentes distribuciones de probabilidad para cada una de las clases, proporcionando límites en el espacio de características que separan los patrones que identifican a cada una de las clases.



En la posterior fase de clasificación se utilizan los límites en el espacio de características y las distribuciones de probabilidad para clasificar las siluetas. *Hidden Markov Model* y *Support Vector Machines* son clasificadores basados en métodos estadísticos. En el modelo de Markov descrito en el artículo [1], cada clase de objetos está representada por una cadena de Markov exclusiva con un número finito de estados.

Posteriormente, se aplica un proceso de aprendizaje supervisado para estimar los parámetros del modelo. En la fase de clasificación, el objeto que está siendo clasificado se compara con el resto de modelos o clases.

Las *Support Vector Machines*, fueron propuestas por Vapnik en 1992 y su principal característica es que permiten construir fronteras de decisión flexibles en problemas de clasificación. Su funcionamiento consiste en obtener, a partir de un conjunto de vectores, otro conjunto de dimensión más alta para solucionar el problema linealmente.

Este proceso se aplica a aquellas clases que no son linealmente separables, para conseguir que al final del proceso lo sean. A continuación, se muestra una imagen en la que se parte de un conjunto no linealmente separable y después al conseguir un conjunto de mayor dimensión sí es posible separar ambas clases.

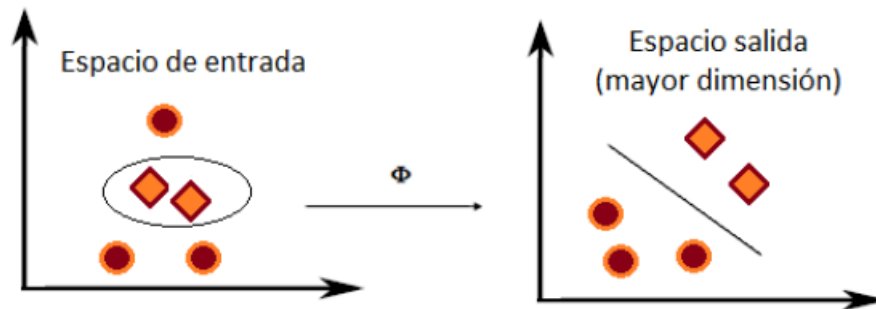


Figura 2.1: Funcionamiento general de las *Support Vector Machines*

### 2.1.2. Métodos basados en transformaciones

La clave en esta categoría de clasificadores está en la transformación de la forma o silueta del objeto del dominio espacial al dominio de la transformada (por ejemplo el dominio de la frecuencia), que aporta un conjunto de coeficientes que representan a la silueta. El conjunto de coeficientes obtenidos puede utilizarse como vector de características para medir la similitud entre las distintas clases de objetos.

La transformada de Fourier es uno de los representantes de esta categoría. En este método, el contorno del objeto se representa como una señal unidimensional. Una vez que se ha calculado la transformada de Fourier, se compara un conjunto seleccionado de coeficientes de Fourier, también conocidos como descriptores de Fourier, frente a otro conjunto de descriptores para poder clasificar el objeto.

Otros clasificadores son la transformada *wavelet* y el *análisis de componentes principales*. Este último consiste en una técnica de reducción de la dimensionalidad de los datos, minimizando la pérdida de información relevante, utilizando para ello una matriz de covarianza.

### 2.1.3. Métodos basados en redes neuronales artificiales

Este tipo de redes tratan de emular las características y propiedades de las redes neuronales biológicas, las cuales otorgan al ser humano la capacidad de aprendizaje, por ello se han tomado como modelo para conseguir un aprendizaje automatizado.

La estructura general de estas redes consiste en un conjunto de neuronas en cada una de las capas que conforman la red. Cada neurona recibe como entrada el valor de salida de todas las neuronas de la capa anterior, multiplicado por el peso de la conexión de cada una de ellas con la actual. La función de activación consiste en la suma de los resultados de la capa anterior multiplicados por sus respectivos pesos, por lo que esta función podría compararse con los impulsos eléctricos que recorren las dendritas de la red biológica que activan las neuronas necesarias para el proceso correspondiente.

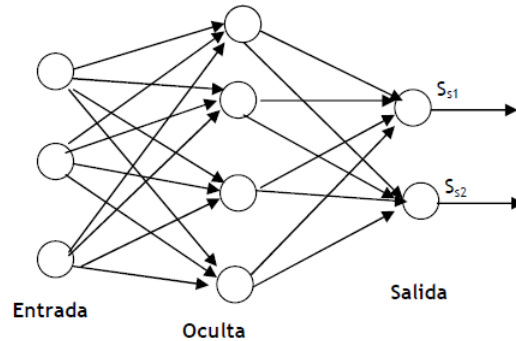


Figura 2.2: Esquema general de una red neuronal artificial con una capa oculta

Como se puede apreciar en la figura 2.2, la red consta de 3 capas, la capa de entrada estará formada por tantas neuronas como de elementos conste el vector de características dado. Las capas ocultas reciben los resultados de la función de activación de cada capa anterior y pueden estar formadas por cualquier número de neuronas, aunque hay que tener en cuenta que a mayor número de neuronas, el coste computacional va a ser más elevado, lo que al final conlleva en más tiempo de ejecución. El número de capas ocultas también es variable y al igual que con el número de neuronas, hay que alcanzar un convenio entre la precisión del resultado de clasificación y el tiempo de cómputo.

Como último comentario sobre la figura 2.2, hay que decir que la capa de salida consta de tantas neuronas como diferentes tipos de clases se están clasificando y de todas ellas sólo se activará la que obtenga el valor más alto de activación.

Las redes neuronales artificiales [6] son una de las herramientas más utilizadas en las tareas de clasificación y reconocimiento. Su fundamento en la forma de trabajar es parecido a los métodos estadísticos, ya que también consta de una etapa de aprendizaje y finalmente la etapa de clasificación. En el aprendizaje se utiliza un conjunto de características extraídas de todas las clases de objetos, para que la red pueda ir adaptando sus parámetros hacia unos resultados de clasificación óptimos. En la etapa de clasificación se suministran vectores de características, para que la red neuronal los catalogue

en la capa de salida.

Hay diversas variaciones de redes neuronales empezando por el aprendizaje para el cual pueden utilizarse algoritmos como el de retropropagación (*backpropagation*) o el de propagación rápida (*quick propagation*), otro aspecto a tener en cuenta es el número de capas ocultas que se han utilizado y también la función de activación entre cada capa entre las que se encuentran la función sigmoïdal o la función tangente hiperbólica, entre otras.

A pesar de las diferentes variaciones en los algoritmos que puedan utilizarse con la red, todas ellas siguen la idea esencial descrita al principio de esta sección.

#### 2.1.4. Método de programación dinámica

Consiste en representar como cadenas, el contorno de un objeto desconocido a clasificar y los contornos de objetos utilizados como prototipos, es decir aquellos objetos de clases ya conocidas, en nuestro caso serían las imágenes de la base de datos con la que vamos a realizar el estudio.

El problema se reduce a calcular la distancia entre dos cadenas. Donde el prototipo con la distancia mínima frente a la cadena del objeto desconocido, se considera la clase a la que pertenece.

Bunke y Buhler introdujeron un algoritmo robusto para el reconocimiento de objetos bidimensionales, que es invariante frente al escalado, rotación y traslación.

## 2.2. Redes complejas

Las Redes Complejas pueden situarse en un campo de estudio entre la teoría de grafos y la mecánica estadística, dotándola de una naturaleza multidisciplinar, ya que se aplica en diversos campos como por ejemplo en el de la Inteligencia Artificial, las Matemáticas o incluso la Física. Gracias a su versatilidad, se ha generado un gran interés por aplicar este modelo a muchos de los problemas e investigaciones actuales.

Aunque en el campo de la Visión Artificial no se ha explotado demasiado, incluso a pesar de que muchos de los estudios podrían realizarse con la teoría de Redes Complejas. Las pocas investigaciones realizadas en este campo consisten en el estudio de la representación de redes complejas y el posterior análisis del crecimiento de las redes complejas y su topología.

Un ejemplo de las posibles aplicaciones de las Redes Complejas puede ser el modelo desarrollado por Fontoura Costa, cuya meta era analizar imágenes de células de epitelio transformando toda la información referente a las conexiones entre las células a los correspondientes grafos [8]. También ha elaborado otros trabajos más llamativos basados en Redes Complejas como por ejemplo, un estudio donde se calcula la dificultad en encontrar la salida de un laberinto [3], el análisis de la accesibilidad a las redes de metro de las ciudades de Londres y París e incluso un trabajo que tiene como objetivo el procesamiento del lenguaje [9].

Como ya se ha comentado, las redes complejas se basan en grafos con pesos, los cuales consisten en un conjunto de vértices conectados por aristas ponderadas. El grafo se representa habitualmente con la nomenclatura  $G(V, E)$ , donde  $V = \{v_1, v_2, \dots, v_n\}$  es el conjunto de todos los vértices y  $E = \{e_1, e_2, \dots, e_n\}$  es el conjunto de aristas. Una representación visual del grafo de una red compleja, puede verse en la siguiente imagen.

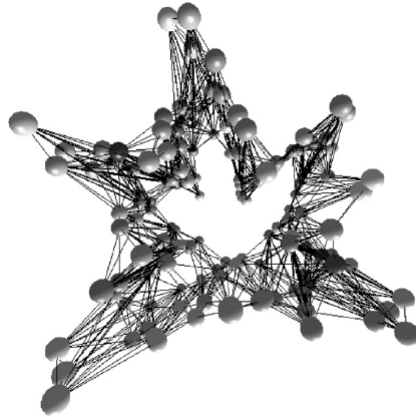


Figura 2.3: Representación de un contorno modelado como una red compleja.

Las redes complejas están clasificadas según sus propiedades en tres grandes categorías: *random networks*, *small-world networks*, *scale-free networks*. Las redes aleatorias o *random networks* fueron propuestas por Erdős y Rényi [7] y son consideradas como el modelo más básico de redes. En este modelo, las aristas se forman de manera completamente aleatoria.

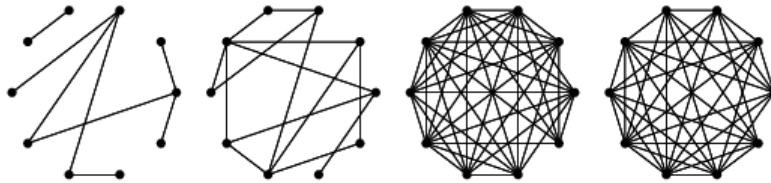


Figura 2.4: Ejemplo de la distribución de las aristas en las distintas redes aleatorias, contando con el mismo número de nodos en todas ellas.

La propiedad principal de estas redes es la posibilidad de que un vértice puede ser alcanzado por cualquier otro a través de pocas conexiones o aristas, lo que tiene como resultado un gran número de bucles de longitud 3, que consiste en que si el vértice  $x$  está conectado a los vértices  $v$  y  $w$ , hay una probabilidad muy alta de que los vértices  $v$  y  $w$  también estén conectados entre ellos, característica conocida como el coeficiente de *clustering*.

De las propiedades anteriormente extraídas, gracias al estudio que realizaron Watts y Strogatz sobre este primer conjunto de redes, surgió la segunda categoría, las redes *small world* [14].

En las redes *small world* las aristas no son colocadas de forma totalmente aleatoria, sino que son el resultado de sustituir aleatoriamente una fracción  $P$  pequeña de los enlaces de una retícula de dimensión  $d$  con nuevas aristas aleatorias, interpoladas entre el caso sin aleatoriedad ( $P = 0$ ) y el grafo completamente aleatorio ( $P = 1$ ). Una red *small world* está caracterizada por las siguientes propiedades:

1. La vecindad local se preserva, como en el caso de los grafos reticulados.
2. El diámetro de la red está cuantificado por la media de la distancia

mínima entre dos vértices, el cual se incrementa con el número de vértices como en el caso de las redes aleatorias.

La segunda propiedad es la que da significado al nombre de este tipo de redes, porque es posible conectar cualquier par de vértices en la red a través de unas pocas aristas, y la conectividad local sugiere que la red tiene una dimensionalidad finita.

Por último, con respecto a este segundo tipo de redes, cabe destacar que varias redes del mundo real comparten las propiedades y estructura de las redes *small world*, como por ejemplo las redes eléctricas de múltiples países o la red formada por las colaboraciones que ha realizado en películas un actor de cine.

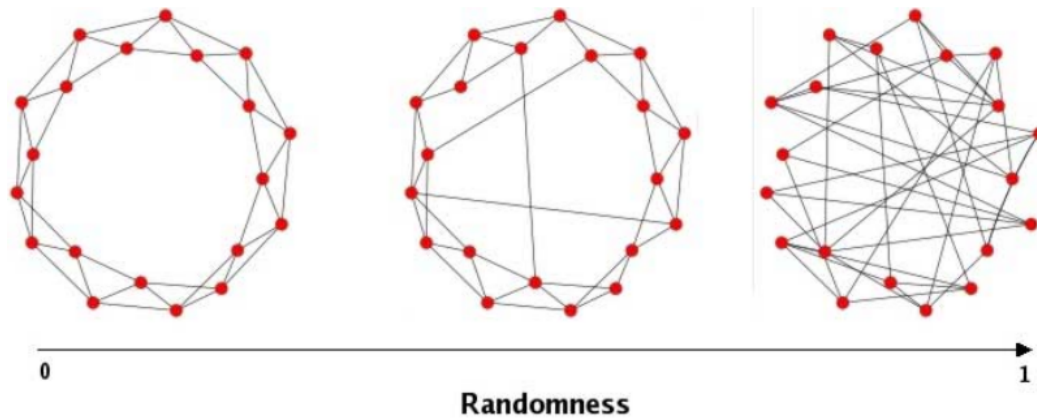


Figura 2.5: Evolución de una red regular hasta una red completamente aleatoria, según se incrementa  $P$ . El caso intermedio se corresponde con una red *small world*.

El tercer tipo de redes complejas son las denominadas *scale free*, propuestas por Barabási y Albert [2], que surgen en el contexto de una red dinámica que crece formando nuevas aristas entre los vértices con un nivel mayor de conectividad en la red.

Estas redes podrían también clasificarse como un tipo específico de redes *small world*, porque tienen un coeficiente de *clustering* mayor que en el caso

de las redes aleatorias y su diámetro se incrementa de forma logarítmica con el número de vértices. La conectividad de los vértices sigue una distribución asintótica dada por  $P(k) \approx k^{-\gamma}$  con  $\gamma = 3$  y  $n \rightarrow \infty$ .

Entre las características de las redes libres a escala o *scale free*, destaca el bajo número de vértices con un grado que supera ampliamente al valor medio del grado. A estos vértices con un valor alto de grado se les denomina *hubs* y tienen un propósito específico en estas redes, que depende del dominio que se está modelando. Los *hubs* siguen una jerarquía donde a cada conjunto le sigue otro con un valor de grado inferior, lo que permite que estas redes presenten tolerancia a los fallos. Por ejemplo si se produjera un fallo en la red que afectara a los nodos con los valores más bajos, sería muy improbable que afectara a algún *hub*. Incluso si se produjera un fallo en algún *hub*, la red no perdería su conectividad, gracias a los *hubs* restantes. Mientras que si se seleccionan algunos de los *hubs* con mayor grado y se extraen de la red, tendría como resultado una red con grafos aislados, por lo que los *hubs* constituyen tanto la fortaleza como la debilidad de estas redes.

Otra propiedad importante de estas redes es el coeficiente de *clustering*, ya mencionado anteriormente, la cual decrece cuando el grado de los nodos se va incrementando. Esta distribución implica que nodos con un valor de grado bajo constituyen parte de subgrafos muy poblados, los cuales están conectados a otros subgrafos a través de *hubs*.

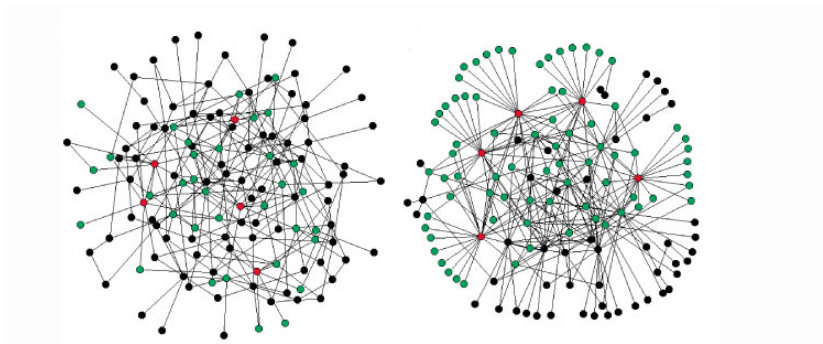


Figura 2.6: Comparativa entre una *random network* (figura de la izquierda) y *scale free network* (figura de la derecha).



En la figura 2.6 se puede observar como las aristas de la red *scale free* se distribuye de una manera no homogénea, frente al ejemplo de la izquierda de la red aleatoria. Porque las redes *scale free* cuentan con un número pequeño de nodos (coloreados en rojo en la imagen anterior) con un valor de grado muy superior al resto, estos nodos son los denominados *hub*, que sirven como puentes de unión entre los subgrafos.

Un ejemplo de redes *scale free* puede verse en Internet en las redes sociales. En este último ejemplo, la gente se conoce entre todos en grupos reducidos, lo que sería un subgrafo, pero alguna de estas personas puede conocer, por ejemplo, a una celebridad la cual está conectada a un gran número de grupos, y por lo tanto esta persona es la que actuaría como *hub* en la red *scale free*.

Para finalizar este apartado se muestra una imagen en la que se sitúa un tipo diferente de red por columna, a las que se van reduciendo el número de aristas en cada una de las sucesivas filas de la imagen.

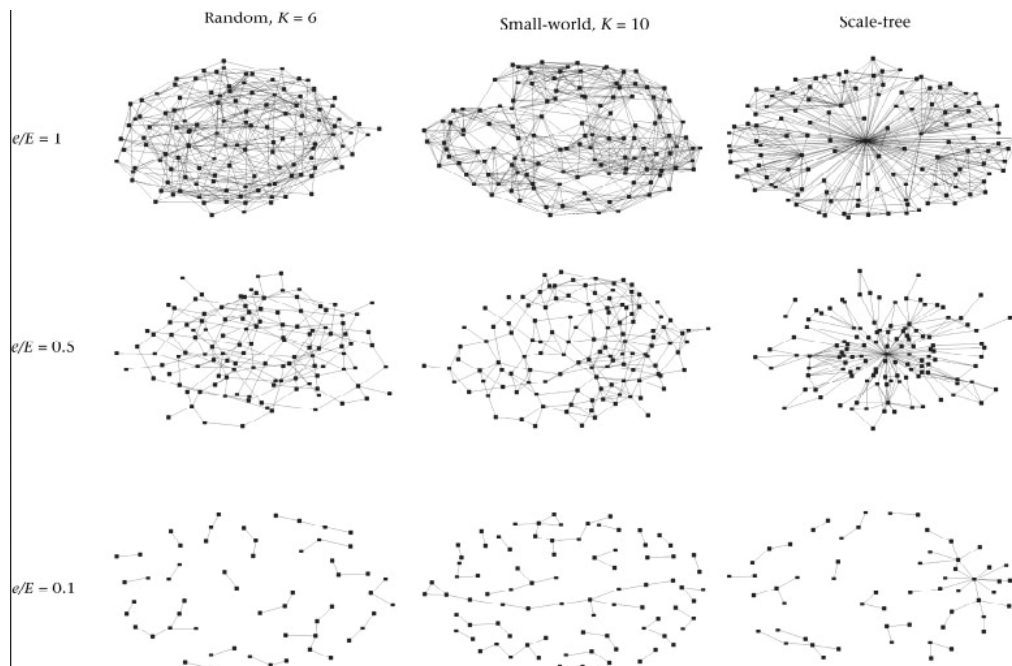


Figura 2.7: Comparativa de los diferentes tipos de redes que existen, con valor de grado  $K$  y con un porcentaje de aristas dado por la expresión  $e/E$ .

## 2.3. Métricas en las redes complejas

Cada red compleja presenta una serie de características específicas sobre su topología, las cuales caracterizan su conectividad, la dinámica y la función de los procesos ejecutados en la red.

Por tanto, el análisis de una red compleja depende en el uso de medidas capaces de expresar las características topológicas más importantes de ella y su correspondiente clasificación. A continuación, se comentan algunas de las propiedades principales de una red compleja, junto con las métricas que se pueden extraer de cada una de ellas.

### 2.3.1. Grado

En el artículo [5], se explica como el grado  $k_i$  del nodo  $i$ -ésimo es el número de aristas que están directamente conectadas a un nodo, y se define en términos de la matriz de adyacencia  $A$  como sigue:

$$k_i = \sum_{j=1}^N a_{ij} \quad (2.1)$$

donde  $N$  es el número de vértices en la red.

El grado es una característica, de la que se pueden obtener varias métricas, como por ejemplo:

- El grado máximo :

$$k_k = \max_i \{k_i\} \quad (2.2)$$

- El grado medio :

$$k_\mu = \frac{1}{N} \sum_{i=1}^N k_i \quad (2.3)$$

### 2.3.2. Joint degree

También comentado en el artículo [5], es una característica que intenta encontrar una correlación entre los grados de diferentes vértices. Esta característica juega un papel importante en muchas de las propiedades topológicas de la red. Se puede expresar como  $P(k_i, k'_i)$ , es decir la probabilidad de que

exista una arista que conecte los vértices  $i$  con grado  $k_i$ , con otro vértice de la red con grado  $k'$ .

En el artículo [5] indica que la elección de  $k'$  puede hacerse de forma arbitraria y ellos toman el caso  $k_i = k'$ , es decir la probabilidad de que un vértice tenga un vecino con el mismo grado. Pero para el proyecto se ha generalizado este caso a obtener la probabilidad de todos los posibles grados disponibles en la red con el resto. Hay varias medidas que se pueden obtener de esta característica y que enumeramos a continuación:

- La entropía, que ofrece información sobre la cantidad de orden y desorden o caos en un sistema, y se define como :

$$H = - \sum_{i=1}^N P(k_i, k')_i \log_2 P(k_i, k')_i \quad (2.4)$$

- La energía de un sistema dada por la siguiente ecuación :

$$E = \sum_{i=1}^N (P(k_i, k')_i)^2 \quad (2.5)$$

- La media :

$$\frac{1}{N} \sum_{i=1}^N P(k_i, k')_i \quad (2.6)$$

### 2.3.3. Conectividad

En el artículo [4], se habla de esta propiedad, que en esencia es similar al grado pero en este caso, se tiene en cuenta el peso de las aristas que conectan los nodos de la red.

Como en las Redes Complejas se centra en el conjunto de vértices y aristas que forman sus propiedades topológicas, en todos los estudios se da importancia a las propiedades de la conectividad y el grado ya que se obtienen a partir del número de conexiones entre vértices.

El caso concreto de la conectividad se aplica a las redes cuyas aristas tienen pesos y se calcula de la siguiente forma:

$$kw_i = \sum_{j=1}^N w_{ij} \quad (2.7)$$

donde  $w_{ij}$ , es el peso de la conexión entre los vértices  $i$  y  $j$ .

### 2.3.4. Grado Jerárquico

La jerarquía juega un papel importante en el campo de las redes complejas. El nivel jerárquico se puede definir como una dilatación del grafo  $\delta(g)$ , obteniendo un subgrafo que contiene los vértices del subgrafo  $g$  y todos aquellos vértices directamente conectados a alguno de los vértices del subgrafo.

Esta característica es iterativa, por lo que en cada paso o dilatación se van a insertar en el conjunto todos aquellos vértices que sean adyacentes a alguno de los vértices que conforma el subgrafo que se ha dilatado, contemplando que ese vértice no haya sido dilatado anteriormente.

El proceso de dilatación sobre el grafo  $g$  aplicado sucesivas  $d$  veces se muestra a continuación :

$$\delta_d(g) = \delta(\delta(\dots(g)\dots)) \quad (2.8)$$

El anillo de dilatación de un subgrafo  $g$  se define como  $R_d(g)$  y se compone de los vértices y aristas, tal como indica la siguiente ecuación :

$$R_d(g) = V(\delta_d(g)) - V(\delta_{d-1}(g)) \quad (2.9)$$

donde  $V$  es el conjunto de los vértices .

El grado jerárquico  $k_i^d$  puede definirse como el número de aristas que conectan los anillos  $R_d(i)$  y  $R_{d-1}(i)$  al vértice  $i$ , también podría realizarse el cálculo del grado jerárquico ponderado sumando los pesos de las aristas.

# Capítulo 3

## Descripción Informática

### 3.1. Metodología

Para el desarrollo del proyecto se ha seguido el modelo de proceso en espiral, el cual es adecuado en proyectos en los cuales el problema no está completamente definido al principio y conlleva ciertos riesgos como, por ejemplo, no poder encontrar el resultado esperado por el usuario, al desconocer como se comportará el programa ante determinados datos de entrada.

También es adecuado ante el caso de proyectos dinámicos en los que por necesidades del cliente, o por imposibilidad de continuar por un camino que no tiene solución, es necesario recurrir a otros caminos para alcanzar la solución al problema.

En el caso de este proyecto se han producido diversos cambios a lo largo de su desarrollo, ya que en un principio la información de entrada del sistema eran imágenes en escala de grises, pero en un cambio de rumbo del proyecto, se tomó como decisión tratar sólo imágenes en blanco y negro. También durante una fase del proyecto se estuvo tratando con redes neuronales artificiales, pero fue descartado al evaluar que los datos de clasificación aportados mediante estas redes no alcanzaban el objetivo del proyecto.

Entre las desventajas con las que nos encontramos al elegir este modelo es que requiere un mayor tiempo de desarrollo de la aplicación, que otros modelos como puede ser, por ejemplo, *Extreme Programming*, en la que el

tiempo de desarrollo es más corto. Como consecuencia del mayor tiempo de desarrollo, también aumentan los costes para su realización.

Como última gran desventaja es que es necesario de disponer de expertos capaces de predecir posibles riesgos y que sean capaces de resolver estos problemas. Aspecto resuelto ya que para el desarrollo del proyecto se ha contado con 2 tutores, expertos en los campos de la Visión Artificial y de las Matemáticas.

El modelo en espiral es un modelo híbrido, resultado de combinar el modelo en cascada y el evolutivo, fue definido por Barry Boehm en el año 1988. Es un modelo evolutivo, donde en cada vuelta en espiral va desarrollándose la aplicación. Empieza como un modelo en papel durante las primeras vueltas donde el cliente, en este caso el tutor indica las especificaciones del producto, mientras va avanzando el desarrollo, se obtiene un prototipo que es un esbozo del producto final para que el cliente conozca como evoluciona el producto. Por último en las iteraciones finales ya se obtiene la versión definitiva del producto.

En cada iteración de la espiral hay que tener en cuenta una serie de puntos, en primer lugar hay que conocer los objetivos, es decir las necesidades que el proyecto debe cubrir. En segundo lugar, para evaluar las posibles alternativas que surgen durante el desarrollo hay que tener en cuenta varios criterios, como son la experiencia de los desarrolladores, las especificaciones técnicas del equipo del cliente sobre el que se va ejecutar la aplicación y los riesgos que son posibles asumir.

Cada ciclo de la espiral representa una etapa en el desarrollo del proyecto, y un ciclo consta de 4 fases:

1. Definición de objetivos: En esta fase es esencial la comunicación con el tutor para determinar los objetivos del ciclo actual. También deben evaluarse las alternativas posibles por si alguna de ellas ofrece un resultado superior, o son menos costosas de desarrollar, debido a las restricciones con las que cuenta cada posible ruta de desarrollo.
2. Evaluación y reducción de riesgos: Estudia todos los casos en los que la solución escogida para el desarrollo puede fallar o pueden surgir eventos

inesperados. A cada uno de los riesgos identificados se intenta encontrar una alternativa y una manera de solucionarlos.

3. Desarrollo y validación: En esta fase el alumno traduce al código implementado, todos los objetivos propuestos por el profesor en las anteriores fases, teniendo en cuenta las alternativas disponible y los posibles riesgos a encontrar en cada una de ellas. Una vez terminada la codificación, es necesario reunirse con el tutor para ver si los resultados obtenidos son los esperados. En caso de obtener unos resultados adecuados se pasa a la siguiente fase.
4. Planificación: Se analiza todo lo realizado hasta ahora, para evaluar si se puede continuar con el proyecto y se hace una previsión de los recursos necesarios para la siguiente iteración.

Como principales características de este modelo es que con él es posible obtener el producto final a partir de piezas más pequeñas. El número de actividades a realizar se incrementa según el proyecto se va alejando del centro de la espiral. Se considera que la espiral tiene forma de caracola y se compone de dos dimensiones, la angular que indica el avance del proyecto dentro del ciclo, es decir, en cual de las cuatro fases se encuentra actualmente el proyecto. Mientras que la radial indica como aumenta el coste de desarrollo, ya que en esta dimensión se consideran las iteraciones o vueltas que se han realizado. También permite evaluar el progreso del proyecto al final de cada iteración para identificar si es necesario realizar cambios.

Por último indicar, como ya se ha comentado varias veces en este apartado, que es un modelo orientado a riesgos, de tal modo que uno de sus objetivos centrales es el conocer que en cualquier momento es posible que el desarrollo vaya mal y determinar como solucionarlo.

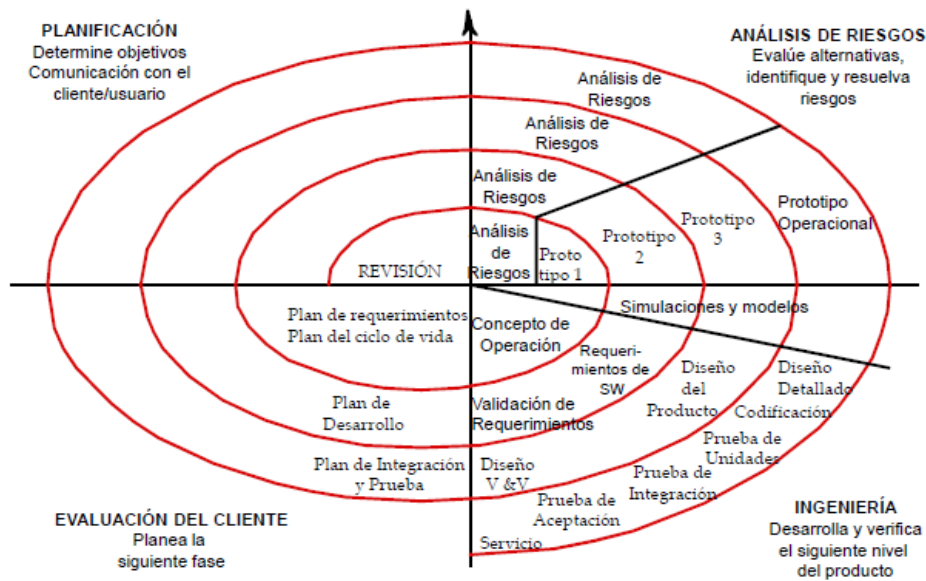


Figura 3.1: Diagrama del modelo de proceso de desarrollo en espiral.

## 3.2. Herramientas utilizadas

En las siguientes secciones se comentan las aplicaciones y herramientas utilizadas en el desarrollo del proyecto.

### 3.2.1. Matlab

Matlab cuyo nombre proviene de *MATRIX LABORATORY*, es un software matemático que ofrece un entorno de desarrollo integrado (*IDE*) a través del cual se pueden desarrollar proyectos donde se deban realizar cálculos matemáticos pesados y visualizaciones gráficas de los mismos. Destaca en el cálculo y manipulación de matrices de forma eficaz, también cuenta con representación simbólica de datos y funciones, e implementación de algoritmos entre otras muchas características.

Actualmente Matlab cuenta con numerosas herramientas adicionales como Simulink, una plataforma de simulación multidominio y GUIDE un editor de interfaces de usuario, muy completo y de fácil aprendizaje. Las capacida-



des de Matlab se ven ampliadas gracias al uso de *toolboxes*, las cuales abarcan una gran cantidad de ramas de la ingeniería y la simulación como son *nntool* para el manejo de las redes neuronales, o el *toolbox* que contiene todo lo necesario para el campo de la Visión Artificial.

Matlab fue creado por Cleve Moler en 1984 y se basa en el lenguaje de programación M desarrollado en 1970, el cual, proporciona un sencillo acceso al software de matrices LINPACK y EISPACK.

Matlab se concibió para desarrollar paquetes de subrutinas escritas en Fortran en los cursos de álgebra lineal y análisis numérico, sin necesidad de escribir programas en dicho lenguaje. Aunque Moler codificó en un principio Matlab en Fortran, actualmente está escrito en C.

Una de las mejores características que ofrece Matlab es la gran guía y tutoriales con los que aprender a utilizar la herramienta, también cuenta con una extensa comunidad de usuarios en los foros de Mathworks[12], donde se exponen y se resuelven la gran mayoría de las dudas que surgen durante el manejo del programa.

En contra tiene como principal inconveniente que no es un software libre, por lo que se debe pagar una licencia por el uso del mismo, ya que aunque dispone de una versión para estudiantes ésta no incluye todos los *toolboxes* necesarios para el proyecto.

### 3.2.2. L<sup>A</sup>T<sub>E</sub>X

Para la elaboración de la memoria se ha utilizado la herramienta L<sup>A</sup>T<sub>E</sub>X, la cual es muy utilizada como sistema de composición de una gran variedad de textos como pueden ser artículos, libros o cartas.

Está basado en el lenguaje e intérprete T<sub>E</sub>X, creado por Donald E. Knuth, su diseño tiene como objetivo producir textos de gran calidad. Está orientado a la creación de documentos científicos, ya que cuenta con una base para la representación de ecuaciones matemáticas. Alguna de sus características es que es independiente de la plataforma y del sistema operativo. Y sobre todo es un sistema *WYSIWYG* (*What You See Is What You Get*) en el que hasta que no se realiza la compilación del documento no se ve su acabado final.

$\LaTeX$ , fue creado por Leslie Lamport y consta de un conjunto de macros escritas en  $\TeX$ . Entre sus principales características destaca un acceso más sencillo que el dado por  $\TeX$ , en este caso no hay que preocuparse por detalles tipográficos. Permite aplicar al texto un tipo de presentación, es decir su estilo, describiendo su estructura lógica al principio del mismo. Y al igual que con  $\TeX$ , no se ve el aspecto del texto hasta que este es procesado, siendo necesario la utilización de algún visualizador para obtener su representación durante la redacción del documento.

A continuación, se muestra un diagrama en el que aparecen los componentes que conforman  $\LaTeX$ , junto con sus entradas y sus ficheros resultados correspondientes.

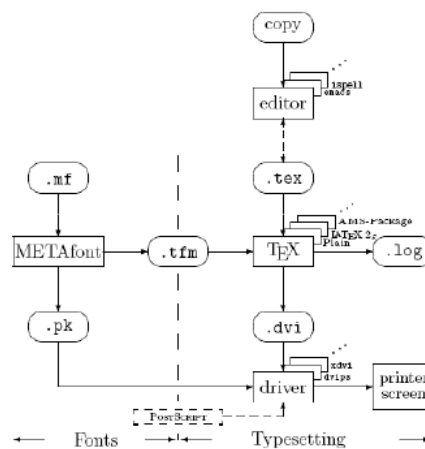


Figura 3.2: Diagrama general de los componentes involucrados en la generación de un documento usando  $\LaTeX$

### 3.2.3. Hardware y entorno de desarrollo

El hardware utilizado para el desarrollo del proyecto y las posteriores pruebas es el equipo Acer modelo Aspire 58120TG, el cual cuenta con el microprocesador *Intel Core 2 Solo Procesor SU3500* con una frecuencia de reloj de 1.4 GHz y con una memoria RAM de 4 GB DDR3. El sistema operativo utilizado ha sido la versión de 32 bits de Windows 7.

En cuanto al entorno de desarrollo del proyecto, se ha utilizado el IDE con el que cuenta Matlab (versión utilizada : R2011b) y que permite realizar la edición, depuración y compilación de *scripts*. Una de sus características más interesantes es que en tiempo de codificación realiza sugerencias de implementación, para obtener un código más eficiente.

Las sugerencias evitan casos como el de la relocalización de memoria en la que, por ejemplo, si no se indica antes de un bucle la dimensión de una matriz a la que en cada iteración se va asignando un valor al siguiente elemento de la misma, provoca que Matlab tenga que estar reservando un nuevo bloque de memoria y copiando el contenido de la matriz a este nuevo bloque.

Si no se hace caso a estos consejos, supone en la mayoría de los casos, un aumento significativo en el uso de los recursos del ordenador y por tanto, el tiempo que tarda en completarse la ejecución de las pruebas se ve también prolongado.

Finalmente para redactar la memoria se ha utilizado MikTex la distribución de  $\text{\LaTeX}$ , que corre bajo el sistema operativo Windows.

# Capítulo 4

## Métodos de clasificación

Este es el primero de los dos capítulos centrales del proyecto, en el cual se va a explicar el proceso seguido para construir el grafo y cómo se calculan las propiedades asociadas a cada una de las imágenes de la base de datos.

Si el lector se remite a la Figura 1.2 del primer capítulo, puede observarse que en el presente capítulo se va a explicar hasta la etapa etiquetada como “sintetizar la información mediante diferentes métricas de la red”, debido a que ésta y sus posteriores etapas van a ser analizadas en el próximo capítulo.

Una vez procesados todos los datos, en el próximo capítulo se procederá con su estudio para determinar cuál es el enfoque con el que se obtienen unos mejores resultados de clasificación.

### 4.1. Crear la lista de adyacencia

Para el tratamiento de las imágenes en primer lugar es necesario trasladarlas a una representación con la que poder trabajar de una forma cómoda y eficiente.

Por ello en un primer momento se barajó la idea de utilizar matrices de adyacencias, dado el potencial que posee Matlab para trabajar con matrices, pero se descartó tras comprobar las limitaciones para su representación. Debido a que ya con imágenes de dimensiones reducidas no era posible obtener

la matriz porque el número de elementos de la misma se salía del rango de dimensiones aceptadas por Matlab.

Por tanto, la solución consiste en crear una lista de adyacencia con tantos elementos o nodos como número de píxeles tiene la imagen. Para poder realizar un buen procesamiento de la imagen cada nodo de la lista contiene información adicional, entre otras, cuenta con las coordenadas del píxel con respecto al origen que se encuentra situado en el extremo superior izquierdo de la imagen y el valor de intensidad del píxel, la implementación de esta estructura se encuentra en el fichero `listaAdyacencia.m`.

También contiene un vector de registros, donde almacena todos los nodos o píxeles que son sus vecinos o adyacentes. Los píxeles adyacentes se consideran todos aquellos contenidos en un radio  $r$  proporcionado por el usuario, siendo el centro del radio el píxel que se está tratando actualmente. En un principio se pensó en tomar como vecinos aquéllos cuya distancia euclídea fuera inferior al radio dado, pero debido a que los cálculos para obtener estos adyacentes provocaban un aumento considerable en el tiempo de construcción de la lista se optó por seleccionar como adyacentes todos aquellos píxeles que se encuentren dentro de esa ventana cuyos lados tienen dimensión  $r$ .

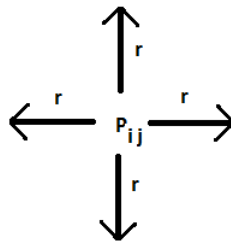


Figura 4.1: Para un píxel  $p_{ij}$  sus vecinos son aquellos comprendidos en la ventana cuyos lados tienen tamaño  $r$ , tomando a  $p$  como centro

De los píxeles adyacentes también se almacena la siguiente información: sus coordenadas y el peso de conexión entre ellos. En un principio con la idea expuesta en el artículo [4], la ecuación para obtener los vecinos y los pesos

de conexiones correspondientes fue la siguiente :

$$w_{ij} = \begin{cases} |I(p_i) - I(p_j)| & \text{si } dist(p_i, p_j) \leq r \\ 0 & \text{si } dist(p_i, p_j) > r \end{cases} \quad (4.1)$$

Pero debido al tiempo elevado de cómputo para crear la lista de adyacencia, finalmente el peso de conexión entre píxeles viene dado por la diferencia en valor absoluto de las intensidades de los dos píxeles.

La diferencia se calcula para todos los píxeles contenidos en el radio actual, entre los cuales van a tener conexión, cuando la arista cuente con un peso de valor 1, siendo el único caso en el que los nodos adyacentes sean de intensidades diferentes.

Otro aspecto a tener en cuenta es que el radio no va a ser el mismo para todos aquellos píxeles situados en el marco externo de la imagen, para ello debe controlarse en todo momento que para cada píxel no se salga de los límites de la imagen, mientras se calculan sus adyacentes. Para el cálculo de la posición del píxel correspondiente en la lista de adyacencia, debido a que es un vector, es necesario obtener su índice en la secuencia a partir de los valores de las coordenadas del píxel, mediante la siguiente ecuación :

$$pos = (i - 1)nc + j \quad (4.2)$$

donde  $i$  es el valor de la fila del píxel si consideramos la imagen como una matriz, o el valor del eje de las ordenadas en el caso del plano cartesiano,  $j$  se corresponde con el número de columna o el eje de las abcisas, por último  $nc$  es el valor del número de columnas en la imagen.

La última propiedad que almacena cada elemento de la lista de adyacencia es el grado del píxel, característica que se calcula posteriormente, mediante la llamada a la función implementado para ello y comentada en el siguiente apartado.

Para concluir con lo explicado anteriormente, se muestra a continuación un pseudocódigo general de los pasos seguidos para construir la lista de adyacencia.

**Data:** Matriz con los valores de intensidades de los píxeles de la imagen, radio de la vecindad

**Result:** Lista de adyacencia con la información de grado y píxeles adyacentes de cada uno de los píxeles de la imagen

```

for  $i = 1$  hasta  $i =$ cantidad total de filas do
  for  $j = 1$  hasta  $j =$ cantidad total de columnas do
    Obtener índice inicial y final de las filas de la ventana de
    vecindad a partir de la posición  $i$ ;
    Obtener índice inicial y final de las columnas de la ventana de
    vecindad a partir de la posición  $j$ ;
    Insertar el registro en la posición correspondiente en la lista;
    for inicio ventana vecindad hasta final ventana do
      if no es el pixel actual then
        Calcular diferencia de intensidades entre píxeles;
        Insertar píxel en la lista de píxeles adyacentes del píxel
        actual;
      end
    end
    Reordenar la lista de píxeles adyacentes;
  end
end

```

**Algorithm 1:** Pseudocódigo correspondiente a la construcción de la lista de adyacencia

El tiempo que necesita el algoritmo para generar la lista de adyacencia de una imagen de la base de datos de un tamaño de  $107 * 115$  y utilizando un valor de radio de 4, es de 58 segundos.

Por lo tanto el tiempo medio para conseguir representar como una lista de adyacencia todas las imágenes de la base de datos en el ordenador comentado en el apartado correspondiente es de 3 horas y 12 minutos.

## 4.2. Cálculo de la característica de grado

El algoritmo que calcula esta propiedad se encuentra en el fichero `calcularGrado.m`. El aspecto teórico de esta propiedad ya fue comentado en el capítulo 2, en el que de forma resumida el grado es el número de píxeles que están conectados mediante aristas al píxel del que se está calculando esta

propiedad.

Por tanto la forma de implementarlo ha tenido como objetivo hacerlo de la forma más sencilla y eficientemente posible, para ello el algoritmo recorre toda la lista de adyacencia calculando el grado para cada píxel.

Para calcular el grado de cada píxel, lo primero que hace es extraer en un vector el valor del peso de las conexiones con todos los píxeles de la ventana  $r * r$ , explicada anteriormente, donde  $r$  es el valor de radio. A continuación, suma los datos de este vector, dentro del cual sólo va a contener 2 tipos de valores, el valor 0 caso en el que no están conectados y el valor 1, caso en el que esos píxeles son adyacentes. El valor de ese sumatorio es el valor del grado de cada píxel. El tiempo de ejecución para obtener el grado de todos los píxeles de una imagen es de un segundo y medio, tomando como máximo el valor 4 para el radio.

Como en el apartado anterior, se muestra a continuación un pseudocódigo para conocer de forma más directa el funcionamiento del algoritmo.

**Data:** Lista de adyacencia y de forma opcional un vector que indica para cada píxel la posición del último píxel considerado como adyacente para una escala dada

**Result:** Lista de adyacencia con la información de grado inicializada

```

for  $i = 1$  hasta  $i = total$  elementos de la lista do
  |  $lista(i).grado =$  sumatorio de los pesos de todas las conexiones con
  | los vecinos del nodo  $i$ ;
end

```

**Algorithm 2:** Pseudocódigo para el cálculo de la característica de grado

Una vez calculado el grado para todos los píxeles de la imagen, es necesario obtener su histograma. Para ello en primer lugar es necesario conocer cuál es el rango de valores de grado con el que se va a trabajar, así después de calcular el grado para todas las imágenes se observó que los valores de grado para todas las muestras, comprenden el intervalo entre 1 y 74. El rango de valores para el grado se ha obtenido tras calcular el grado máximo entre todas las imágenes de la BBDD, y el valor máximo depende directamente del radio de la ventana de vecindad, siendo posible con una ventana de  $r = 4$  obtener un valor máximo de 80 ( $9 * 9$  menos el píxel central), si es el caso extremo de un píxel negro y el resto de los contenidos en la ventana son



píxeles blancos. De esta manera el eje de las abcisas del histograma de todas las imágenes tendrá el mismo rango para un posterior análisis.

Para construir el histograma se obtiene para cada valor de grado la cantidad de píxeles con ese grado, haciendo uso de la función `find` sobre un vector en el que se ha recuperado el valor de grado de todos los píxeles. Posteriormente se aplica la función `length` al vector resultado anterior que contiene las posiciones sobre el vector original de aquellos píxeles con el valor buscado para obtener su cantidad.

Este proceso se repite para los 73 grados restantes. Una vez obtenidas las cantidades para cada grado hay que normalizar el vector y para ello se debe dividir cada valor por la cantidad total de píxeles con grado distinto de cero. De esta forma ahora cada valor comprende en el rango entre 0 y 1, siendo la suma de todos los elementos igual a 1.

La implementación del algoritmo para calcular el histograma se encuentra en el fichero `histogramaGrados.m`. El pseudocódigo correspondiente es el siguiente:

**Data:** Lista de adyacencia

**Result:** Vector normalizado con el porcentaje píxeles para cada uno de los posibles grados

totalNodos=número de píxeles con grado distinto de cero ;

**for**  $i = 1$  hasta  $i = gradoMaximo$  **do**

    | ocurrencias= la cantidad de píxeles de grado  $i$ ;  
    |  $resultado(i) = \frac{ocurrencias}{totalNodos}$ ;

**end**

**Algorithm 3:** Pseudocódigo para calcular el histograma referente a la propiedad grados

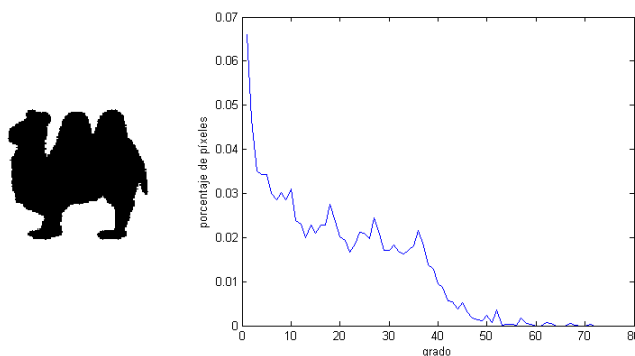


Figura 4.2: Figura en la que se muestra el histograma para la característica grado asociada a una imagen de la Base de Datos. En la gráfica el eje de abcisas se refiere al valor de grado y el eje de ordenadas se corresponde al porcentaje de píxeles de la imagen que poseen dicho grado

### 4.3. Cálculo de la circularidad

Esta característica se encuentra implementada en el fichero `contorno.m`. Para el cálculo de la circularidad, el código se basa en una variación del algoritmo de *Bresenham* conocido como *midpoint circle algorithm*. Una forma inicial de entender esta propiedad, es la dada por la siguiente imagen.



Figura 4.3: representación de las circunferencias que se trazan sobre una imagen para conocer la propiedad de circularidad de la imagen

Para calcular la circularidad hay que tomar un punto de inicio de los anillos que se van a ir trazando en cada iteración, este origen va a ser el centro

de masas del objeto de la imagen, función implementada en `calcularCentroMasas.m`. Para determinar el centro de masas, hay que dividir el proceso en 2 partes. En la primera hay que calcular la coordenada  $x$  y la coordenada  $y$  del centro de masas.

Para calcular cada una de estas coordenadas, hay que obtener los valores de las coordenadas  $x$  de todos los píxeles pertenecientes al objeto de la imagen y una vez calculada la suma de todos esos valores debe dividirse por el número total de píxeles pertenecientes al objeto, de esta forma ya se ha obtenido una de las dos coordenadas del centro de masas. Para la otra coordenada basta con repetir el mismo proceso.

Un aspecto a tener en cuenta durante el cálculo de las coordenadas es que en Matlab se tratan de forma contraria al plano cartesiano, ya que un subíndice de fila se corresponde con las ordenadas y las columnas con el eje de abscisas.

La segunda parte del proceso no es necesaria si el centro de masas obtenido corresponde con un píxel del objeto de la imagen. Pero si el centro de masas resultante no pertenece a un píxel del objeto hay que recorrer la lista a partir de la posición del centro obtenido en ambas direcciones, buscando cual de ellas es la más cercana a un píxel de objeto al centro calculado previamente y se considera esa posición como el centro definitivo.

A continuación se muestra el pseudocódigo perteneciente al cálculo del centro de masas:

**Data:** Lista de adyacencia, número total de columnas de la matriz correspondiente a las intensidades

**Result:** Las coordenadas del centro de masas de la imagen  
*pixelesObjeto* = Obtener el número de píxeles de la imagen correspondientes al objeto;  
*coordenadaX* = sumatorio de la componente *j* de los elementos de la lista *pixelesObjeto*;  

$$centroX = \frac{coordenadaX}{cantidadPixelesObjeto};$$
*coordenadaY* = sumatorio de la componente *i* de los elementos de la lista *pixelesObjeto*;  

$$centroY = \frac{coordenadaY}{cantidadPixelesObjeto};$$
*indice* = (*CentroY* - 1)*totalColumnas* + *CentroX*;  
**if** *lista(centroY, centroX) ≠ pixelObjeto* **then**  
    *indice1 = indice*;  
    **while** *intensidad de lista(indice1) ≠ 0* **do**  
        *indice1 = indice + 1*;  
        *distancia1 = distancia + 1*;  
    **end**  
    *indice2 = indice*;  
    **while** *intensidad de lista(indice2) ≠ 0* **do**  
        *indice2 = indice - 1*;  
        *distancia2 = distancia - 1*;  
    **end**  
    **if** *distancia1 < distancia2* **then**  
        *centroX = lista(indice1).j*;  
        *centroY = lista(indice1).i*;  
    **end**  
    **else**  
        *centroX = lista(indice2).j*;  
        *centroY = lista(indice2).i*;  
    **end**  
**end**

**Algorithm 4:** Pseudocódigo perteneciente al cálculo del centro de masas de una imagen

Una vez calculado el centro de masas a partir de él se van a ir trazando hasta  $c = 20$  circunferencias con un radio que se va incrementando por un valor constante calculado a partir de las dimensiones de la imagen.

Para cada iteración se va a invocar a una función que implementa el algoritmo de *Bresenham* aplicado a las circunferencias, el cual necesita principalmente las coordenadas del centro de masas y el valor del radio de la circunferencia.

A continuación se va a resumir el funcionamiento del algoritmo de *Bresenham* aplicado a las circunferencias, debido a que éste es una variación del algoritmo de aproximación de rectas. La idea principal es la de aproximar los puntos en los que hay mayor probabilidad de que la circunferencia pase por ellos.

Este algoritmo también es conocido como el algoritmo del punto medio de la circunferencia (*midpoint circle algorithm*). En el que la función de la circunferencia se define como:

$$x^2 + y^2 = r^2 \quad (4.3)$$

El algoritmo comienza considerando sólo el primer octante y dibuja una curva que comienza en el punto  $(r, 0)$  y continúa trazándola en sentido horario hasta alcanzar un ángulo de 45 grados.

Este algoritmo se denomina incremental, porque la posición del siguiente píxel se calcula a partir del obtenido anteriormente, en vez de calcular todos los puntos a partir de una ecuación general, esta lógica tiene como ventaja la disminución del tiempo de cómputo.

La propiedad de simetría de las circunferencias es muy ventajosa para este algoritmo. Primero como la circunferencia es simétrica respecto al eje de las  $x$ , sólo es necesario calcular los primeros 180 grados, de forma análoga respecto al eje  $y$ , sólo es necesario calcular las coordenadas para los primeros 90 grados. Por último, como la circunferencia es simétrica al eje con una inclinación de 45 grados, basta con calcular la posición de los píxeles de los primeros 45 grados de la circunferencia en sentido antihorario.

El algoritmo asume que la circunferencia se encuentra colocada en el origen, pero en el proyecto se toma como origen las coordenadas del centro de masas calculado previamente. Como se comentó previamente, el algoritmo

sólo obtiene los píxeles de la circunferencia correspondiente a los primeros 45 grados de la misma. Por tanto, para cada posición  $(x, y)$  calculada es necesario dibujar el píxel en cada uno de los octantes en los que se ha dividido el plano, por tanto las ocho coordenadas serían las siguientes:

- $(centroX + x, centroY + y)$
  - $(centroX + x, centroY - y)$
  - $(centroX - x, centroY + y)$
  - $(centroX - x, centroY - y)$
- $(centroX + y, centroY + x)$
  - $(centroX + y, centroY - x)$
  - $(centroX - y, centroY + x)$
  - $(centroX - y, centroY - x)$

Donde  $centroX$  y  $centroY$  se corresponden con las coordenadas del centro de masas calculado previamente.

Si se considera el arco de puntos continuos calculado por el algoritmo que se muestra en la siguiente imagen, es necesario discretizarlo a las coordenadas de los píxeles de la figura que se está tratando.

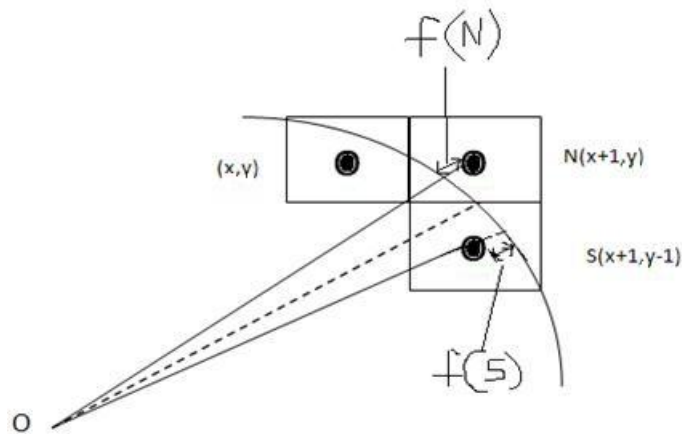


Figura 4.4: Posibles aproximaciones a partir del píxel  $(x, y)$  calculado por el algoritmo

La manera en la que se discretiza las posiciones obtenidas del arco es mediante una aproximación seleccionando los píxeles que van a ser dibujados.

A cualquier punto  $(x, y)$  obtenido, sólo hay disponibles dos opciones, elegir el píxel situado al este del punto considerado, es decir, el punto  $N(x + 1, y)$ , o por el contrario seleccionar el punto situado en las coordenadas  $S(x + 1, y - 1)$ .

La elección del punto es determinado por aquella opción que tenga un error de aproximación menor. Tal como aparece en la figura 4.4 el error viene dado por el valor de  $f\{N\}$  y  $f\{S\}$ , siendo la distancia de cada uno de los dos puntos respecto al punto real de la circunferencia. Las funciones de error son las siguientes:

$$f\{N\} = (x_i + 1)^2 + y_i^2 - r^2 \quad (4.4)$$

$$f\{S\} = (x_i + 1)^2 + (y_i - 1)^2 - r^2 \quad (4.5)$$

Una vez definidas ambas funciones de error, es necesario establecer el parámetro de decisión siguiente:

$$d_i = f\{N\} + f\{S\} \quad (4.6)$$

Si este parámetro es inferior o igual a cero se toma como el nuevo píxel calculado el dado por  $N(x + 1, y)$ , en cambio si es superior a cero el siguiente píxel es  $S(x + 1, y - 1)$ .

Pero para conocer el valor de este parámetro de decisión para la iteración  $i$  es necesario conocer cuál es su valor para el caso base. Para ello es suficiente con sustituir el valor de  $x$  por cero y el valor de la coordenada  $y$  por el valor del radio, a la función anterior, a la que sólo que se ha expandido y reagrupado el contenido de  $f\{N\}$  y  $f\{S\}$ :

$$d_i = 2(x_i + 1)^2 + y_i^2 + (y_i - 1)^2 - 2r^2 \quad (4.7)$$

Obteniendo como resultado de sustituir los valores iniciales de  $x$  e  $y$  la siguiente función, que obtiene el parámetro de decisión inicial:

$$d_0 = 2 + r^2 + (r - 1)^2 - 2r^2 \quad (4.8)$$

Y simplificando la ecuación anterior, se consigue:

$$d_0 = 3 - 2r \quad (4.9)$$

El último caso a considerar con respecto al parámetro de decisión es el proceso para calcularlo para la siguiente iteración. Se parte de la ecuación:

$$d_{i+1} = 2(x_i + 2)^2 + y_{i+1} + (y_{i+1} - 1)^2 - 2r^2 \quad (4.10)$$

A continuación se calcula la diferencia entre las ecuaciones 4.10 y 4.7, como se muestra a continuación:

$$d_{i+1} - d_i = 2((x_i + 2)^2 - (x_i + 1)^2) + (y_{i+1}^2 - y_i^2) + ((y_{i+1} - 1)^2 - (y_i - 1)^2) \quad (4.11)$$

Desarrollando la anterior ecuación se obtiene como resultado:

$$d_{i+1} = d_i + 2(2x_i + 3) + ((y_{i+1} + y_i)(y_{i+1} - y_i)(y_{i+1} - 1 + y_i + 1)) \quad (4.12)$$

Con la ecuación desarrollada hasta ahora hay que tener en cuenta el valor del parámetro de decisión previo para los valores de las coordenadas  $x$  e  $y$ .

$$\begin{cases} x_{i+1} = x_i + 1, y_{i+1} = y_i & \text{Si } (d_i \leq 0) \\ x_{i+1} = x_i + 1, y_{i+1} = y_i - 1 & \text{Si } (d_i > 0) \end{cases} \quad (4.13)$$

En el caso en que  $d_i$  es igual o inferior a cero, el siguiente parámetro de decisión se obtiene sustituyendo en 4.12 los valores de las coordenadas  $x$  e  $y$  correspondientes a este caso:

$$\begin{aligned} d_{i+1} = d_i + & 2(2x_i + 3) + (y_{i+1} + y_i)(y_{i+1} - y_i) + \\ & + (y_{i+1} - 1 + y_i - 1)(y_{i+1} - 1 - y_i + 1) \end{aligned} \quad (4.14)$$

Como resultado del desarrollo para este primer caso, la función de decisión para la siguiente iteración es:

$$d_{i+1} = d_i + 4x_i + 6 \quad (4.15)$$

Y el caso en que el valor de  $d_i$  sea superior a cero deben tomarse los valores de  $x$  e  $y$  correspondientes al segundo caso, para sustituirlos en la ecuación 4.12:

$$\begin{aligned} d_{i+1} = d_i + & 2(2x_i + 3) + ((y_i - 1 + y_i)(y_i - 1 - y_i) + \\ & + ((y_i - 2 + y_i - 1)(y_i - 2 - y_i + 1)) \end{aligned} \quad (4.16)$$

Una vez desarrollada la ecuación se obtiene la ecuación para el siguiente paso:



$$d_{i+1} = d_i + 4(x_i - y_i) + 10 \quad (4.17)$$

Para concluir con la explicación del algoritmo *midpoint circle*, se muestra un pseudocódigo que resume los principales pasos del mismo.

**Data:** Lista de adyacencia, las coordenadas del centro de masas, el radio de la circunferencia y el número total de columnas de la matriz correspondiente a las intensidades

**Result:** El ratio de píxeles que pertenecen al objeto en la circunferencia calculada

$x = 0;$

$y = \text{radio};$

El punto  $(x, y)$  se toma como el comienzo del algoritmo;

$dI = 3 - 2r;$

Calcular el punto de cada octante a partir de los valores de  $x$  e  $y$ ;

**while**  $x < y$  **do**

**if**  $dI \leq 0$  **then**

$d = d + (4x + 6);$

**end**

**else**

$d = d + 4(x - y) + 10;$

$y = y - 1;$

**end**

$x = x + 1;$

    Calcular el punto de cada octante a partir de los valores de  $x$  e  $y$ ;

**end**

**Algorithm 5:** Pseudocódigo del algoritmo *midpoint circle*

En el pseudocódigo las líneas en las que se calculan los puntos a cada octante, hacen referencia a la lista de 8 coordenadas enumeradas anteriormente cuando se hacía referencia a la propiedad de simetría de la circunferencia.

Una vez descritos los aspectos teóricos del algoritmo, se van a enumerar los pasos que sigue su implementación, junto con las modificaciones necesarias para obtener los datos necesarios, para la propiedad de circularidad necesaria para el proyecto.

En primer lugar se pasa a la función (contenida en el fichero *Bresenha-*

mImplementacion.m) a parte de otros parámetros, el radio  $r$  y el centro de masas, que va a actuar como centro de la circunferencia.

La implementación del algoritmo, es una transcripción del pseudocódigo que se ha desarrollado previamente. El algoritmo se desarrolla de igual forma hasta el momento en el que se calculan los puntos pertenecientes a cada octante a partir de las coordenadas calculadas por el algoritmo de *Bresenham*, ya que deben tomarse una serie de medidas respecto a estos puntos, para conseguir un histograma correcto sin información redundante.

Para cada uno de estos puntos calculados, primero debe validarse si no se han procesado anteriormente, ya que hay que evitar duplicar resultados. Una vez terminado este paso, hay que comprobar a qué zona de la imagen pertenecen, incrementando el contador de coincidencias sólo en el caso de encontrar un píxel que pertenezca a la figura de la imagen. También se tienen en cuenta aquellos píxeles que se encuentren en el límite del contorno de la figura. Para comprobar si uno de los píxeles aproximados por la función pertenecen al contorno, debe llamarse a la función contenida en el fichero *esBorde.m*, cuya funcionalidad consiste en determinar si pertenece o no al contorno.

Esta función comprueba el valor de intensidad de todos los píxeles adyacentes y en caso de que alguno tenga una intensidad de 1 y el píxel a comprobar sea negro, es decir con intensidad cero, se tiene una ocurrencia de píxel contorno.

Por último se divide el número de coincidencias entre el número total de píxeles de la circunferencia trazada para obtener el porcentaje de píxeles contorno encontrados con ese radio de la circunferencia.

Este proceso de aplicar el algoritmo del *midpoint circle* se repite, para tomar las 20 muestras con valor de radio creciente, para obtener el histograma correspondiente a esta característica, para una imagen concreta. Los valores del radio se han seleccionado teniendo en cuenta las dimensiones de las imágenes de entrada y, la posición de los objetos de cada imagen en la que suelen estar colocados en el centro. Por tanto teniendo en cuenta que a un radio muy pequeño el porcentaje de píxeles de la circunferencia aplicada sobre la imagen va a ser cercano al 100 % mientras que a circunferencias de

radio muy grande, el porcentaje perteneciente al objeto va a ser muy bajo. Por tanto los rangos de valores del radio de la circunferencia abarca desde 12 píxeles hasta un radio de 64, el valor de radio aumenta siguiendo la fórmula  $\frac{(\text{radioInicial}-\text{radioFinal})}{\text{numeroMuestras}}$ , siendo en el caso estudiado un valor de incremento en cada iteración de 2.6 píxeles.

Como paso final, para obtener un histograma normalizado, es necesario dividir cada uno de los elementos de este vector por la suma de todos sus valores para que el rango de valores comprenda entre 0 y 1.

El tiempo necesario por el algoritmo implementado para obtener el histograma es de aproximadamente 2 segundos, para la misma imagen ejemplo de la lista de adyacencia. El histograma que se genera para esa imagen es el siguiente:

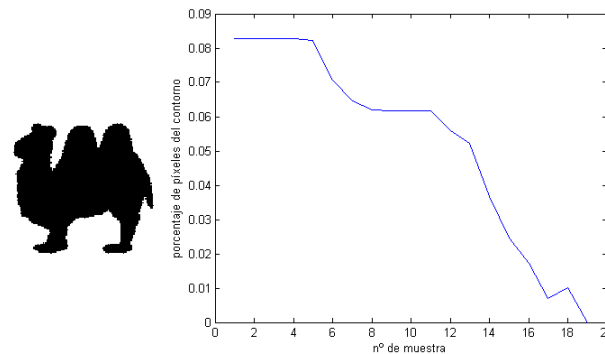


Figura 4.5: Imagen junto su histograma para la característica de circularidad obtenida a partir del vector de 20 características. El eje de abscisas se corresponde con el identificador de la muestra tomada y, el eje de las ordenadas representa al porcentaje de píxeles del objeto contenido en la circunferencia calculada por el algoritmo

#### 4.4. Cálculo de la propiedad Joint degree probability

La implementación de esta característica se encuentra en el fichero `joint-Vectorizado.m`. La idea central de esta característica descrita en el capítulo

2, es la de obtener el porcentaje de probabilidad de encontrar píxeles de un determinado grado  $i$  estén conectados a píxeles de un grado  $j$ .

Teniendo en cuenta el dato aportado en el apartado anterior del cálculo de la característica de grado, en el que después de calcularlo para todas las imágenes de la base de datos, el valor de grado máximo obtenido es 74.

Junto con el valor anterior, también hay que considerar que el grafo construido no es dirigido por lo que la probabilidad de  $(i, j)$  es el mismo caso que el de la probabilidad de grados  $(j, i)$ , entonces el número de casos es el siguiente:

$$nCAsos = \frac{(\text{máx}(g)(\text{máx}(g) + 1))}{2} \quad (4.18)$$

donde  $\text{máx}(g)$  es el valor del grado máximo siendo en el caso de la base de datos utilizada. el valor de grado máximo igual a 74.

Aunque la información de probabilidad se va a almacenar también en un vector de registros con los campos de los valores de los dos grados y su porcentaje de probabilidad, la estructura que realmente se va a utilizar en pasos posteriores del proyecto es un vector en el que se van a almacenar de forma secuencial todos los casos posibles, es decir  $\frac{\text{gradoMaximo}(\text{gradoMaximo}+1)}{2}$  y como en este proyecto el *gradoMaximo* se ha establecido con valor 74, deben analizarse un total de 2775 casos.

A la hora de implementar esta característica hay que tener en cuenta dos casos: la probabilidad de conexión entre grados diferentes y la probabilidad de conexión de píxeles con el mismo grado.

En el primer caso, debe recuperarse inicialmente en un vector la posición de todos aquellos píxeles que tengan valor de grado  $i$ . A partir de esas posiciones se recupera de la lista de adyacencia, todos aquellos píxeles adyacentes de cada píxel grado  $i$ . A continuación, es necesario conocer su posición en la lista de adyacencia, para recuperar el valor de grado de cada uno de ellos. Para ello se hace uso de la función ya referida en ocasiones anteriores:

$$pos = (i - 1)nc + j \quad (4.19)$$

Una vez calculadas sus posiciones, se recupera en un vector los grados de todos los píxeles, y para conocer el número de ocurrencias del caso actual

basta con hacer uso de la función `find` del grado  $j$  sobre el vector de grados. Del resultado dado por `find` basta con utilizarlo como argumento de la función `length` para conocer ese número de ocurrencias, tanto la función `length` como `find` son funciones predefinidas en Matlab. Para conocer el valor de probabilidad es necesario utilizar la siguiente ecuación :

$$prob(i, j) = \frac{n}{p_i p_j} \quad (4.20)$$

donde  $n$  es el número de ocurrencias del caso  $(i, j)$ ,  $p_i$  y  $p_j$  son el número de píxeles con grado  $i$  y grado  $j$  respectivamente. Una vez calculada esta probabilidad se almacena en la posición correspondiente del vector resultado.

Para el caso en el que el grado de ambos píxeles debe ser igual, hay que tener en cuenta una consideración importante. En este caso hay que evitar que se contabilice en más de una ocasión una ocurrencia. Porque al recuperar los adyacentes a un píxel, se obtendrá alguno del grado buscado pero para el cual ya se procesaron todas sus conexiones, incluida la del píxel actual. Por ello hay que controlar los píxeles ya procesados mediante un vector binario en el que se vayan marcando a uno todos aquellos píxeles ya procesados.

Al contrario que en el caso anterior en el que se podía procesar de una pasada todos los píxeles de grado  $i$ , ahora a partir del vector de píxeles de grado  $i$ , hay que ir obteniendo píxel a píxel sus adyacentes y comprobar que no hayan sido procesados. Una vez hecha esta comprobación se sigue el proceso de incrementar el número de ocurrencias por aquellos píxeles de grado  $i$  no procesados y se marcan como contabilizados. Por último se utiliza la misma ecuación que en el caso de la circularidad para obtener el porcentaje de este caso.

A continuación se muestra el pseudocódigo correspondiente al algoritmo que calcula la distribución *joint degree*.

**Data:** Lista de adyacencia

**Result:** Un vector de structs con la información de cada caso y un vector normalizado con la probabilidad de ocurrencia de cada uno de los posibles casos

Obtener el número de píxeles de cada grado disponibles en la imagen;

```

for  $i = 1$  hasta  $i = gradoMaximo$  do
  for  $j = i$  hasta  $j = gradoMaximo$  do
    if  $i \neq j$  then
      Recuperar píxeles adyacentes a todos los píxeles de grado  $i$ ;
      Contabilizar el total de píxeles adyacentes con grado de
      valor  $j$ ;
    end
    else
       $ocurrencias = 0$ ;
      for  $k = 1$  hasta  $k = totalPíxelesGradoI$  do
        coincidencias=adyacentes al píxel  $k$  con grado  $i$  no
        marcados;
         $ocurrencias = ocurrencias + coincidencias$ ;
        Marcar el píxel  $k$  como procesado;
      end
    end
     $caso(i, j) = \frac{ocurrencias}{(totalPíxelesGradoI * totalPíxelesGradoJ)}$ ;
  end
end
normalizar vector resultado;

```

**Algorithm 6:** Pseudocódigo que calcula la distribución de probabilidades *joint degree*

Una vez que se ha obtenido el porcentaje para todos los casos hay que normalizar el histograma para ello hay que seguir el mismo procedimiento que en el caso de la propiedad de la circularidad en la que hay que dividir cada uno de los porcentajes por el valor de la suma de todos los valores del vector, de esta forma todos los casos se comprenden entre los valores 0 y 1.

El tiempo necesario para obtener el histograma para la misma imagen de apartados anteriores es de 86 segundos. Se obtiene la siguiente gráfica con la función  $\text{semilog}(y)$  de Matlab, para utilizar una escala logarítmica en el eje de las ordenadas y apreciar mejor el resultado, de la propiedad *joint degree*

para la misma imagen ejemplo que se ha utilizado a lo largo de la descripción:

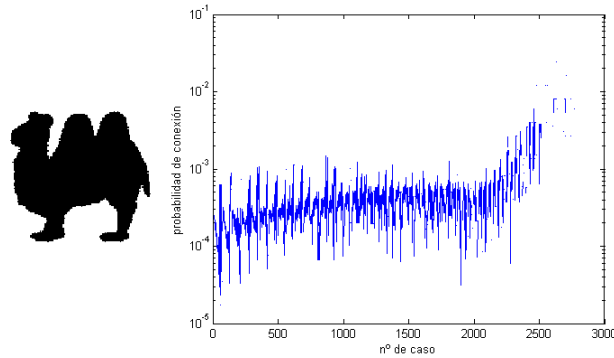


Figura 4.6: Figura de la BBDD junto con su histograma obtenido a partir del vector de la característica joint degree. Donde en el eje de las abcisas se enumeran los 2775 casos asociados al valor de la probabilidad de cada ocurrencia correspondiente en el eje de las ordenadas

## 4.5. Cálculo del ratio de Fisher

El *ratio de Fisher* o también conocido como *F-test*, es utilizado en la mayoría de los casos en los que se realizan comparaciones entre modelos estadísticos que han sido preparados para un conjunto de datos de entrada, cuyo objetivo es identificar cuál de los modelos es el que mejor se adapta a la clase de la cual procede la muestra que se está analizando.

La mayoría de los *F-tests* surgen debido a que tienen en cuenta la descomposición de la variabilidad en una colección de datos en términos de la suma de sus cuadrados. Esta medida estadística, proporciona el ratio de dos sumas escaladas de cuadrados que indican diferentes fuentes de variabilidad. La suma de cuadrados son creadas de tal forma que los resultados tienden a ser mayores cuando la hipótesis de nulidad no es cierta, siendo ésta una hipótesis cuyo objetivo es el descarte para apoyar otras alternativas.

Para que el *F-test* cumpla con la hipótesis nula, es necesario que la suma de los cuadrados sean independientes y los datos de entrada sean independientes y distribuidos con una varianza equitativa.

A continuación se muestra la ecuación del ratio de Fisher en la cual se está evaluando la diferencia o varianza entre la media de una clase respecto a la media global, frente a la varianza de una muestra concreta de una clase respecto a la media de su clase.

$$\frac{\sum_{i=1}^C \|m_i - m\|^2}{\sum_{i=1}^C \frac{1}{n_i} \sum_{x \in C_i} \|x - m_i\|^2} \quad (4.21)$$

Donde  $m_i$  es la media de todos los histogramas de la clase  $i$ ,  $m$  es la media de todos los histogramas de todas las clases analizadas y  $x$  se refiere al vector de características de una figura concreta de una clase. Por lo que el numerador contempla el caso de la variabilidad entre clases, mientras que el denominador analiza la variabilidad de las imágenes pertenecientes a la misma clase.

De forma general, el pseudocódigo para el cálculo de esta distribución es el siguiente:

**Data:** La matriz con los valores de distancia entre cada par de imágenes por cada característica

**Result:** La distribución de pesos para cada una de las características estudiadas

Calcular el vector media para cada característica;

**for**  $i = 1$  *hasta*  $i = totalClaseImágenes$  **do**

    | Obtener el vector media de las 3 características para la clase actual;

**end**

Aplicar la ecuación del ratio de Fisher con los datos obtenidos;

**Algorithm 7:** Pseudocódigo que calcula la distribución de pesos según el ratio de Fisher

El código implementado está contenido en el fichero `calcularFisher.m`, que incluye la función `calcularMedia` encargada de obtener la media aritmética de los valores de distancia para todas las imágenes de una clase, que en el caso de este proyecto los valores de distancia de las imágenes de cada clase se corresponde con un bloque de dimensiones de 12\*12 situado en la diagonal de la matriz, ya que hay 12 imágenes de cada clase en la BBDD utilizada y éstas se encuentran ordenadas en la matriz, es decir sin intercalar imágenes de otras



clases. La otra función de la que consta el fichero es la denominada **funcion-Fisher**, que es una traducción de la ecuación 4.21 y se encarga de calcular la distribución de los pesos, utilizando todos los datos conseguidos previamente.

El ratio de Fisher ha sido seleccionado para obtener datos cuantitativos sobre cual de las propiedades estudiadas anteriormente es la que mejor clasificación realiza. Por tanto se calcula la media total de los histogramas y la media de los histogramas con respecto a cada una de las clases que componen la base de datos. Una vez calculados los datos anteriores se aplica la ecuación 4.21, consiguiendo los siguientes resultados:

$$\left\{ \begin{array}{l} \text{Ratio para la propiedad grado: } 0.206 \\ \text{Ratio para la propiedad circularidad: } 0.679 \\ \text{Ratio para la propiedad joint degree: } 0.116 \end{array} \right. \quad (4.22)$$

De los resultados anteriores, se concluye que aplicando el *ratio de Fisher*, la propiedad de circularidad es la que se considera mejor de las tres. Debido a que se busca una propiedad que tenga como resultado un valor de numerador muy alto, es decir una diferencia con las demás clases muy alta, mientras que el denominador debe ser un valor bajo, o lo que es lo mismo una diferencia mínima entre imágenes de la misma clase.

# Capítulo 5

## Análisis de la información

Este capítulo comienza con una descripción de la base de datos utilizada para el proyecto, comentando los motivos de la existencia de la misma. Una vez descrita, se analizan los resultados de clasificación obtenidos tras utilizar las propiedades descritas en el capítulo anterior junto con las funciones de combinación expuestas en el presente capítulo.

### 5.1. La base de datos del proyecto

La base de datos de imágenes utilizadas proviene del proyecto de la Universidad de Brown conocido como *Shape Indexing of Image Databases* (SIID) [11]. El objetivo principal de este proyecto ha sido la creación de un sistema que sea capaz de recuperar imágenes a partir de su contorno. Mediante una aproximación que se fundamenta en la caracterización de la simetría en los mapas de borde, frente a otros métodos más clásicos como los basados en estadísticas globales invariantes, plantillas modificables o también aquellos que utilizan la correlación local de los bordes de los objetos en una imagen [13].

La base de datos es utilizada para comprobar la eficacia de la modificación al algoritmo de asignación basado en el grado [10], cuyo objetivo es el emparejamiento de grafos, utilizando para realizar esta unión un tipo especial de grafos denominados de choque (*shock graphs*). Estos *shock graphs* consisten principalmente en obtener el esqueleto de la figura de una imagen mediante

la operación morfológica de erosión, que se aplica forma consecutiva de tal forma que se evite dividir la figura.



Figura 5.1: Ejemplos de esqueletización que ha realizado el equipo de la Universidad de Brown

Entre los motivos por los que han usado este algoritmo para la base de datos se encuentran:

- Evita mínimos locales utilizando la convexidad basada en grados.
- Hace uso de la propiedad *sparse*, en la que se ahorra espacio en el uso de la memoria del ordenador.
- Gestiona la pérdida y duplicidad de nodos y aristas, aspecto importante en la clasificación de grafos.
- Es estable frente a condiciones desfavorables que generan ruido.

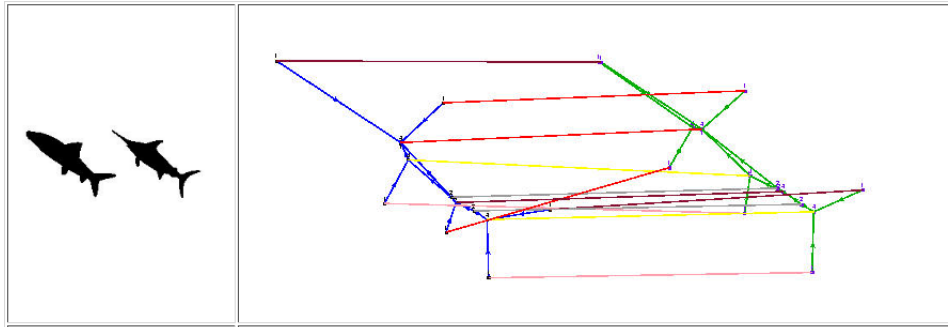


Figura 5.2: Ejemplo del resultado del algoritmo de emparejamiento de esqueletos creado por el equipo del que se ha compartido la misma base de datos de imágenes

El desarrollo de su algoritmo se puede resumir en los siguientes conceptos:

- Deformación de las figuras basada en la distancia entre contornos.
- Particionar el espacio o conjunto de los contornos.
- Definir la colección de caminos de deformación de los contornos.
- Evitar caminos que conlleven una gran complejidad.
- Crear el algoritmo de distancias entre transformaciones o deformaciones.
- Asignar el coste a cada una de las transformaciones.

A continuación, se muestran las clases tal como han sido distribuidas a partir de las imágenes de la base de datos obtenida de la Universidad de Brown[11], para evitar tener que indicar a lo largo de todo el análisis de los resultados a que imágenes pertenece una clase, aunque para los primeros casos se indicarán a cuales corresponden.

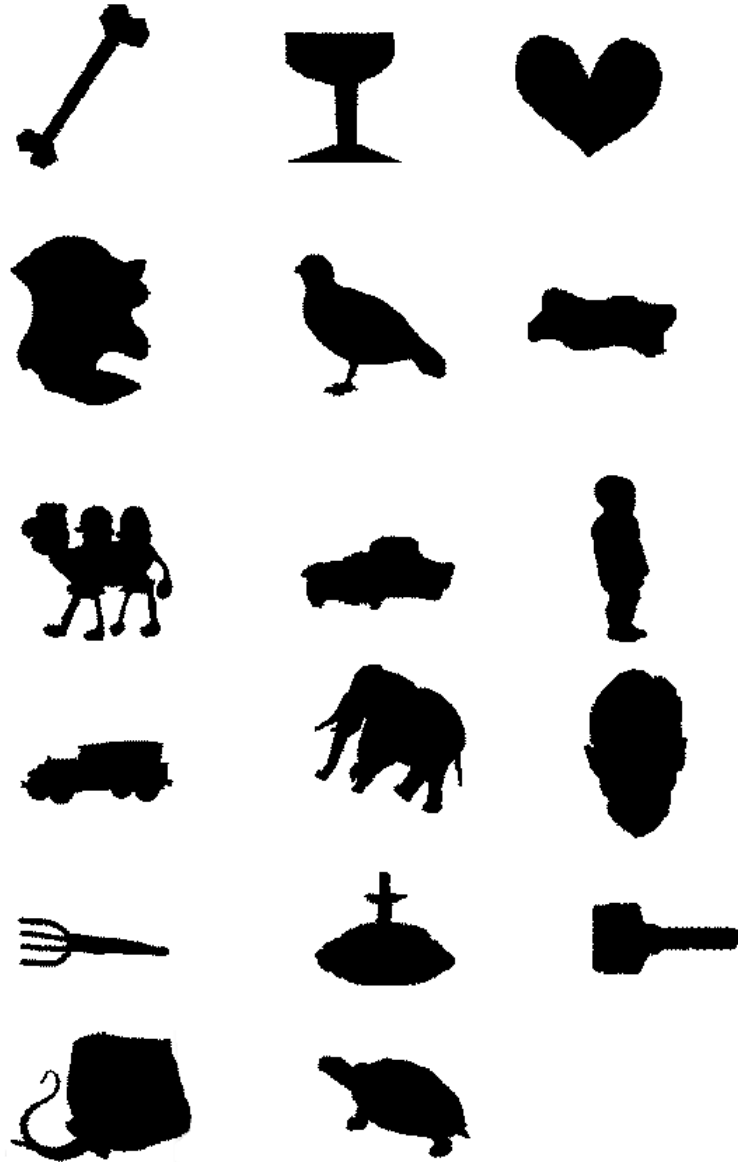


Figura 5.3: Muestra de una figura por cada una de las 17 clases de las que se compone la BBDD

Para resumir la lista de clases a continuación se muestra la lista de todas

las clases de las cuales se ha mostrado una figura por cada una de ellas. En la imagen 5.3 el orden que siguen respecto a la siguiente lista es por filas y en sentido de izquierda a derecha.

- Clase 1: "Bone"
- Clase 2: "Glass"
- Clase 3: "Heart"
- Clase 4: "Misk"
- Clase 5: "Bird"
- Clase 6: "Brick "
- Clase 7: "Camel"
- Clase 8: "Car"
- Clase 9: "Children"
- Clase 10: "Classic"
- Clase 11: "Elephant"
- Clase 12: "Face"
- Clase 13: "Fork"
- Clase 14: "Fountain"
- Clase 15: "Key"
- Clase 16: "Ray"
- Clase 17: "Turtle"

## 5.2. Análisis inicial

Llegados a este punto en el que cada imagen cuenta con 3 histogramas con información sobre las características de grado, circularidad y joint degree. Para el posterior análisis de estos datos se ha creado un script (`creacionEstructuraFinal.m`), en el que básicamente se reúne la información conseguida hasta ahora. Para ello se ha creado un vector de registros, donde cada entrada contiene cuatro campos: el nombre de la imagen y los tres vectores correspondientes a cada histograma.

Con la información ya estructurada, es posible avanzar al siguiente paso, que consiste en reducir la información de cada característica a un sólo número que resume el grado de similitud entre las imágenes. Es decir, partiendo de los histogramas de una característica de dos imágenes, el objetivo al que se quiere llegar es obtener un único dato que indique la similitud o diferencia que hay entre ambas, para poder establecer si pertenecen a la misma clase o categoría.

Para obtener ese dato, en un primer momento se pensó en usar el sumatorio de las diferencias de todos los elementos de ambos valores. Pero se llegó a la conclusión de que el resultado obtenido no estaba normalizado, porque al realizar el sumatorio de los valores de la comparación de todas las imágenes con todas, el resultado no era el valor 1, es decir la totalidad de la tasa.

La ecuación definitiva con la que se han obtenido los resultados normalizados en la distancia de una característica entre dos imágenes es la siguiente:

$$dist(i, j) = \frac{\sum_{k=0}^N |hist_i(k) - hist_j(k)|}{\sum_{k=0}^N \max\{hist_i(k), hist_j(k)\}} \quad (5.1)$$

donde  $N$  es el número de elementos de los histogramas,  $hist_i(k)$  e  $hist_j(k)$  son los elementos  $k$ -ésimos del histograma de las imágenes  $i$  y  $j$  respectivamente.

Este proceso debe realizarse para cada una de las 204 imágenes con el resto de imágenes, por lo que al finalizar el mismo para cada característica se dispondrá de una matriz de  $204 * 204$  con los valores de distancias. La diagonal de estas matrices tendrán valor cero, porque son las posiciones en la que se está comparando una imagen consigo misma por lo que la diferencia es nula.

Un aspecto a tener en cuenta es que he considerado la distancia como cero en los casos donde el resultado de la distancia entre dos imágenes sea inferior a un valor del orden de  $10^{-6}$ . En primer lugar, por ser un valor tan pequeño que no aporta nada en futuras operaciones y en segundo lugar si se intenta operar con valores de orden inferior provoca que Matlab se salga fuera de rango produciéndose resultados inválidos.

Esta última consideración surgió por causa de la característica de *joint degree* ya que al ser un vector de gran dimensión, tiene una gran distribución de los valores que en muchos casos llegan a ser valores muy cercanos a cero.

El tiempo en ejecución que ha necesitado el algoritmo (implementado en el fichero `calcularMatricesHistogramas.m`) es de 35 segundos.

A continuación se muestra la representación gráfica de cada una de ellas, utilizando para ello la función `imshow` del toolbox de Visión Artificial de Matlab. En primer lugar hay que tener en cuenta que valores más cercanos a cero (color negro en las posiciones) indica una distancia menor por lo que son consideradas más parecidas, mientras que si el color de la posición de la matriz tiende a blanco, indica una diferencia importante entre las imágenes.

Otro aspecto a tener en cuenta es cómo están distribuidas las imágenes en la matriz. La distribución que sigue es ordenada, donde todas las imágenes de una clase están juntas y no mezcladas con otras clases. Por lo que la representación ideal de clasificación sería una matriz cuya diagonal estaría formada por bloques de dimensión  $12 * 12$  con un valor cercano a cero, mientras que el resto de la matriz debería tener valores más altos de distancias entre imágenes, es decir colores más claros.

### 5.2.1. Análisis de la característica de grado

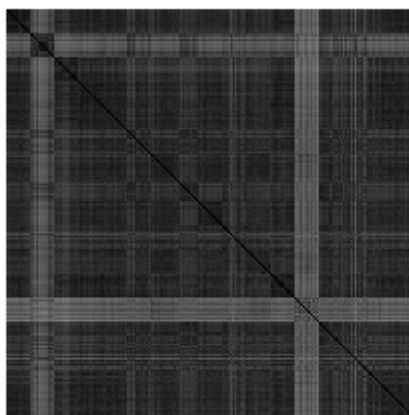


Figura 5.4: Representación de la matriz con las distancias de imágenes basada en el grado



Analizando la imagen 5.4 que se corresponde con la matriz de grados, en un primer momento se observa que sólo la clase 2 de la base de datos, denominada “*Glass*”, es la que cumple una mejor clasificación con respecto al resto de clases, ya que sólo para las imágenes suyas tienen un valor más cercano a cero como se puede observar en el segundo bloque de la diagonal, al que pertenecen todas las imágenes de esa clase.

Mientras que para la clase 13 denominada “*Fork*” la distancia para la característica es bastante elevada para todas las imágenes incluyendo a las de su propia clase. Y en el extremo contrario, se observan distancias similares con el resto de clases para las categorías 4 y 5, “*Misk*” y “*Bird*” respectivamente.

Entre las clases no comentadas todavía, hay algunas en las que sí se aprecia que hay una menor distancia en su clase, ya que la subventana aparece más oscurecida frente al resto de las clases. Pero con el resto de las clases no se aprecia una distancia muy grande, siendo este el caso de las clases 7 (camel), 8 (car), 9 (children), 11 (elephant) y en la que mayor diferencia se aprecia en este subgrupo, que es en la clase 12 (face) en la que su bloque de imágenes de la misma categoría tiene menos distancia con el resto.

Después de analizar esta característica de forma individual, no consigue separar correctamente las clases, porque de forma global la distancia entre cualquier par de imágenes es demasiada baja, exceptuando los casos de las imágenes de la clase 2 y 13 cuya distancia es mayor respecto al resto de clases e incluso en el caso de la clase 13 la distancia es mayor entre imágenes de su mismo grupo.

Como un análisis visual de la capacidad de clasificación no es suficiente para conocer de forma precisa los resultados que se generan, también se analiza su matriz de confusión representada en la tabla 5.1 situada en la sección 5.2.4 con el número de imágenes que es capaz de clasificar correctamente dentro de una clase, siendo una clasificación perfecta el caso en el que se clasifiquen las 12 imágenes de una clase como pertenecientes a la misma. El valor de umbral seleccionado ha sido de 0.30, porque es el elegido en los posteriores análisis con las distintas funciones de combinación y la decisión de tal elección se profundiza en el capítulo 5.

De la tabla 5.1 cabe destacar que aunque se creía que para la segunda

clase se obtenía los mejores resultados, en realidad no es así, porque la clase 13 aunque tiene un valor de distancia en general mayor es la que consigue clasificar 3 imágenes correctamente. Para el resto de clases en ningún caso supera en más de uno, la cantidad de imágenes correctamente clasificadas.

### 5.2.2. Análisis de la característica de la circularidad

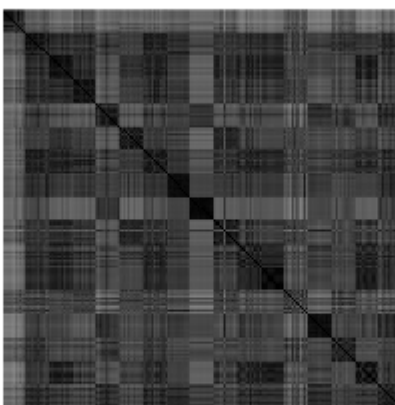


Figura 5.5: Representación de la matriz con las distancias de imágenes basada en la circularidad

En la imagen 5.5, obtenida de invocar `imshow` sobre la matriz de circularidad, se observa en un primer análisis que consigue clasificar mejor la mayoría de las categorías de las que se compone la base de datos.

Así las clases 8 (*car*) y 9 (*children*) en su subventana de la diagonal correspondiente a todas las imágenes de cada categoría respectivamente, se observa que tienen una distancia de cero o muy cercana para las imágenes de la misma clase. Las clases 12 (*face*) y 14 (*fountain*) son similares al caso anterior sólo que tienen para la misma clase una distancia ligeramente superior.

En el resto de las clases, la matriz muestra que su distancia se diferencia del resto ya que se distingue su subventana en la diagonal del resto de imágenes de su fila correspondiente. Excepto para los casos de la clase 2 (Glass), que precisamente con la característica de grado si se conseguía clasificar correctamente, y las clases 10 (classic) y 13 (fork), las cuales tienen distancias similares con otras clases.

Por tanto esta propiedad aunque consigue unos resultados de clasificación más aceptables, es decir, una distancia muy cercana a cero en el caso de las imágenes pertenecientes a una misma categoría y distancias superiores frente a imágenes de otros grupos, es necesario seguir experimentando nuevas formas de realizar la clasificación con la base de datos para obtener mejores resultados.

Como se hizo con la anterior característica, en la tabla 5.2 del apartado 5.2.4 se muestra la matriz de confusión para el caso de la propiedad de la circularidad de forma aislada para comprobar con una mayor precisión su capacidad de clasificación. Aunque los resultados que se obtienen no son suficientes, consigue clasificar 5 imágenes de la primera clase correctamente, mientras que para el resto de los casos no se consigue superar en 2 las imágenes clasificadas correctamente.

### 5.2.3. Analisis de la característica de la distribución joint degree

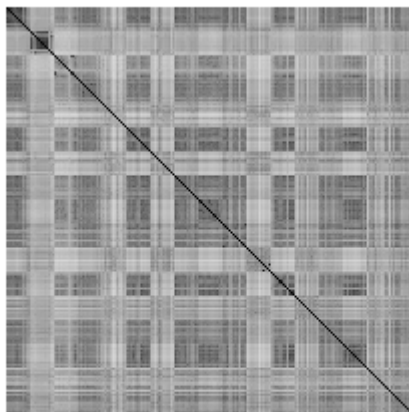


Figura 5.6: Representación de la matriz con las distancias de imágenes basada en joint degree

Para la última característica, la gráfica correspondiente a la matriz de distancias de joint degree, lo primero que se observa es que hay un nivel de distancia mayor frente al resto de características.

Esta matriz indica que las clases 5 (bird), 7 (camel), 11 (elephant), 13 (fork), 16 (ray) y 17 (turtle) tienen un valor de distancia muy alto entre todas las clases incluidas para las imágenes de la misma categoría.

Sólo en las clases 1 (bone), 2 (glass) y 12 (face) se aprecia que tienen una distancia inferior entre imágenes del mismo tipo respecto al resto de clases. El resto de categorías mantiene un nivel de distancia intermedio, aunque de todas formas sigue siendo más elevado frente a los valores vistos en las otras 2 propiedades comentadas anteriormente.

El resultado general de clasificación de esta característica parece ser inferior respecto a lo obtenido en el caso analizado de la circularidad. Esta disminución en la precisión de probabilidad se debe a que en muchas de las categorías las imágenes que pertenecen a ésta tienen una distancia muy elevada, por lo que se pueden producir múltiples casos de falsos negativos.

Por último como en los dos apartados anteriores se concluye el análisis de esta característica con la matriz de confusión para conocer de forma cuantitativa su capacidad de clasificación. Como esta característica tiene de forma general un valor de umbral más alto, para un primer análisis se ha seleccionado el valor de umbral de 0.45 al realizar el escalado.

En este caso los datos proporcionados por la tabla 5.3 del apartado 5.2.4 indican que esta medida de forma aislada es la que consigue mejores resultados, en contra del análisis realizado previamente. Este buen resultado se debe a que a valores mayores de umbral realiza una mayor eliminación de conexiones frente a las otras características por lo que se reduce antes el número de falsos positivos y consigue una mejor clasificación. Aún con estos resultados todavía hay 12 clases en las cuales se cometen errores de clasificación.

Con lo examinado hasta este punto se llega a la conclusión en que no es suficiente contar con varias propiedades y determinar cuál de todas es la que ofrece mejores resultados. En el siguiente apartado se analiza cómo pueden combinarse las características para obtener un clasificador más eficiente.

#### **5.2.4. Matrices de confusión de las características estudiadas**

Tal como se mencionó en los apartados anteriores, a continuación se muestran las tres tablas comentadas previamente en el análisis de cada característica.

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17
c1	1	0	1	1	1	0	1	1	1	0	1	1	0	1	0	1	1
c2	0	1	0	1	1	1	1	1	1	1	1	1	0	1	1	0	0
c3	1	0	1	1	1	1	1	1	1	0	1	1	0	0	0	1	1
c4	1	0	1	1	1	0	1	1	1	0	1	1	0	1	0	1	1
c5	1	0	1	1	1	1	1	1	1	0	1	1	0	0	0	1	1
c6	1	0	0	1	1	1	1	1	1	0	1	1	0	1	0	1	1
c7	1	0	1	1	1	1	1	1	1	0	1	1	0	0	0	1	1
c8	1	0	0	1	1	1	1	1	1	0	1	1	0	1	0	1	1
c9	1	0	1	1	1	1	1	1	1	0	1	1	0	0	0	1	1
c10	1	0	0	0	1	1	1	1	1	1	1	1	0	1	0	1	1
c11	1	0	0	1	1	0	1	1	1	1	1	1	0	1	0	1	1
c12	1	0	1	1	1	1	1	1	1	0	1	1	0	0	0	1	1
c13	0	0	0	0	1	0	2	0	1	1	2	0	3	0	1	1	0
c14	1	0	0	1	1	1	1	1	1	0	1	1	0	1	0	1	1
c15	1	0	0	1	1	1	1	1	1	1	1	1	0	1	0	0	1
c16	1	0	1	1	1	0	1	1	1	0	1	1	0	1	0	1	1
c17	1	0	1	1	1	0	1	1	1	0	1	1	0	1	0	1	1

Cuadro 5.1: Matriz de confusión para la característica grado de forma individual con un valor de umbral de 0.30

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17
c1	5	1	0	1	0	1	0	1	0	1	0	0	1	0	1	0	0
c2	0	1	1	1	0	1	1	1	0	1	1	1	0	0	1	1	1
c3	0	1	1	1	0	1	1	1	0	1	1	1	0	0	1	1	1
c4	0	1	1	1	0	1	1	1	0	1	1	1	1	0	0	1	1
c5	0	1	0	0	2	0	1	1	2	0	1	0	0	2	1	0	1
c6	0	1	2	1	0	2	1	1	0	1	1	0	1	0	0	1	0
c7	0	1	1	1	0	1	1	1	0	1	1	1	0	0	1	1	1
c8	0	1	1	1	1	0	1	1	0	1	1	1	0	1	0	1	1
c9	0	1	0	0	2	0	0	2	2	0	0	0	0	2	1	0	2
c10	0	1	1	1	0	1	1	1	0	1	1	1	1	1	0	1	0
c11	0	1	1	1	0	0	1	1	0	1	1	1	0	1	1	1	1
c12	0	1	1	1	0	0	1	1	0	1	1	1	1	1	0	1	1
c13	0	1	1	2	0	1	1	1	0	1	1	1	1	0	1	0	0
c14	0	1	1	0	1	0	1	1	0	0	1	1	0	2	1	1	1
c15	0	1	1	1	1	0	1	0	0	1	1	0	1	1	2	0	1
c16	0	1	1	1	0	1	1	1	0	1	1	1	0	1	0	1	1
c17	0	1	1	0	1	0	1	1	1	0	1	1	0	1	1	1	1

Cuadro 5.2: Matriz de confusión para la característica de la circularidad de forma individual con un valor de umbral de 0.30

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17
c1	3	0	0	1	0	2	0	1	1	1	0	1	0	0	1	0	1
c2	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c3	0	0	5	0	0	1	0	0	2	0	0	2	0	0	2	0	0
c4	2	0	0	3	0	2	0	1	0	1	0	1	0	0	1	0	1
c5	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0
c6	1	0	0	1	0	2	0	2	2	1	0	1	0	0	2	0	0
c7	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0
c8	2	0	0	1	0	2	0	2	1	1	0	1	0	1	1	0	0
c9	1	0	0	0	0	2	0	1	3	0	0	2	0	0	3	0	0
c10	2	0	0	1	0	1	0	2	0	3	0	0	0	1	1	0	1
c11	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0
c12	1	0	0	1	0	2	0	1	2	1	0	2	0	0	2	0	0
c13	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0
c14	2	0	0	1	1	0	2	0	2	0	0	0	0	3	0	0	1
c15	1	0	0	0	0	2	0	1	2	1	0	2	0	0	3	0	0
c16	2	0	0	1	0	1	0	1	0	1	0	0	0	0	0	5	1
c17	3	0	0	1	0	1	0	2	0	2	0	0	0	1	0	0	2

Cuadro 5.3: Matriz de confusión para la característica de joint degree de forma individual con un valor de umbral de 0.45



### 5.3. Funciones de combinación

Finalmente con las matrices de distancias obtenidas para cada una de las tres características, es necesario combinar todos los datos en una única matriz de la que poder extraer conclusiones acerca de la capacidad de clasificación de estas características.

Pero la cantidad de maneras posibles de poder combinar estas tres características en una sola, es muy amplia. Para el estudio se han recogido 10 posibles ecuaciones. En todas las ecuaciones la nomenclatura  $M_g(i, j)$  es la distancia de las imágenes  $i$  y  $j$  con respecto a la característica grado y  $M_c$  y  $M_{jd}$  son las distancias para las características de circularidad y *joint degree* respectivamente. Las tres principales se corresponden con :

- Media :

$$M_m(i, j) = \frac{(M_g(i, j) + M_c(i, j) + M_{jd}(i, j))}{3} \quad (5.2)$$

- Media cuadrática :

$$M_{mc}(i, j) = \sqrt{\frac{(M_g(i, j))^2 + M_c(i, j)^2 + M_{jd}(i, j)^2}{3}} \quad (5.3)$$

- Máximo :

$$M_{\text{máx}}(i, j) = \text{máx}\{M_g(i, j), M_c(i, j), M_{jd}(i, j)\} \quad (5.4)$$

Aparte de estas funciones como principales representantes de todas las posibles para realizar la combinación también se ha querido experimentar aplicando diferentes distribuciones de pesos a cada una de las tres características. El modo elegido para aplicar los pesos ha sido aplicar un peso mayoritario a una característica y el resto del peso disponible repartirlo de forma equitativa entre el resto de las características.

Los porcentajes aplicados han sido del 0.60 y el 0.40 repartido entre las otras características y en un segundo caso de distribución se da incluso mayor peso a la característica principal siendo de un 0.80 y el resto se divide a un 0.10 cada una de las restantes propiedades.

De tal modo que se han obtenido otras seis funciones ya que son las dos distribuciones explicadas anteriormente por cada una de las tres características que se han extraído a lo largo del proyecto.

La última de las ecuaciones de combinación se corresponde también a una distribución de pesos, en la que los porcentajes se distribuyen según el valor obtenido para cada uno de ellos con los resultados de la ecuación del *ratio de Fisher*.

Con las ecuaciones descritas anteriormente, se van a crear diez matrices una con el resultado correspondiente de la ecuación aplicada para todos los elementos o imágenes de las 3 matrices con las que ha comenzado este proceso.

Ahora al igual que con las tres matrices de características se va a comentar la representación visual de las matrices obtenidas de cada uno de estos experimentos de combinación.

### 5.3.1. Experimentación con la combinación basada en la media

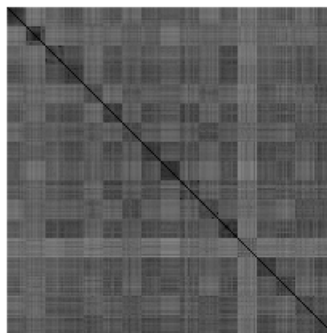


Figura 5.7: Representación de la matriz que sintetiza las 3 características mediante la combinación de la media

La matriz que aparece en la anterior figura se corresponde a la resultante de utilizar la combinación basada en la media. Se puede observar que en general el valor de la distancia es mayor, que en los casos aislados de las características de grado y circularidad.

Otro detalle que se puede apreciar es que para la mayoría de las clases el valor más bajo de distancia, se corresponde con la ventana de la diagonal de la matriz asociada a todas las imágenes de una clase.

Excepto para la clase 13 (fork), la cual sigue teniendo un valor medio de distancia mayor frente a todas las imágenes, incluidas las de su propia clase. También pueden darse problemas en la clasificación ya que en estos resultados, se consideran imágenes de la clase 3 (heart) y 12 (face) como pertenecientes también a la clase 4 (misk), debido a que tienen unos valores de distancias muy similares.

Como conclusión sobre esta combinación se tiene en cuenta que el resultado deriva del obtenido para el caso de la circularidad de forma aislada. Pero consigue una visible mejoría, porque se mantiene o se ha disminuido la distancia entre imágenes de la misma clase, mientras que al mismo tiempo se ha conseguido que la distancia entre categorías diferentes sea mayor comparado con la característica de circularidad únicamente.

### 5.3.2. Experimentación con la combinación basada en la media cuadrática

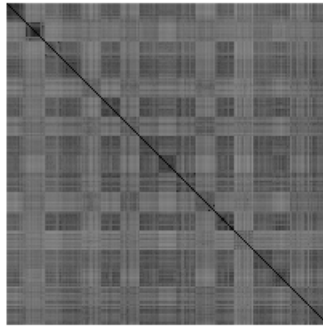


Figura 5.8: Representación de la matriz que sintetiza las 3 características mediante la combinación basada en la media cuadrática

Comparando la nueva matriz visualizada en la figura 5.8, resultado de aplicar la combinación basada en la media cuadrática, se puede llegar a la conclusión inicial de que realiza un proceso de clasificación peor, frente al resultado ofrecido por la media.

El valor de distancia en general para toda la matriz es mayor al de todos los casos vistos hasta ahora excepto para el de la característica individual *joint degree*. El principal problema de tener un valor de distancia más alto en general es que varias clases se van a clasificar como pertenecientes a la misma categoría. Este caso mencionado ocurre en aquellas filas correspondientes a un valor de gris más claro o un valor de distancia más alto. Por tanto las clases que van a incluir imágenes correspondientes a otras categorías van a ser principalmente la 5 (bird), 7 (camel), 11 (elephant) y 13 (fork).

El resto de las clases en su mayoría tiene un nivel mayor de distancia frente al resto de imágenes, aunque en algunos casos, ocurre que algunas de las imágenes de la clase queden excluidas y en cambio se clasifiquen como tales imágenes de otras clases, como ocurre en la clases 3 (heart), 4 (misk), 8

(car), 10 (classic) y 16 (ray).

De forma general en este experimento sólo se ha conseguido elevar ligeramente la distancia entre imágenes, la cual tiene como inconveniente el caso de un valor de umbralización alto, se van a considerar como imágenes pertenecientes al mismo grupo otras que no lo son, aumentando por tanto la cantidad de falsos positivos, mientras que si se realiza una un escalado más restrictivo con un valor de umbral bajo, es posible que para algunos grupos no se consiga clasificar incluso ninguna de las correspondientes a esta categoría, elevando de forma considerable el número de falsos negativos, especialmente en las clases que cuentan con un valor de distancia más elevado en general, siendo este el caso de las clases ya mencionadas anteriormente (5, 7, 11 y 13).

### 5.3.3. Experimentación con la combinación basada en el máximo

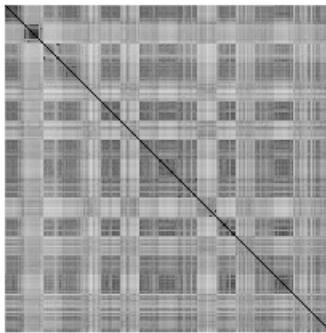


Figura 5.9: Representación de la matriz que sintetiza las 3 características mediante la combinación basada en el máximo

En las distancias obtenidas seleccionando como distancia para cada imagen el valor máximo de las 3 características, se obtiene como cabría esperar, la matriz con los valores más altos lo que se resume en una imagen con una

intensidad de gris más suave, como puede comprobarse en la figura 5.9.

Como en los casos ya comentados con la ecuación de la media cuadrática en la que se acentúa el valor de la distancia para determinadas clases lo que conlleva el incluir en varias categorías imágenes de otro tipo, en este resultado, vuelve a ocurrir con los tipos 5 (bird), 7 (camel), 11 (elephant) y 13 (fork).

También hay que comentar que para la clase 2 (glass) se llega a realizar una buena clasificación de sus imágenes, ya que entre ellas hay una distancia de un valor muy próximo al cero, mientras que para el resto de clases la distancia es mayor. En menor medida también realiza una buena clasificación de la primera clase (bone) donde la distancia de las imágenes “bone” es inferior al resto.

También es necesario comentar que en el caso de la clase 12, hay algunas imágenes que se clasifican perfectamente, pero el resto tiene un valor de distancia similar a otras clases por lo que se mezclaría en esta clase imágenes de varias categorías.

Por último, evaluando los valores de distancia de esta matriz, se comprueba que tiene mayor similitud con la característica de la distribución *joint degree*. Este punto es lógico ya que para cada posición de la matriz se está seleccionando aquel valor cuya distancia sea mayor y de forma general la matriz que tiene un valor medio de distancia mayor es la referente a la propiedad *joint degree*, por tanto este caso comparte todo lo estudiado en el caso de la propiedad de *joint degree* aislada.

### 5.3.4. Experimentación con la combinación basada en la distribución de pesos primera

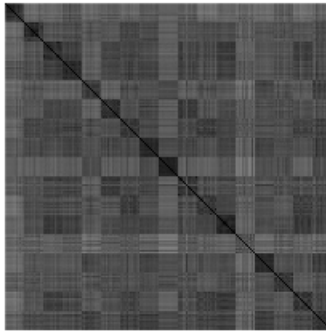


Figura 5.10: representación de la matriz que sintetiza las 3 características mediante la combinación basada en la primera distribución de pesos

La matriz de la figura 5.10 se ha obtenido dando una mayor importancia a la característica de circularidad, con un valor de 0.60 del peso total, mientras que a las características de grado y *joint degree* se les ha aplicado un valor del 0.20 del total a cada una de ellas.

Si se compara frente a la matriz de la característica de circularidad se observa que la distribución es más homogénea, donde ahora los valores de las distancias predomina un valor más cercano a cero que a uno, incluso en las distancias de los tipos 5 (bird), 7 (camel), 11 (elephant) y 13 (fork) frente al resto de imágenes.

Aunque se haya reducido en general la distancia, se consigue un clasificación aceptable en el resto de clases, principalmente en la clase 9 (children). Como se puede observar en la imagen obtenida, la distancia entre imágenes de la misma clase, es decir todos los bloques de la diagonal de la matriz tienen un valor de distancia inferior frente al resto de sus filas correspondientes, obteniendo una cantidad menor de errores en la clasificación.

Con los resultados mostrados por esta matriz se concluye que el mejor aproximamiento para realizar una buena clasificación es combinando todas las propiedades en un sólo resultado pero distribuyendo el peso de una manera lo más equitativa posible si no se conoce a priori cuál es la que ofrece los mejores resultados, caso que ya se ha realizado con la ecuación de las medias. Pero el objetivo de estas ecuaciones de distribuciones de pesos es determinar cual de las tres características es la más influyente en el proceso de clasificación.

De momento para esta primera distribución se observa que para la mayoría de las clases puede realizar una buena clasificación si se utiliza un valor de umbral más bien bajo para conseguir excluir las figuras de otros tipos. Principalmente en los casos de las figuras de las clases 5, 11 y 13 se tiene como efecto negativo el hecho de que se van a excluir imágenes de sus propias clases, lo que provoca un aumento de los falsos negativos generados por este método.

### 5.3.5. Experimentación con la combinación basada en la distribución de pesos segunda

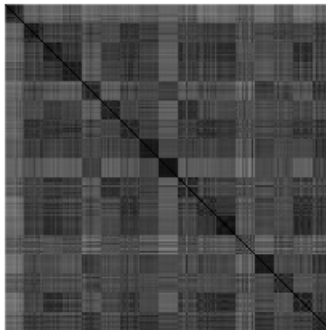


Figura 5.11: Representación de la matriz que sintetiza las 3 características mediante la combinación basada en la segunda distribución de pesos



En el caso de la figura 5.11 se ha dado incluso mayor importancia a la característica de la circularidad, en concreto se le ha aplicado a esta característica el 0.80 del peso total, mientras que a las restantes características se les ha aplicado el 0.10 de su peso.

Por tanto, esta matriz es muy similar a la obtenida para la característica de circularidad aislada, aunque con unos valores de distancia intra-clase algo superiores a los de esta última. Pero estos valores de distancia no son lo suficientemente mayores para clasificar como clases distintas. Por ejemplo, a la clase 3 (heart) frente a las clases 4 (misk), 12 (face) y 16 (ray) de las cuales clasificará como de su mismo tipo.

Y aunque en general se conserva para la mayoría de las imágenes de una clase una distancia mayor frente al resto, tendría que aplicarse una estrategia de eliminación de conexiones muy restrictiva para no incluir dentro de una categoría imágenes de otras clases. Por tanto la teoría de no aplicar casi todo el peso total a una propiedad se demuestra en este caso. Porque los resultados obtenidos por este clasificador son inferiores comparados con la ecuación donde se distribuía un 0.60 del peso total a la característica de la circularidad, donde el número de falsos negativos era más aceptable cuando se aplicaba al proceso de escalado un valor de umbral bajo. Pero en el caso actualmente estudiado, como se ha disminuido aún más la distancia en todos los casos, para conseguir reducir el número de falsos positivos es necesario reducir también el valor del umbral lo que conlleva que se van a excluir más figuras del tipo clasificado.

### 5.3.6. Experimentación con la combinación basada en la distribución de pesos tercera

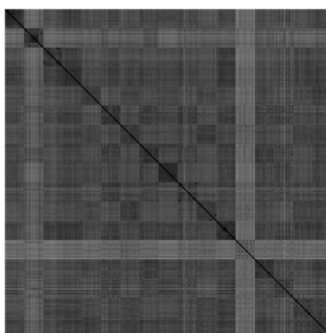


Figura 5.12: Representación de la matriz que sintetiza las 3 características mediante la combinación basada en la tercera distribución de pesos

En el caso de la figura 5.12 se da prioridad a la característica de grado, la cual por sí sola proporcionaba peores resultados de clasificación. La distribución que sigue es la de un 0.60 del peso total se le otorga a la característica de grado y el 0.40 restante se distribuye de manera equitativa.

Realizando un breve análisis se puede observar que en general ha aumentado el valor de la distancia para todos los casos, lo que ha conseguido como resultado que las distancias mínimas se encuentren entre las imágenes de cada clase.

Pero aunque en los bloques de la diagonal, en su mayoría tengan un valor inferior frente al resto, hay clases como la 5 (bird) o la 7 (camel) en la que se incluirán como imágenes de esta clase las de otros tipos por tener una distancia similar o incluso inferior, como en el caso de la clase 5 donde hay imágenes de la clase 17 (turtle) para los cuales tiene una distancia inferior frente a los de la suya propia.

Se vuelve a demostrar que la característica de grado no es la mejor de las 3 que se están estudiando. También se comprueba que gracias a la ecuación de combinación se consigue mejorar los resultados ofrecidos por esta propiedad debido a los valores aportados por las otras propiedades, esta mejora se observa en el aumento de la distancia entre figuras de distintas clases de manera global. Por tanto se sigue aceptando la conclusión de que el uso combinado de todas las propiedades con una distribución similar es beneficioso frente al uso individual de alguna de ellas.

### 5.3.7. Experimentación con la combinación basada en la distribución de pesos cuarta

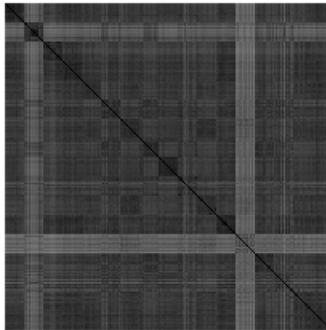


Figura 5.13: Representación de la matriz que sintetiza las 3 características mediante la combinación basada en la cuarta distribución de pesos

Para la combinación basada en la cuarta distribución de pesos se aplica mayor importancia a la característica de grado, en concreto de un 0.80, mientras que a las características de circularidad y grado se les aplica un 0.10 a cada una de ellas, como se puede comprobar en la figura 5.13.

En este caso como era de suponer, el resultado obtenido es muy similar al de la matriz con la característica de grado únicamente. Como se ha concluido en los experimentos evaluados anteriormente, se observa que aplicar

casi todo el peso de la distribución a una sólo característica, normalmente conlleva una disminución bastante visible de los aciertos del clasificador. Si a la afirmación anterior se le une el hecho de que la propiedad sobre la que recae el mayor peso es la que peores resultados ofrece de forma aislada se llega a la situación donde los resultados obtenidos no son los mejores para realizar la clasificación.

Aunque se disminuye ligeramente la distancia general entre todas las imágenes, y en los casos de las clases 1 (bone), 2 (glass), 9 (children) y 12 (face) se obtiene para esas imágenes una clasificación aceptable si se aplican medidas muy restrictivas, dependiendo del valor de distancia hasta el que se pueden considerar imágenes de una misma clase. Pero adoptando esa medida para algunas clases no se conseguiría asociar ninguna imagen. Por lo que en definitiva la característica de grado no es la mejor de las 3 para realizar la clasificación.

### 5.3.8. Experimentación con la combinación basada en la distribución de pesos quinta

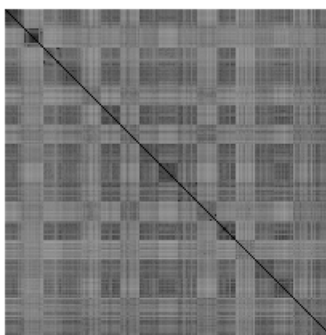


Figura 5.14: Representación de la matriz que sintetiza las 3 características mediante la combinación basada en la quinta distribución de pesos

En la quinta ecuación de distribución de pesos, se otorga un 0.60 del peso total a la característica *joint degree* y a las propiedades de grado y circularidad un 0.20 a cada una de ellas, como puede observarse en la figura 5.14.

Si se compara frente a la matriz de *joint degree* se observa que se ha reducido la distancia de forma general. Pero de todas formas esta reducción no es suficiente para conseguir clasificar correctamente las clases, porque aparte de las clases 1 (bone) y 2 (glass) que tienen una distancia inferior al resto, el resto de clases no tienen una distancia significativamente inferior frente a determinados tipos de figuras. Por tanto para conseguir mejores resultados es necesario aplicar un umbral de escalado no tan restrictivo como en experimentos previos, debido principalmente al valor de distancia más elevado. Este umbral más elevado va a permitir resultados con un bajo número tanto de falsos positivos y negativos, pero todo depende del valor del umbral seleccionado por lo que si se utiliza un valor muy bajo se corre el riesgo de excluir dentro de cada grupo muchas de sus figuras.

### 5.3.9. Experimentación con la combinación basada en la distribución de pesos sexta

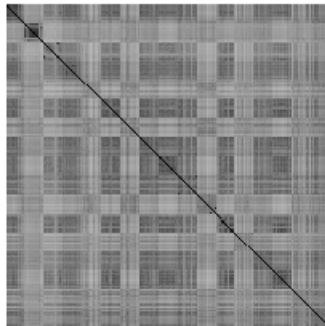


Figura 5.15: Representación de la matriz que sintetiza las 3 características mediante la combinación basada en la sexta distribución de pesos

La figura 5.15 muestra los resultados de la última ecuación de distribución de pesos, que se han distribuido de la siguiente manera: 0.80 para la propiedad *joint degree* y un 0.10 para la circularidad y grado.

Como era de esperar la distancia de toda la matriz en general ha aumentado, llegando a tener un valor de distancia ligeramente inferior que en el caso individual de *joint degree*. Al ser muy similar a esta última característica, reúne los mismos inconvenientes que ya se comentaron para ella, por lo que no es la más adecuada para obtener una clasificación lo más precisa posible.

Como se ha dicho a lo largo del análisis de todos estos experimentos en los que se ha jugado con la distribución de pesos, lo más favorable es realizar una distribución lo más equitativa posible si no se conoce cómo se van a comportar cada una de las características y en caso de conocer la propiedad más acertada intentar aplicar mayor peso a ésta, tal y como ocurre en el experimento de la siguiente sección.

### 5.3.10. Experimentación con la combinación basada en la distribución de pesos según el ratio de Fisher

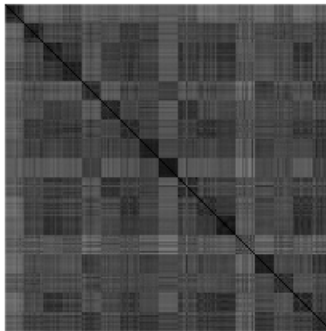


Figura 5.16: Representación de la matriz que sintetiza las 3 características mediante la combinación basada en la distribución de pesos utilizando el *ratio de Fisher* calculado

Según los valores normalizados del *ratio de Fisher* calculado a partir de las tres propiedades estudiadas en todo el proyecto, la circularidad es la característica con más influencia en el proceso de clasificación. Por tanto como la circularidad tiene más influencia se le ha aplicado un 0.67 del total, entonces según estos resultados, la circularidad es la que cuenta con una mayor distancia entre tipos de figuras distintas mientras que la distancia entre imágenes de la misma clase es pequeña. Por otra parte, la característica que supone con una peor capacidad de clasificación es la característica de *joint degree* con sólo un 0.11 del peso total.

Los datos conseguidos por esta distribución de pesos son muy similares a los vistos en el caso de la primera combinación basada en la distribución de pesos, por tanto comparte con ésta las mismas cualidades para realizar la clasificación de las imágenes de la base de datos utilizada.

### 5.3.11. Conclusiones de los análisis iniciales

Una vez finalizado el análisis de todas las matrices obtenidas tras aplicar las diferentes ecuaciones se llega a las siguientes conclusiones.

En primer lugar, en el caso de todas las combinaciones relacionadas con la distribución de pesos se han obtenido unos resultados que tienden a reducir la dispersión de los valores de las distancias de cada una de las características, dando un valor más homogéneo frente a cada una de ellas consideradas de forma individual. Es decir, se consiguen mejores resultados cuando se combinan todas las características y la forma en que se distribuyen, dando importancia a que el reparto de pesos sea lo más equilibrado posible, teniendo en cuenta la capacidad de clasificación de cada una de ellas. Por ejemplo, en la situación de una distancia mayor general, como es en el caso de *joint degree* las combinaciones de pesos en las que supone una mayor importancia a esta propiedad, la distancia tiende a disminuir. Mientras que en el caso del grado las distancias tienden a unos valores más altos, en las ecuaciones en las que se le aplican el mayor porcentaje del peso total.

En segundo lugar de todas las combinaciones que sintetizan las 3 características estudiadas las que mejores resultados aportan son la media, la media cuadrática y la ecuación que distribuye con un 0.60 del peso total a la

característica de circularidad.

Ya que en el resto de casos aunque se aplique un valor de umbral muy bajo, para controlar las conexiones entre imágenes, hay muchos casos en los que se va a clasificar como imágenes de una clase aquellas que pertenecen a otros tipos u otra posibilidad es la de excluir imágenes de un tipo concreto por tener un valor de distancia superior frente al resto de imágenes de su clase.

Para conocer realmente cuál de todas las combinaciones anteriormente expuestas es la que se comporta mejor como clasificador es necesario analizarlas de una manera más precisa, para ello se va a estudiar de forma cuantitativa los resultados obtenidos por cada una de ellas en el siguiente apartado.

## 5.4. Estudiando la topología

Uno de los principales objetos de estudio en las Redes Complejas es analizar como varían dinámicamente, al ir modificando de forma gradual las conexiones de los nodos adyacentes de la red. Dentro del proyecto realizado este proceso se situaría en el paso indicado como “analizar los resultados obtenidos”, tal y como aparece en la figura resumen 1.2 del primer capítulo. A continuación se puede ver una representación visual de la idea principal de este apartado.



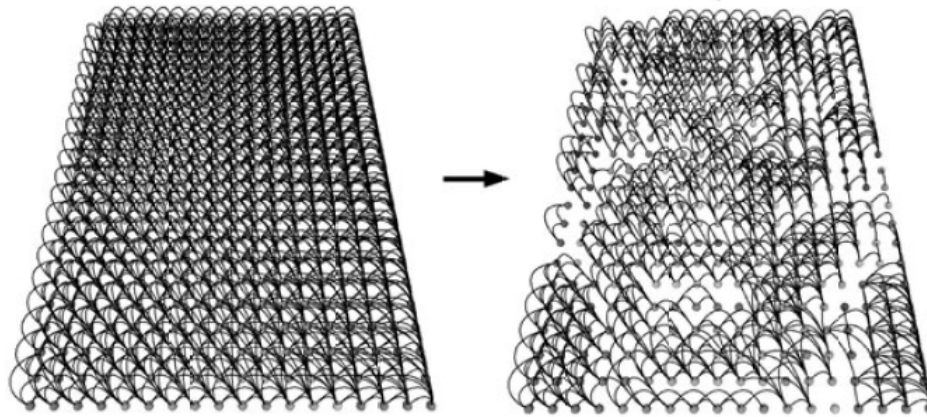


Figura 5.17: Resultado de eliminar conexiones superiores a un determinado umbral.

Por lo que se va a estudiar la topología de la red para cada una de las matrices resultado de aplicar las 10 combinaciones descritas en el apartado anterior.

El intervalo de los valores utilizado para estudiar la topología comprende el rango entre 1 y 0 disminuyendo el valor en 0.05 en cada paso. Analizando en cada uno de estos valores como se van perdiendo conexiones entre nodos. En el script `obtenerErroresMetrica.m`, se observa como se va a ir cambiando la estructura. A continuación, se muestra un pseudocódigo del algoritmo.

**Data:** La matriz con los valores de distancia entre cada para de imágenes y la ruta donde se va a almacenar el resultado

**Result:** Los vectores con la información de errores generales, falsos positivos, falsos negativos y la matriz de confusión

Obtener las dimensiones de la matriz;

```

for umbral = 1 downto umbral = 0 con pasos en 0,05 do
  for i = 1 hasta i = numFilasMatriz do
    for j = 1 hasta j = numColumnasMatriz do
      if matrizEntrada(i, j) > umbral then
        | matrizEntrada(i, j) = 0;
      end
      else
        | matrizEntrada(i, j) = 1;
      end
    end
  end
  Obtener los vectores de errores a partir de matrizEntrada;
  Obtener la matriz de confusión a partir de matrizEntrada;
end

```

**Algorithm 8:** Pseudocódigo para estudiar la topología de la red

Básicamente consiste en un bucle en el que se toma un valor umbral, el cual tiene como comienzo el valor 1. Ese valor establece el límite en el peso de las conexiones entre nodos, de tal forma que todas las aristas que lo superen se eliminan, es decir se ponen a cero dichas conexiones y todas aquellas conexiones que tengan un valor inferior o igual al del umbral, se establece el peso de su conexión a 1. Por lo tanto, para cada conexión que se mantiene, se supone que es entre imágenes de la misma clase.

Una vez que se ha establecido para la iteración actual la topología de la red, hay que obtener la cuantificación de los errores de clasificación para ese valor de umbral. Para encontrar todos aquellos casos de clasificación erróneos se utiliza una matriz de referencia en la que se conoce cuáles son los elementos que tienen valor 1, es decir son imágenes de la misma clase. Esta matriz ideal tiene a 1 los bloques de  $12 * 12$  de la diagonal, mientras que el resto de los elementos de la matriz tienen un valor de cero, esta colocación de los valores en la matriz referencia es debido a la estructura que sigue la matriz con los valores de distancia entre imágenes donde las filas y las columnas están

ordenadas por los conjuntos de figuras de cada clase, tal y como ya se explicó.

### 5.4.1. Errores generales en la clasificación

Para calcular los errores de clasificación, hay que obtener la matriz resultado de la diferencia entre la matriz a la que se le ha aplicado el umbral y la matriz referencia. Todos los casos en los que se haya cometido un fallo, se obtiene un valor de uno en esa matriz resultado. Finalmente para conocer el número de errores basta con sumar todos los elementos de la matriz.

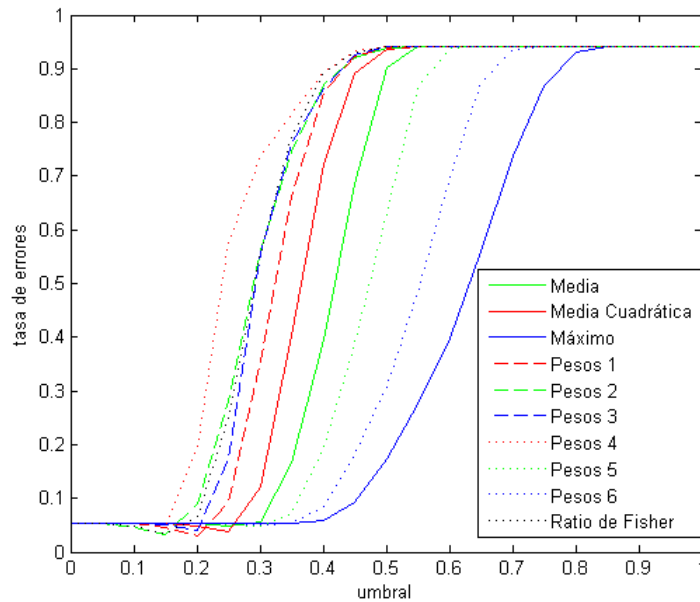


Figura 5.18: Representación de la capacidad de clasificación para las 10 combinaciones estudiadas.

La siguiente tabla muestra las distribuciones de pesos de las diferentes ecuaciones etiquetadas como peso que aparecen en la leyenda de la gráfica de la figura 5.18.

	Circularidad	Joint Degree	Grado
Pesos 1	0.60	0.20	0.20
Pesos 2	0.80	0.10	0.10
Pesos 3	0.20	0.20	0.60
Pesos 4	0.10	0.10	0.80
Pesos 5	0.20	0.60	0.20
Pesos 6	0.10	0.80	0.10
Ratio de Fisher	0.206	0.679	0.116

Cuadro 5.4: Tabla en la que se indica la distribución de los pesos de las funciones de combinación indicadas en la leyenda de la figura 5.18

En la gráfica mostrada en la figura 5.18 se observa que las curvas de las 10 combinaciones son crecientes, pero porque se ha representado de forma visual a la inversa del proceso que sigue el algoritmo. Como ya se ha dicho, el algoritmo varía la estructura de la red eliminando todas aquellas conexiones superiores a un valor dado que comienza en 1 y decrece hasta 0, en pasos de 0.05 unidades.

De esta forma se va refinando gradualmente la clasificación, partiendo de una matriz de clasificación donde se comete un 0.90 de fallos hasta alcanzar el caso de una matriz en la que el error de clasificación es inferior al 0.10. La decisión de utilizar la lógica inversa que se ha utilizado a la hora de realizar el estudio de la evolución de la topología de la red, se debe a que se ha intentado seguir de la manera más fielmente posible al método de estudio de la dinámica de la red explicado en el artículo dedicado al estudio del reconocimiento facial [4].

Por tanto la forma de evaluar la figura anterior es desde la derecha hacia la izquierda, conociendo que el eje de las abcisas es el valor del umbral permitido en las conexiones y el eje de ordenadas indica el porcentaje de fallos de clasificación que se han cometido.

Se observa que para todas las ecuaciones el alto error de clasificación, no empieza a descender hasta que se eliminan conexiones con un valor superior al del rango comprendido entre 0.30 y 0.60 dependiendo de la matriz correspondiente a la ecuación evaluada. Este caso se da debido a que para todos

los datos generados, no hay valores de distancia muy altos o cercanos al valor 1, excepto para aquellos casos en los que la combinación aporte una mayor importancia a la característica *joint degree*.

Las funciones de combinación de los máximos es la que se comporta mejor para el rango de valores entre 0.70 y 0.40 ya que el nivel de error disminuye más rápidamente. El resto de ecuaciones necesitan un valor umbral más bajo para que el porcentaje de errores disminuya, y además se puede observar que el cambio en el número de errores se produce de forma más brusca que en el caso de los máximos.

También se puede apreciar en la gráfica que la combinación que peor se comporta hasta el valor de umbral 0.15, es la que daba mayor distribución de pesos (un 0.80) a la característica de grado, la cual ya se había concluido en apartados anteriores, que no era la mejor de las características para la tarea de clasificación, unido al caso de que en general tampoco es beneficioso dar un valor de peso muy elevado a una sólo característica.

En la gráfica destaca el hecho de que la tasa de errores de la función correspondiente a la distribución de pesos según el *ratio de Fisher* esté más cerca a los casos de las ecuaciones de distribución de pesos donde se aplica un 0.80 a la circularidad y a la ecuación que aplica un 0.60 a la propiedad de grado, en vez de tener una tasa de errores similar a la que obtiene la ecuación de distribución de pesos primera. Por lo que un pequeño aumento en el peso de la propiedad sirve para disminuir los aciertos del clasificador.

Un último aspecto a destacar de la gráfica es el comportamiento de las ecuaciones de media y las cuatro primeras ecuaciones de distribución de pesos. Donde la tasa de errores disminuye entre el rango de umbral de 0.25 hasta 0.05, dependiendo de la ecuación, pero a continuación vuelve a aumentar el número de errores después de reducir más el valor de umbral de las conexiones.

Este suceso es debido a que como se utiliza un valor tan bajo de distancia para diferenciar las clases, el clasificador identifica como clases distintas, imágenes de una misma categoría formando una especie de subgrupos dentro de una misma clase. Similar a lo comentado en la introducción del proyecto donde dentro de una categoría como por ejemplo las “aves” hay algunos

casos muy diferentes al resto, es decir entre por ejemplo una “gaviota” y un “avestruz”. Lo que se ha explicado ahora es lo mismo que se ha dicho en varias ocasiones y se resume en el hecho de que hay algunas imágenes de una clase con un valor de distancia superior al resto lo que provoca que dichas imágenes se incluyan en el grupo recién clasificado.

Para el resto de ecuaciones el porcentaje de errores se estabiliza a partir del valor 0.30, en el cual se llega a una tasa del 0.05 en errores de clasificación para las combinaciones basadas en el máximo, pesos 5, pesos 6 y una tasa de 0.06 para la función de la media cuadrática.

#### 5.4.2. Falsos positivos en la clasificación

Aparte de conocer el número de errores en la clasificación, también interesa conocer el número de errores donde se han clasificado como iguales, imágenes de clases diferentes, es decir falsos positivos.

Para calcular los falsos positivos, basta con realizar la diferencia de la matriz referencia, explicada en el apartado de errores generales, con la matriz resultado de la clasificación. Los falsos positivos serán todos aquellos elementos de la matriz resultado con valor -1. Y para obtener la cantidad de coincidencias basta con utilizar las funciones de Matlab `find`, seguida de `length` sobre el vector resultado dado por la función anterior, tal y como se ha indicado varias veces a lo largo de la memoria.

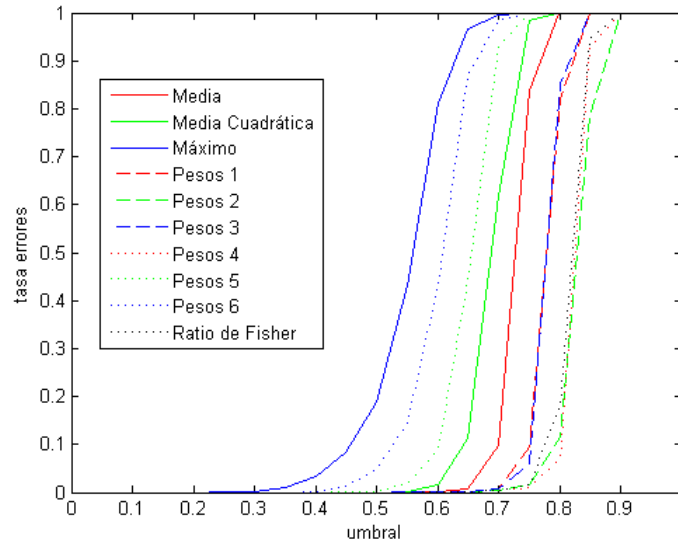


Figura 5.19: Representación gráfica de los falsos positivos para las 10 combinaciones. La distribución de los pesos de las diferentes funciones, aparece en la tabla 5.4

Tal como se observa en la gráfica 5.19, el comportamiento de todas las funciones de combinación en el rango de valores de umbral desde 0.5 hasta 1, es igual al visto para el caso en el que no se distingue el tipo de error en la clasificación.

Mientras que en el rango de valores entre 0 y 0.5, el error de falsos positivos es muy bajo. Debido a que se han eliminado la mayoría de conexiones entre imágenes, incluso entre aquellas que pertenecen a la misma categoría por lo que aunque el clasificador no consiga agrupar todas las imágenes de una categoría no van a producirse falsos positivos, debido a que las conexiones restantes a un umbral bajo son entre imágenes de la misma categoría.

La curva es ascendente porque a mayor valor de umbral, se están permitiendo conexiones con un valor de distancia más alto suavizando las restricciones de la clasificación, por tanto se están cometiendo más errores ya que se clasifican como imágenes de la misma clase aquellas que no lo son, el último tipo específico de errores se va a desarrollar en el siguiente apartado.

### 5.4.3. Falsos negativos en la clasificación

Por último, también interesa conocer la cantidad de errores en los que no se estaba clasificando una imagen a una clase, cuando en realidad sí pertenece a ella, este caso se denomina falsos negativos. Los pasos para obtener el número de falsos negativos son los mismos que en el caso de los falsos positivos, sólo que ahora hay que buscar aquellos elementos en la matriz resultado con valor 1.

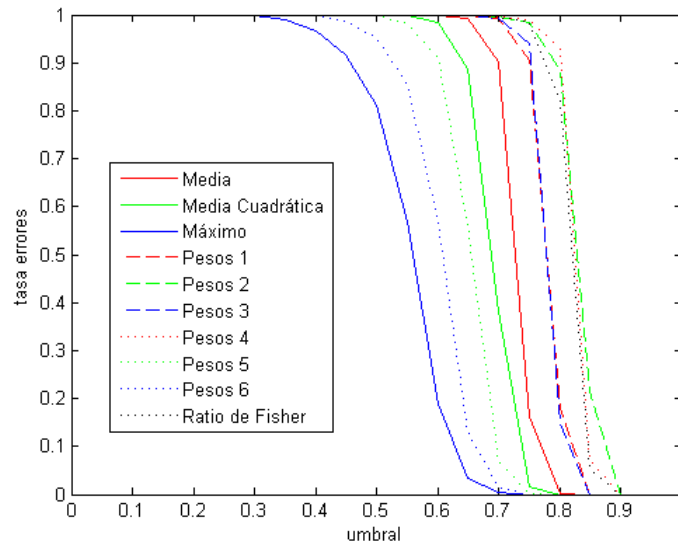


Figura 5.20: Representación gráfica de los falsos negativos para las 10 combinaciones. En el eje de las abcisas aparece el umbral utilizado, mientras que en el eje de las ordenadas se corresponde a la tasa de falsos negativos.

La distribución de los pesos de las diferentes combinaciones que aparecen etiquetadas en la leyenda de la gráfica 5.20, aparece en la tabla 5.4.

La gráfica 5.20 muestra de forma rápida, como es la idea opuesta a los falsos positivos. Las curvas de las ecuaciones son decrecientes, debido a que a



un bajo valor de umbral se están eliminando la mayoría de conexiones entre imágenes, siendo muchas de ellas correspondientes a imágenes de la misma clase. Aunque el porcentaje de error inicial no es tan alto como cuando se están identificando falsos positivos.

Como los valores de distancia, en su mayoría no superan la cantidad de 0.5, a partir de ese umbral se conservan casi todas las conexiones, por lo que no hay falsos negativos, mientras que de forma simultánea va creciendo el número de falsos positivos.

También se puede observar en la gráfica que la función que mejor se comporta es la combinación basada en la distribución que daba un 0.80 del peso total a la característica de circularidad, mientras que la que obtiene una mayor cantidad de falsos negativos, es la combinación correspondiente al máximo, la cual de forma general ofrece un mejor porcentaje de clasificación, excepto para este caso concreto, debido a que consta con un valor de distancia superior frente al resto de ecuaciones en las que no se otorga el mayor peso a la característica *joint degree*.

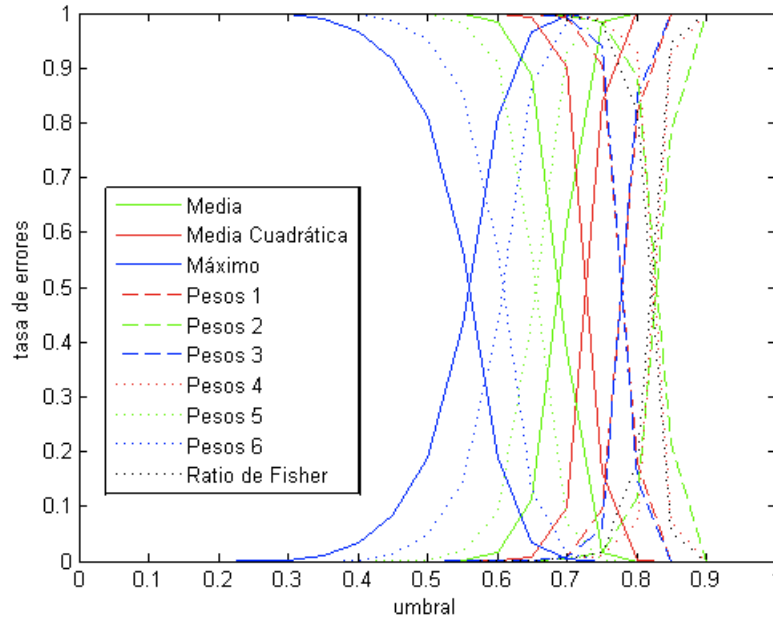


Figura 5.21: Representación gráfica de los falsos negativos junto con los falsos positivos para cada una de las 10 combinaciones. En el eje de las abcisas aparece el umbral utilizado, mientras que en el eje de las ordenadas se corresponde a la tasa de errores.

La gráfica 5.22 muestra para cada posible combinación el valor de umbral para el cual se intersecan las curvas de los falsos positivos con los falsos negativos. El punto de intersección en todos los casos es en el 0.50 de la tasa de fallos, debido a que se han calculado tanto los falsos positivos como los falsos negativos con respecto al número total de errores sin tener en cuenta la proporción de los aciertos obtenidos por el clasificador. En la siguiente gráfica sí se tienen en cuenta los aciertos por lo que el “*equal error rate*” ya no es el mismo para todas las combinaciones.

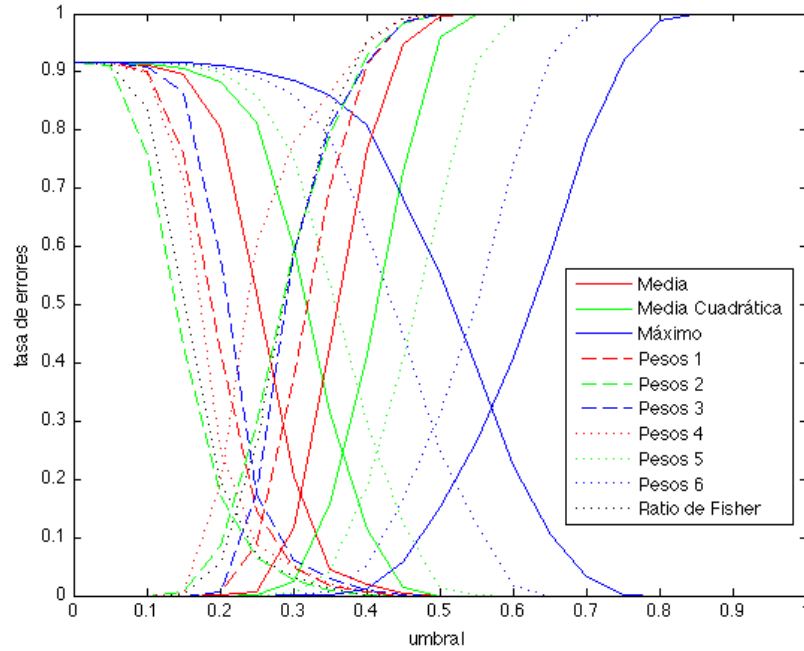


Figura 5.22: Representación gráfica de los falsos negativos junto con los falsos positivos para cada una de las 10 combinaciones, teniendo en cuenta la proporción de aciertos del clasificador. En el eje de las abscisas aparece el umbral utilizado, mientras que en el eje de las ordenadas se corresponde a la tasa de errores.

Para concluir se muestra una gráfica en la que se representan los falsos positivos en función de los falsos negativos para conocer el *equal error rate* de cada una de las combinaciones estudiadas. El clasificador con mejores resultados es aquel cuyo valor de *equal error rate* sea menor, por lo que en la figura 5.23 donde la recta de la función  $x = y$  indica este valor para cada combinación.

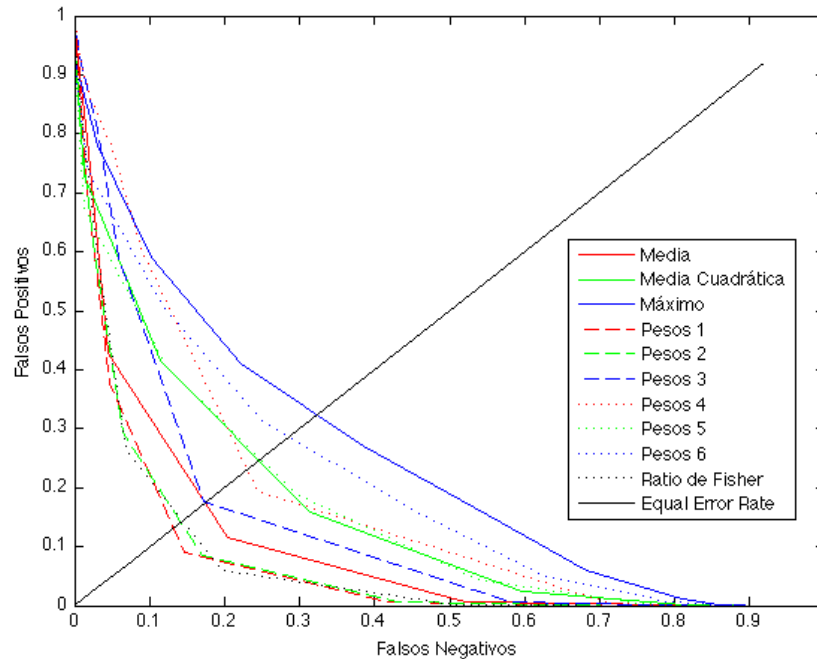


Figura 5.23: Representación gráfica de los falsos positivo en función de los falsos negativos para cada una de las 10 combinaciones, teniendo en cuenta la proporción de aciertos del clasificador. En el eje de las abcisas representa a los falsos negativos, mientras que en el eje de las ordenadas se corresponde con los falsos positivos.

Por lo que según la proporción de falsos positivos y negativos en la figura 5.23 es la combinación basada en la primera distribución de pesos la que mejor relación de falsos positivos y negativos posee, mientras que la combinación con la peor proporción es la combinación basada en los máximos.

## 5.5. Análisis cuantitativo de la clasificación

Los resultados aportados en los apartados anteriores por las gráficas de los distintos tipos de errores posibles, resumen la capacidad de clasificación de las características estudiadas de forma general. Pero con esos datos no se conoce cuál es el porcentaje de fallos para una determinada categoría y

tampoco especifica con qué imágenes de otra clase se equivoca, a lo largo del proceso.

Es necesario obtener resultados cuantitativos para todos los casos analizados en el apartado anterior en el que se obtenía una única matriz a partir de cada una de las 10 ecuaciones que combinaban las 3 propiedades analizadas de las imágenes. Ello servirá para poder realizar un análisis más preciso y así poder establecer de todos los datos aportados, cuál es el que proporciona una mejor clasificación.

Para ello se procede a calcular la matriz de confusión en la que se indica el número de aciertos para cada clase y el número de imágenes de otras clases que se ha clasificado erróneamente como perteneciente a esa clase. Esta matriz se calcula ejecutando el script `calcularMatrizConfusion.m`. A continuación, el pseudocódigo de la función.

**Data:** La matriz de distancia tras aplicar el umbral

**Result:** La matriz de confusión

```

for clase = 1 hasta clase = totalClases do
    contador = 0;
    for i = filaInicioClase hasta i = filaFinClase do
        for j = 1 hasta j = numColumnasMatriz do
            if matrizEntrada(i, j) == 1 then
                claseImagen =  $\frac{j}{totalClases}$ ;
                contador = contador + 1;
                matrizConfusion(clase, claseImagen) =
                    matrizConfusion(clase, claseImagen) + 1;
            end
        end
    end
    matrizConfusion(clase, 1 : totalClases) =
         $\frac{matrizConfusion(clase, 1 : totalClases)}{contador}$ ;
end

```

**Algorithm 9:** Pseudocódigo para obtener la matriz de confusión

La dimensión de esta matriz de confusión es de  $17 * 17$ , el número de clases diferentes de la base de datos estudiada. Aunque la función se utiliza para obtener la matriz de confusión para cada umbral de 0.05 desde 1 descendiendo hasta cero, para cada ecuación, es más representativo mostrar

los resultados para un valor de umbral en el que se hayan eliminado conexiones entre clases distintas. Por ello, tras evaluar el comportamiento de las diferentes ecuaciones de combinación, se ha decidido mostrar las matrices de confusión de todas las ecuaciones con un valor de umbral de 0.30, para el cual, como se verá a continuación los peores resultados de clasificación son para las 4 primeras ecuaciones de distribución de pesos, mientras que para el resto se obtiene una tasa de aciertos más elevada.

### 5.5.1. Datos cuantitativos de la combinación basada en los máximos

Para el caso de esta combinación de la cual se muestra en la tabla 5.5 del apartado 5.5.11 su matriz de confusión, para un valor de umbral de 0.30 tiene una tasa de error global del 0.05.

A este valor de umbral, hay varias clases para las cuales el clasificador no funciona correctamente. Empezando por la clase 6 donde comete un error del 0.08, clasificando como de esta clase una imagen de la clase 12. En el caso de la clase 12 el fallo es mayor ya que se agrupa como perteneciente a esta categoría una imagen de la clase 6 y un 0.17 de los errores para clasificar esta clase se debe a que se reconocen como del mismo grupo dos imágenes de la clase 15, donde los valores de distancia de algunas imágenes de la clase “key” han sido de forma sorprendente, muy similares a los de esta clase. Y para el caso contrario, en la clase 15 se comete un 0.25 de error de clasificación porque se incluyen como pertenecientes a esta categoría 3 imágenes de la clase 12.

Entonces para afinar el reconocimiento de esas clases hay que hacer más estricto el valor umbral, para eliminar conexiones incorrectas. Para la ecuación de los máximos, a partir de un valor de un umbral de 0.40 o inferior realiza una clasificación con muy buenos resultados, mientras que para un umbral entre 0.45 y 0.65 tiene problemas en clasificar principalmente las clases 6, 8, 12 y 15, donde el porcentaje de aciertos para cada una de estas ecuaciones no es superior al 0.60. Siendo el peor caso para la clase 6 de la cual sólo clasifica bien el 0.46. El salto en mejoras que experimenta este clasificador al reducir el valor del umbral al realizar el escalado se nota de una manera significativa si se compara entre un valor de umbral 0.75 (con

porcentaje de 0.87 fallos) frente al valor de 0.65 (con un total de 0.56 fallos) donde se reduce en un 0.31 la tasa de fallos

Pero con un valor de umbral de 0.30, se comporta como un buen clasificador porque consigue identificar correctamente todas las imágenes para 14 clases de las 17 que se están estudiando, y sólo en las clases 12 y 15 tiene una tasa de fallos mayor.

### 5.5.2. Datos cuantitativos de la combinación basada en la media

La tabla 5.6 del apartado 5.5.11 muestra la matriz de confusión para la ecuación media, la cual para un valor de 0.30 tiene una tasa de errores global del 0.12. Por los datos aportados por la tabla 5.6, se observa que la combinación mediante la media para un valor de umbral en las conexiones de 0.30 obtiene unos resultados peores que los dados por el caso basado en el máximo.

Sólo consigue clasificar correctamente las clases 2 y 13 sin incluir en alguna de estas categorías imágenes de otras clases. Sólo para la primera clase con un total de aciertos del 0.67, consigue una clasificación con un menor número de fallos respecto al resto de las categorías donde el número de aciertos disminuye en gran medida.

El siguiente caso mejor clasificado es para la clase 5, donde el 0.50 de las imágenes se han clasificado correctamente, mientras que el resto de imágenes se divide entre varias clases, de las cuales con las que el clasificador ha cometido un error del 0.42 al considerar como pertenecientes a la clase 5, imágenes de la categoría 17.

La clase 12, sólo agrupa correctamente 2 de las 12 imágenes, siendo el caso donde menor cantidad de imágenes de su clase consigue catalogar como suyas propias, e incluso el número de imágenes de las clases 3 y 4 es el mismo al de las imágenes que sí ha clasificado bien, por lo que este clasificador no trabaja adecuadamente para este grupo de datos.

En el resto de las categorías clasificadas consigue identificar imágenes que

se corresponden con el tipo correcto entre los porcentajes del 0.20 hasta el 0.50 mientras que el resto se dividen entre imágenes de otras clases.

Si se disminuye el umbral hasta 0.20, la tasa de fallos disminuye significativamente del 0.12 al 0.04, donde ahora se consigue clasificar las imágenes de forma acertada para la mayoría de las clases. Pero si se observa su comportamiento de clasificación con un valor de umbral del 0.30, se observa como sus resultados son muy inferiores a los conseguidos mediante la combinación basada en los máximos evaluada en el apartado anterior, porque tiene una tasa de fallos de 0.40 frente al 0.05 conseguido por la ecuación de los máximos. La afirmación anterior choca con lo expuesto en la sección donde se analizaba visualmente los resultados dados por esta ecuación, ya que todo parecía indicar que la distancia de las imágenes pertenecientes a una clase tenían un valor menor frente al resto de clases, pero realizando una evaluación cuantitativa se ha demostrado que dichas conclusiones no son acertadas.

### 5.5.3. Datos cuantitativos de la combinación basada en la media cuadrática

La tabla 5.7 del apartado 5.5.11 es la correspondiente a la matriz de confusión referente a la combinación de los datos de la distancia aportados por cada una de las 3 características mediante la media cuadrática, que tiene una tasa total de fallos del 0.06.

Este clasificador consigue mejores resultados que para el caso de la combinación basada en las medias, pero inferiores al caso de los máximos. Consigue clasificar perfectamente las imágenes de las clases 2, 5, 11 y 13. Mientras que las clases en las que encuentra más dificultad en clasificar son la 8, 10, 12 y 17 donde la cantidad de imágenes asociadas correctamente asciende entre 4 y 5 unidades dependiendo de la clase y en ninguno de estos casos se agrupan más de 2 imágenes de clases incorrectas. Los peores resultados de este análisis son mejores frente a los vistos en la combinación de las medias, porque siempre se agrupan bien una mayor cantidad frente al resto de imágenes de otras clases que se incluyen dentro del mismo grupo.

En el caso de la clase 8, los valores de las distancias de algunas imágenes



de las clases 6, 10 y 12 son muy cercanos a los de la propia clase, por ello son imágenes de estos tipos los que incluye en este conjunto. Mientras que al clasificar imágenes de la clase 10, confunde como pertenecientes a este grupo, sobre todo un 0.25 de imágenes de la clase 6 y un 0.17 del total analizado para la clase 10 se confunde con imágenes de la clase 8.

Para identificar imágenes de la clase 17, también encuentra problemas al distinguir imágenes de la clase 8, principalmente. Por tanto hay que considerar que para el caso que se está evaluando actualmente, la clase con las que más dificultades tiene para realizar una correcta clasificación es con la clase 8.

El tipo 6 como última clase donde la tasa es más baja que el resto (0.42 del total clasificado), también tiene problemas en reconocer como distintas de la clase, imágenes de las clases 8, 10 y 12, donde considera pertenecientes a su grupo 2 imágenes de cada una de estas clases. Para el resto de clases tiene un número más elevado de aciertos, siempre con una tasa superior al 0.50.

Este clasificador con umbral del 0.25 se disminuye la cantidad de errores al mínimo, por lo que consigue clasificar de forma adecuada. De forma global para un umbral de 0.30 la ecuación de la media cuadrática consigue una tasa de fallos del 0.06, aunque sólo realice una clasificación perfecta para 4 clases, siendo este un resultado muy superior frente al que ofrece cuando se aplica un valor de umbral de 0.40 al aplicar el escalado, donde la tasa de errores se sitúa en un 0.40. Y como para el resto de casos estudiados, si su umbral se reduce a 0.25 consigue elevar el número de aciertos significativamente, aproximándose a 1, para todas las clases. Lo que la convierte en una de las posibles candidatas para realizar la clasificación.

#### **5.5.4. Datos cuantitativos de la combinación basada en la distribución de pesos primera**

La matriz de confusión de la combinación basada en la distribución de pesos primera con un valor de umbral del 0.30, aparece en la tabla 5.8 del apartado 5.5.11.

Este es uno de los clasificadores con la tasa de aciertos más bajos de todos los estudiados a lo largo del proyecto porque cuenta con una tasa de fallos del 0.36. En la mayoría de las categorías clasifica pocas imágenes como pertenecientes a la misma, entre un rango del 0.10 al 0.20, es decir entre 1 y 3 imágenes, y clasificando de forma equivocada como pertenecientes a una clase, imágenes de casi todas las otras clases. Siendo los peores resultados cuando clasifica las clases 8 y 12, donde sólo un 0.08 de las imágenes de esta clase son identificadas correctamente, reconociendo erróneamente como de la clase propia, imágenes del resto de tipos.

Para la clase que consigue un mejor resultado es para la primera clase, donde el 0.75 de todas las imágenes son reconocidas como pertenecientes a esta clase, de verdad lo son, confundiendo principalmente con imágenes de la clase 4 y 8. El siguiente mejor resultado, el de la clase 13, que se sitúa con un 0.42 de las imágenes clasificadas correctamente, mientras que las clases que faltan por comentar se distribuyen con unos resultados similares, no siendo superiores al 0.25 de clasificación correcta, frente a las imágenes de las otras clases.

Para esta combinación, consigue unos resultados bastante aproximados para la mayoría de las clases que se están estudiando para un umbral del 0.20, con el cual el nivel de precisión aumenta de una forma más notable que para el resto de ecuaciones, porque consigue una de las tasas más bajas de errores, en concreto para el umbral 0.20 tiene sólo un 0.03 de errores, aunque con valor de umbral 0.15 se estabiliza a un total de errores del 0.045. Y con un valor de 0.15 las imágenes clasificadas se agrupa cada una a su categoría apropiada. Pero para el valor de umbral de 0.30 que es para el que se está realizando la comparativa, obtiene una tasa de fallos global del 0.36, valor que la aleja de los mejores casos estudiados hasta ahora.

Estos resultados indican que aunque la propiedad de circularidad sea la que mejor se comporte de manera individual si se le aplica un mayor peso a ella no se van a obtener unos resultados más precisos, por lo tanto no es de las mejores ecuaciones de combinación para lograr un buen clasificador, si no es aplicando la medida restrictiva de reducir el umbral, pero como se quiere conservar el mayor número de conexiones entre imágenes esta opción queda pospuesta como una de las últimas posibilidades.

### 5.5.5. Datos cuantitativos de la combinación basada en la distribución de pesos segunda

La matriz de confusión de la ecuación de distribución de pesos segunda con un valor de umbral de 0.30 al realizar el escalado de la matriz, se muestra en la tabla 5.9 del apartado 5.5.11.

En este caso como se está dando más peso a la característica de circularidad (en concreto un 0.80 del total), como anteriormente se ha mencionado, la circularidad de forma aislada con los resultados conseguidos se concluía que era la que mejor característica con la que trabajar de forma aislada. Ahora analizando su matriz de confusión, se observa que se obtiene unos resultados peores a la anterior ecuación de distribución de pesos.

En este caso la tasa de imágenes bien agrupados para una clase, no supera en la mayoría de casos el 0.08 por lo que se produce un resultado que no se esperaba antes de obtener los datos mostrados en la tabla anterior. Aunque una de las clases para las que obtiene el segundo mejor resultado es la 9, donde en el caso anterior estudiado era la que peor resultado obtenía y ahora alcanza un 0.17 de las imágenes clasificadas, al reconocer de forma correcta dos imágenes de este grupo.

Y la clase para la que obtiene mejores resultados sigue siendo con la primera clase, pero ahora el porcentaje de aciertos se ha reducido hasta la mitad, porque es capaz de agrupar correctamente 6 de las 12 imágenes asociadas a esta categoría, mientras que el 50 % restante se reparte entre figuras de múltiples categorías.

Para obtener un buen resultado, se tiene que restringir el valor de umbral hasta un 0.20 para obtener una clasificación aceptable para la mayoría de las categorías, con una tasa de fallos general del 0.09, aunque para algunas clases como la 3, 7, 11 y 16 el porcentaje de aciertos sigue siendo demasiado bajo, debido a que ninguna de estas clases supera el 0.33 de aciertos. Para conseguir mejorar los resultados de clasificación es necesario reducir el valor del umbral a 0.15 a la hora de realizar el escalado, de esta forma se consigue una tasa de errores global muy baja, donde la peor clase catalogada tiene una tasa de fallos del 0.50.

Los datos estudiados para esta ecuación, se resumen en que para el umbral de 0.30 estudiado para las diez ecuaciones, obtiene una tasa de 0.56 de fallos cometidos de manera general para clasificar imágenes, obteniendo peores resultados que cuando se le daba un 0.60 del peso total a la característica de circularidad. La tasa de aciertos no mejora significativamente, como se ha dicho en el párrafo anterior, hasta que se aplican medidas restrictivas con un umbral de 0.15, con el cual se consigue que la tasa general de fallos no supere el 0.10 pero esto supone perder conexiones con imágenes de la misma categoría que posean un valor de distancia superior a este umbral, por lo que se consigue reducir la posibilidad de que el clasificador incluya dentro de una categoría imágenes pertenecientes a otras, pero a cambio se incrementan las posibilidades de excluir alguna imagen que sí pertenece a ese conjunto pero tiene alguna diferencia mayor con el resto que hace que el valor de distancia frente al resto sea mayor.

Como conclusión se demuestra la suposición de que aplicar un mayor peso sólo a una de las características provoca que los aciertos en la clasificación de cada una de las categorías disminuya, aunque esta propiedad ofreciera buenos resultados de forma individual.

#### **5.5.6. Datos cuantitativos de la combinación basada en la distribución de pesos tercera**

La matriz de confusión de la combinación basada en la distribución de pesos tercera con un valor de umbral de 0.30, se muestra en la tabla 5.10 del apartado 5.5.11.

En este caso, la ecuación daba una importancia del 0.60 a la propiedad de grado frente al resto de características y ya desde los primeros análisis se estableció que no era una propiedad muy adecuada para la clasificación de forma aislada.

Aunque en general es peor que en las ecuaciones de distribución de pesos que dan más importancia a otras características, es decir aquellas en las que se aplica un 0.80 a una de las 3 propiedades. Entre sus mejores resultados de clasificación, se observa que para la clase 13 consigue catalogar un 0.92 de

las imágenes correctamente. Y para la segunda clase consigue obtener una tasa del 0.42 de imágenes bien agrupadas. Los 2 casos expuestos antes son para los que mejor se comporta este clasificador, mientras que para el resto de las clases no supera en ningún caso el 0.08 de imágenes correctamente agrupadas, es decir sólo una imagen bien catalogada por grupo, clasificando en cada grupo o categoría una imagen de casi todo el resto de clases.

Esta ecuación tiene una tasa de fallos de 0.56, tan alta como en el caso anterior, donde se aplicó el 0.80 del peso total a la característica de la circularidad. La afirmación anterior respalda lo dicho anteriormente, es decir, que la propiedad de grado es la menos indicada para soportar el mayor peso a la hora de realizar la clasificación con respecto a las otras características, por lo que un buen clasificador contará con una distribución de peso bajo para esta propiedad.

Para mejorar esta clasificación hay que reducir el umbral, si se restringe hasta un 0.20, el clasificador reduce notablemente el número de errores cometidos al realizar la clasificación y con un valor de umbral de 0.15 consigue clasificar correctamente las imágenes que analiza. Pero como en todos los casos, hay que tener sólo en cuenta esta ecuación para el umbral que se está estudiando de 0.30, por la razón que se ha comentado en varias ocasiones, donde a un valor de umbral bajo se restringe de tal forma la distancia que provoca que se excluyan del grupo imágenes pertenecientes a el mismo.

### **5.5.7. Datos cuantitativos de la combinación basada en la distribución de pesos cuarta**

La matriz de confusión de la combinación basada en la cuarta distribución de pesos con un valor de umbral de 0.30, se muestra en la tabla 5.11 del apartado 5.5.11.

Esta tabla se corresponde a la matriz de confusión de la ecuación a la que se distribuye el 0.80 del peso total a la propiedad de grado. Como se puede observar, los resultados de clasificación obtenidos no son suficientes para clasificar de manera aceptable las imágenes de una clase en su lugar correspondiente. Porque para cada clase, las imágenes que se agrupan en ella

son una distribución de imágenes del resto de las clases y en la mayoría de los casos sólo se incluye en el grupo una imagen correcta (el 0.08 del total) de las doce agrupadas como correspondiente a la clase y este resto de imágenes pertenecen todas ellas a clases diferentes. Estos casos donde el porcentaje de imágenes correctamente clasificadas es igual o inferior al porcentaje de imágenes de otras clases clasificadas como pertenecientes a esta, indica que este clasificador es el peor de todos los evaluados a lo largo del proyecto.

La clase 13, es la única para la que se consigue una tasa de aciertos del 0.58, con el cual se puede realizar una clasificación aproximada de esa clase en la mitad de los casos, confundiéndola en el resto de análisis principalmente con la clases 7 con un 0.25 del total de las imágenes agrupadas y la clase 11 con un 0.17 del total de figuras agrupadas en la clase 13. La otra clase con el mayor número de aciertos frente al resto, es muy bajo como para conseguir clasificar mínimamente la clase 2, ya que sólo identifica un 0.17 de imágenes de esa clase.

Por tanto la característica de grado, tanto aislada como proporcionándola un valor de peso alto, no son las alternativas más adecuadas para obtener una buena clasificación. Porque con un valor de umbral de 0.30, la tasa de error global es del 0.74, teniendo que adoptar medidas muy restrictivas para conseguir que las imágenes que se analizan se clasifiquen en su clase correspondiente. Y sólo se consigue resultados aceptables con un límite de 0.15 en el valor de las conexiones, donde con este umbral se consigue reducir la cantidad total de errores a un 0.20. Pero como en todos los casos hay que tener en cuenta el balance entre incluir figuras de otras clases y la posibilidad de excluir imágenes de la propia clase, aunque para este clasificador se observa que es necesario hacer más restrictivo el valor del umbral para conseguir disminuir el número de imágenes que se incluyen erróneamente como pertenecientes a una clase.

### **5.5.8. Datos cuantitativos de la combinación basada en la distribución de pesos quinta**

La matriz de confusión de la combinación basada en la distribución de pesos quinta con un valor de umbral de 0.30, se muestra en la tabla 5.12 del

apartado 5.5.11.

La tabla 5.12 se corresponde, por tanto, con la matriz de confusión de la combinación de distribución de pesos, al que se le aplica un 0.60 del total a la propiedad *joint degree* y el peso restante se reparte de forma equitativa entre las otras 2 propiedades.

En un primer análisis de los datos obtenidos para esta ecuación se observa que para cada clase clasifica un porcentaje muy elevado de imágenes de la clase correcta, por lo que tiene una tasa global de error muy bajo de un 0.05. Aunque no consigue igualar el alto número de clases que consigue clasificar perfectamente el caso de la combinación basado en los máximos, sí consigue clasificar de forma totalmente correcta 6 clases frente a las 14 clases que la ecuación de los máximos consigue agrupar bien del todo.

Algunas de las clases con las que más problemas tiene en clasificar imágenes, son la clase 6, 12 y 15 en las cuales consigue clasificar entre un 0.58 y 0.67 (6 o 7 figuras) de todas las imágenes de manera correcta. Y de estas 3 clases, la mayoría de los fallos de clasificación es debido a que entre cada una de ellas el porcentaje restante de figuras clasificadas se agrupa principalmente con imágenes de las otras dos clases. Por tanto el clasificador encuentra sus mayores problemas en agrupar estas 3 clases, aunque en estos casos concretos se comete un nivel de errores menor que en casos estudiados previamente.

En el resto de clases se consigue una tasa de aciertos que oscila entre un 0.75 y 0.92, e incluso consigue clasificar perfectamente todas las imágenes de la clase 2, 5, 7, 11, 13 y 16. Pero aunque obtiene una media de aciertos muy elevada no consigue superar a los resultados ofrecidos por el caso de los máximos con el mismo valor de umbral, tal y como se ha dicho previamente, para superarlo debe reducir el umbral de conexiones permitidas en 0.05 unidades, consiguiendo de esta manera una tasa de fallos menor donde se consigue clasificar correctamente 15 de las 17 clases evaluadas en este proyecto.

Los resultados expuestos en la matriz de confusión anterior contradicen las suposiciones que se habían establecido con respecto a esta característica donde se suponía que al tener un valor de distancia general para todas las imágenes de la base de datos podía suponer una desventaja en el momento de realizar la clasificación, pero es justamente este valor de distancia más

elevado entre figuras de clases distintas, lo que permite que con un valor de umbral de 0.30 se puedan excluir una gran cantidad de imágenes pertenecientes a otras clases, sin tener que descartar al mismo tiempo imágenes que verdaderamente sí corresponden al grupo actual.

### 5.5.9. Datos cuantitativos de la combinación basada en la distribución de pesos sexta

La matriz de confusión de la combinación mediante la combinación de la sexta distribución de pesos con un valor de umbral de 0.30, se muestra en la tabla 5.13 del apartado 5.5.11.

Ahora se analiza la matriz de confusión para la ecuación en la que se aplica un 0.80 del peso total a la característica del *joint degree*. Antes de estudiar esta matriz se puede esperar que se consiga obtener unos resultados de clasificación inferiores al caso anterior donde se aplicaba un 0.60 a esta propiedad, basándose principalmente en los 2 casos anteriores en los que se aplicaba a las otras propiedades el 0.80. Pero en este caso en vez de verse aumentada la tasa de fallos, esta se mantiene, e incluso para valores de umbral más alto como es en el caso del valor de umbral 0.40, la cantidad de errores es inferior en esta última ecuación de distribución. Por tanto puede considerarse la propiedad de *joint degree* como la que mejor resultado consigue combinadas con las otras características en menor medida, de tal forma, que este resultado niega lo expuesto en anteriores apartados, ya que es la excepción en las funciones de combinación donde aplicando mayor importancia a una sólo característica se consiguen mejores resultados.

Con esta distribución se observa que la tasa de aciertos se eleva en general, obteniendo ahora un mayor número de clases para las que se obtiene una clasificación completamente correcta, en concreto para 11 de las 17 clases estudiadas en total, por lo que se ha aumentado en 5 clases la capacidad de clasificar dichos tipos correctamente. Y las clases 1 y 9 tienen un porcentaje de aciertos muy elevado de un 0.92, mientras que las clases que siguen teniendo una tasa de aciertos más baja, ha crecido hasta un 0.67 (reconocen 8 figuras de su clase correctamente) siguen siendo las clases 6, 12 y 15.



La tasa global de fallos es muy bajo, con sólo un 0.05 y si se reduce el umbral a 0.25 se consigue clasificar correctamente todas las figuras que analiza a su clase correspondiente. Por lo que con los datos obtenidos convierten a esta, en la segunda mejor opción de todos los clasificadores analizados. Este hecho se debe por tanto, a la característica *joint degree*, si se analiza la otra ecuación de combinación que ofrece los mejores resultados (combinación basada en los máximos), se llega a la conclusión de que en la mayoría de los valores que selecciona para construir la matriz que reúne a las 3 propiedades se corresponde con la propiedad de *joint degree* que es la que consta de forma general con unos valores de distancia superiores al resto. El hecho de poseer un valor de distancia más elevado en general es lo que consigue que a valores más altos de umbral se puedan eliminar conexiones con figuras pertenecientes a otras clases, conservando las uniones con las imágenes del mismo tipo que tienen un valor de distancia inferior, consiguiendo como resultado el mayor número de clases bien agrupadas.

#### 5.5.10. Datos cuantitativos de la combinación basada en la distribución de pesos dada por el ratio de Fisher

La combinación basada en la séptima distribución de pesos se corresponde con los valores calculados por la función del *ratio de Fisher*, cuya matriz de confusión para un valor de umbral de 0.30 se encuentra en la tabla 5.14 del apartado 5.5.11.

Por los resultados que se muestran en la tabla 5.14, se observa que esta distribución de pesos no es la más acertada porque en ninguna de las clases de la base de datos se consigue clasificar más de dos imágenes como correspondientes a la clase a la que realmente pertenecen. Estos datos se deben principalmente a que el *ratio de Fisher* otorga un 0.67 del peso total a la característica de la circularidad, un 0.21 a la propiedad de grado y de forma sorprendente sólo un 0.12 a la propiedad de la distribución *joint degree*, propiedad que en los casos en los que se le proporcionaba un mayor peso en las funciones de distribución anteriores, conseguía unos resultados de clasificación superiores frente al resto de funciones de combinación excepto para el caso del máximo que continúa siendo la que proporciona un mejor resultado,

pero de la que ya se ha dicho en el apartado anterior que está directamente influenciada por la propiedad *joint degree*.

Al variar ligeramente los pesos repartidos frente a las dos primeras funciones de distribución, la tasa de fallos que se consigue según se va variando el umbral se sitúa a un valor intermedio entre ambas ecuaciones, aunque lógicamente se encuentra más cercano a la que se aplica un 0.80 del peso total a la propiedad de circularidad e incluso en determinados momentos se comporta peor debido a que se ha disminuido aún más la propiedad *joint degree* que es la que determina el alto número de aciertos en el momento de clasificar las figuras.

#### **5.5.11. Matrices de confusión de las 10 combinaciones estudiadas**

En este apartado se recopilan las matrices de confusión de todas las combinaciones que se han analizado anteriormente, a continuación se muestran en el mismo orden en el que se ha realizado el estudio de cada una de ellas.

	c 1	c 2	c 3	c 4	c 5	c 6	c 7	c 8	c 9	c 10	c 11	c 12	c 13	c 14	c 15	c 16	c 17
c 1	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c 2	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c 3	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c 4	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0
c 5	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0
c 6	0	0	0	0	0	11	0	0	0	0	0	1	0	0	0	0	0
c 7	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0
c 8	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0
c 9	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0
c 10	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0
c 11	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0
c 12	0	0	0	0	0	1	0	0	0	0	0	9	0	0	2	0	0
c 13	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0
c 14	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
c 15	0	0	0	0	0	0	0	0	0	0	0	3	0	0	9	0	0
c 16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0
c 17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12

Cuadro 5.5: Matriz de confusión para el caso de la combinación mediante el máximo con un valor de umbral de 0.30

	c 1	c 2	c 3	c 4	c 5	c 6	c 7	c 8	c 9	c 10	c 11	c 12	c 13	c 14	c 15	c 16	c 17
c 1	8	0	0	1	0	1	0	1	0	1	0	0	0	0	0	0	0
c 2	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c 3	0	0	4	2	0	1	0	0	0	1	0	3	0	0	1	0	0
c 4	1	0	1	3	0	1	0	1	0	1	0	2	0	0	0	1	1
c 5	0	0	0	0	6	0	0	0	0	0	0	0	0	1	0	0	5
c 6	0	0	0	1	0	3	0	2	0	2	0	2	0	1	1	0	0
c 7	0	0	0	1	0	0	4	0	0	0	4	0	0	0	0	1	2
c 8	0	0	0	1	0	1	0	3	1	1	0	1	0	2	1	0	1
c 9	0	0	0	0	0	0	0	2	6	0	0	2	0	0	2	0	0
c 10	0	0	0	1	0	2	0	2	0	2	0	1	0	1	1	1	1
c 11	0	0	0	0	0	0	4	0	0	0	6	0	0	0	0	1	1
c 12	0	0	2	2	0	1	0	1	0	1	0	2	0	0	1	1	1
c 13	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0
c 14	0	0	0	0	0	1	0	3	0	1	0	1	0	4	0	0	2
c 15	0	0	0	1	0	2	0	1	2	1	0	2	0	0	3	0	0
c 16	0	0	1	1	0	0	1	0	0	1	1	3	0	0	0	2	2
c 17	0	0	0	1	1	0	1	1	0	1	0	1	0	2	0	1	3

Cuadro 5.6: Matriz de confusión para el caso de la combinación basado en la media con un valor de umbral de 0.30

	c 1	c 2	c 3	c 4	c 5	c 6	c 7	c 8	c 9	c 10	c 11	c 12	c 13	c 14	c 15	c 16	c 17
c 1	11	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0
c 2	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c 3	0	0	7	0	0	1	0	0	0	0	0	3	0	0	1	0	0
c 4	0	0	0	7	0	1	0	1	0	1	0	0	1	0	0	0	1
c 5	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0
c 6	0	0	0	0	0	5	0	2	0	2	0	2	0	0	1	0	0
c 7	0	0	0	0	0	0	10	0	0	0	0	0	0	0	0	0	2
c 8	0	0	0	1	0	2	0	4	0	1	0	1	0	1	1	0	1
c 9	0	0	0	0	0	0	0	1	9	0	0	1	0	0	1	0	0
c 10	0	0	0	1	0	3	0	2	0	4	0	1	0	0	0	0	1
c 11	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0
c 12	0	0	1	1	0	2	0	2	0	0	0	4	0	0	2	0	0
c 13	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0
c 14	0	0	0	0	0	0	0	2	0	1	0	0	0	8	0	0	1
c 15	0	0	0	0	0	1	0	1	1	1	0	2	0	0	6	0	0
c 16	0	0	0	0	0	0	0	0	0	2	0	1	0	0	0	8	1
c 17	0	0	0	1	0	0	0	2	0	1	0	1	0	2	0	0	5

Cuadro 5.7: Matriz de confusión para el caso de la combinación basada en la media cuadrática con un valor de umbral de 0.30

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17
c1	9	0	0	1	0	0	0	1	0	1	0	0	0	0	0	0	0
c2	0	2	1	1	0	0	2	1	0	1	1	1	0	0	1	0	1
c3	0	0	7	0	0	1	0	0	0	1	1	1	0	0	1	0	0
c4	0	1	1	1	0	1	1	1	0	1	1	1	0	0	1	1	1
c5	0	0	0	0	3	0	1	0	2	0	1	0	0	2	1	0	2
c6	0	0	2	1	0	2	1	1	0	2	0	1	0	0	1	1	0
c7	0	1	1	1	1	0	1	1	0	1	1	1	0	1	0	1	1
c8	0	1	1	1	0	1	1	1	1	1	0	1	0	1	1	0	1
c9	0	0	0	0	2	0	0	2	4	0	0	0	0	1	1	0	2
c10	0	0	1	1	0	1	1	1	0	1	1	1	0	1	1	0	2
c11	0	1	1	1	0	0	1	0	0	1	2	2	0	1	0	1	1
c12	0	1	1	1	0	0	1	1	0	1	1	1	0	1	1	1	1
c13	0	0	0	1	0	1	1	1	0	1	0	0	5	0	1	0	1
c14	0	0	0	0	0	1	1	2	0	0	1	1	0	2	1	1	2
c15	0	1	0	1	1	1	0	1	1	1	1	0	0	1	2	0	1
c16	0	0	1	1	0	1	1	0	0	1	2	2	0	0	0	2	1
c17	0	1	1	1	1	0	1	1	1	0	1	1	0	1	0	1	1

Cuadro 5.8: Matriz de confusión para el caso de combinación basada en la primera distribución de pesos con un valor de umbral de 0.30

	c 1	c 2	c 3	c 4	c 5	c 6	c 7	c 8	c 9	c 10	c 11	c 12	c 13	c 14	c 15	c 16	c 17
c 1	6	0	0	2	0	1	0	1	0	1	0	0	1	0	0	0	0
c 2	0	1	1	1	0	0	1	1	0	1	1	1	0	1	1	1	1
c 3	0	1	1	1	0	1	1	1	0	1	1	1	0	0	1	1	1
c 4	0	1	1	1	0	1	1	1	0	1	1	1	1	0	0	1	1
c 5	0	1	0	0	2	0	1	1	1	0	1	0	0	1	2	0	2
c 6	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0
c 7	0	1	1	1	0	1	1	1	0	1	1	1	0	0	1	1	1
c 8	0	1	1	1	0	0	1	1	1	1	1	1	0	1	0	1	1
c 9	0	0	0	0	2	0	1	2	2	0	0	0	0	2	1	0	2
c 10	0	1	1	1	0	1	1	1	0	1	1	1	0	0	1	1	1
c 11	0	1	1	1	0	1	1	1	0	1	1	1	0	1	0	1	1
c 12	0	1	1	1	0	1	1	1	0	1	1	1	0	1	0	1	1
c 13	0	1	1	1	0	1	2	1	0	1	1	0	2	0	1	0	0
c 14	0	1	1	0	1	0	1	1	1	0	1	1	0	1	1	1	1
c 15	0	1	1	0	1	1	1	1	0	1	0	1	1	1	1	0	1
c 16	0	1	1	1	0	1	1	1	0	1	1	1	0	1	0	1	1
c 17	0	1	1	0	1	0	1	1	1	0	1	1	0	1	1	1	1

Cuadro 5.9: Matriz de confusión para el caso de la combinación basada en la segunda distribución de pesos con un valor de umbral de 0.30

	c 1	c 2	c 3	c 4	c 5	c 6	c 7	c 8	c 9	c 10	c 11	c 12	c 13	c 14	c 15	c 16	c 17
c 1	1	0	1	1	1	1	0	1	1	1	0	1	0	1	1	0	1
c 2	0	5	0	0	0	1	1	2	0	1	0	0	0	1	1	0	0
c 3	1	0	1	1	0	1	1	1	1	1	0	1	0	0	1	1	1
c 4	1	0	1	1	1	1	0	1	1	1	1	1	0	0	0	1	1
c 5	1	0	0	1	1	0	1	1	1	1	1	1	0	1	0	1	1
c 6	1	0	1	1	0	1	1	1	1	1	0	1	0	1	1	0	1
c 7	0	0	0	1	1	1	1	1	1	1	1	1	0	1	0	1	1
c 8	1	0	0	1	1	1	1	1	1	1	0	1	0	1	1	0	1
c 9	1	0	0	1	1	1	1	1	1	1	0	1	0	1	1	0	1
c 10	0	0	1	1	0	1	1	1	1	1	0	1	0	1	1	1	1
c 11	0	0	1	1	1	1	1	1	0	1	1	1	0	1	0	1	1
c 12	0	0	1	1	0	1	1	1	1	1	1	1	0	1	0	1	1
c 13	0	0	0	0	0	0	1	0	0	0	0	0	11	0	0	0	0
c 14	0	0	0	0	1	1	1	1	1	1	1	1	0	1	1	1	1
c 15	1	0	1	1	0	1	1	1	1	1	0	1	0	1	1	0	1
c 16	0	0	1	1	1	1	1	1	0	1	1	1	0	1	0	1	1
c 17	0	0	1	1	1	0	1	1	1	1	1	1	0	1	0	1	1

Cuadro 5.10: Matriz de confusión para el caso de la combinación basada en la tercera distribución de pesos con un valor de umbral de 0.30



	c 1	c 2	c 3	c 4	c 5	c 6	c 7	c 8	c 9	c 10	c 11	c 12	c 13	c 14	c 15	c 16	c 17
c 1	1	0	0	1	1	1	1	1	1	0	1	1	0	1	0	1	1
c 2	0	2	0	1	1	1	1	1	0	1	1	1	0	1	1	0	0
c 3	1	0	1	1	1	1	1	1	1	0	1	1	0	0	0	1	1
c 4	1	0	1	1	1	1	1	1	1	0	1	1	0	1	0	0	1
c 5	1	0	1	1	1	0	1	1	1	0	1	1	0	1	0	1	1
c 6	1	0	1	1	1	1	1	1	1	1	0	1	0	1	0	1	0
c 7	1	0	1	1	1	0	1	1	1	0	1	1	0	1	0	1	1
c 8	1	0	1	1	0	1	0	1	1	0	1	1	0	1	1	1	1
c 9	1	0	1	1	1	1	1	1	1	0	1	1	0	1	0	1	0
c 10	1	0	0	1	1	1	1	1	1	1	1	1	0	1	0	1	0
c 11	1	0	0	1	1	0	1	1	1	1	1	1	0	1	0	1	1
c 12	1	0	1	1	1	1	1	1	1	0	1	1	0	1	0	1	0
c 13	0	0	0	0	0	0	3	0	0	0	2	0	7	0	0	0	0
c 14	1	0	0	1	1	1	1	1	1	0	1	1	0	1	0	1	1
c 15	1	0	0	0	1	1	1	1	1	1	0	1	0	1	1	1	1
c 16	1	0	1	1	1	1	1	1	1	0	1	1	0	1	0	1	0
c 17	0	0	1	1	1	1	1	1	1	0	1	1	0	1	0	1	1

Cuadro 5.11: Matriz de confusión para el caso de la combinación basada en la cuarta distribución de pesos con un valor de umbral de 0.30

	c 1	c 2	c 3	c 4	c 5	c 6	c 7	c 8	c 9	c 10	c 11	c 12	c 13	c 14	c 15	c 16	c 17
c 1	11	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
c 2	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c 3	0	0	10	0	0	1	0	0	0	0	0	1	0	0	0	0	0
c 4	0	0	0	11	0	0	0	0	0	0	0	1	0	0	0	0	0
c 5	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0
c 6	0	0	0	0	0	8	0	1	0	1	0	1	0	0	1	0	0
c 7	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0
c 8	0	0	0	0	0	1	0	9	0	0	0	1	0	0	1	0	0
c 9	0	0	0	0	0	0	0	0	11	0	0	1	0	0	0	0	0
c 10	0	0	0	0	0	1	0	1	0	9	0	1	0	0	0	0	0
c 11	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0
c 12	0	0	0	0	0	1	0	1	0	0	0	8	0	0	2	0	0
c 13	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0
c 14	0	0	0	0	0	0	0	1	0	0	0	0	0	11	0	0	0
c 15	0	0	0	0	0	2	0	1	0	0	0	2	0	0	7	0	0
c 16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0
c 17	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	0	10

Cuadro 5.12: Matriz de confusión para el caso de la combinación mediante la quinta distribución de pesos con un valor de umbral de 0.30

	c 1	c 2	c 3	c 4	c 5	c 6	c 7	c 8	c 9	c 10	c 11	c 12	c 13	c 14	c 15	c 16	c 17
c 1	11	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0
c 2	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c 3	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0	0
c 4	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0
c 5	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0	0	0
c 6	0	0	0	0	0	7	0	1	0	0	0	2	0	0	2	0	0
c 7	0	0	0	0	0	0	12	0	0	0	0	0	0	0	0	0	0
c 8	1	0	0	0	0	1	0	9	0	0	0	1	0	0	0	0	0
c 9	0	0	0	0	0	0	0	0	11	0	0	1	0	0	0	0	0
c 10	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0
c 11	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0
c 12	0	0	0	0	0	1	0	1	0	0	0	8	0	0	2	0	0
c 13	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0
c 14	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0
c 15	0	0	0	0	0	1	0	0	0	0	0	3	0	0	8	0	0
c 16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0
c 17	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12

Cuadro 5.13: Matriz de confusión para el caso de la combinación mediante la sexta distribución de pesos con un valor de umbral de 0.30

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16	c17
c1	7	0	0	2	0	1	0	1	0	1	0	0	0	0	0	0	0
c2	0	2	0	1	0	0	1	1	0	1	1	1	0	1	1	1	1
c3	0	0	1	1	0	1	1	1	0	1	1	1	0	1	1	1	1
c4	0	1	1	1	0	1	1	1	0	1	1	1	0	0	1	1	1
c5	0	1	0	0	2	0	1	1	1	0	1	1	0	1	1	0	2
c6	0	1	1	1	0	2	1	1	0	1	0	1	0	1	1	1	0
c7	0	1	1	1	0	0	1	1	0	1	1	1	0	1	1	1	1
c8	0	1	1	1	0	1	1	1	1	1	0	1	0	1	1	0	1
c9	0	0	0	0	2	0	0	2	3	0	0	0	0	2	1	0	2
c10	0	0	1	1	0	1	1	1	0	1	1	1	0	1	1	1	1
c11	0	1	1	1	1	0	1	1	0	1	1	1	0	1	0	1	1
c12	0	1	1	1	0	1	1	1	0	1	1	1	0	1	0	1	1
c13	0	1	0	1	0	1	2	1	0	1	0	0	4	0	1	0	0
c14	0	0	0	0	1	1	1	1	1	1	1	1	0	1	1	1	1
c15	0	1	0	1	1	1	1	1	1	1	0	1	0	1	1	0	1
c16	0	1	1	1	0	1	1	1	0	1	1	1	0	1	0	1	1
c17	0	1	1	1	1	0	1	1	1	0	1	1	0	1	0	1	1

Cuadro 5.14: Matriz de confusión para el caso de la función de distribución de pesos según el ratio de Fisher con un valor de umbral de 0.30

clasificador/umbral	0.20	0.25	0.30	0.35	0.40
Grado	0.5643	0.7167	0.7866	0.8526	0.9082
Circularidad	0.2451	0.4575	0.6681	0.7822	0.8732
Joint Degree	0.0536	0.0530	0.0524	0.0525	0.0608
Máximos	0.0536	0.0530	0.0523	<b>0.0523</b>	<b>0.0589</b>
Media	0.0472	<b>0.0366</b>	0.1212	0.4049	0.7239
Media cuadrática	0.0520	0.0486	0.0569	0.1681	0.3970
Pesos 1	<b>0.0301</b>	0.0925	0.3576	0.6632	0.8609
Pesos 2	0.0905	0.2807	0.5629	0.7469	0.8740
Pesos 3	0.0401	0.1757	0.5587	0.7626	0.8629
Pesos 4	0.1975	0.5756	0.7383	0.8163	0.8938
Pesos 5	0.0529	0.0507	<b>0.0475</b>	0.0723	0.1955
Pesos 6	0.0533	0.0525	0.0514	0.0534	0.0852
Ratio de Fisher	0.0666	0.2503	0.5575	0.7767	0.8991

Cuadro 5.15: Tabla con la tasa de fallos de todas las ecuaciones de combinación con el rango de umbral 0.20-0.40 en pasos de 0.05 unidades

### 5.5.12. Conclusión del análisis cuantitativo

Después de analizar los resultados de las funciones de combinación que daban una mayor prioridad a la propiedad de circularidad, los resultados obtenidos no son lo que se esperaba en un primer momento, ya que tanto con el primer análisis visual de los valores de distancia de la figura basados en la característica de circularidad como los valores proporcionados por el *ratio de Fisher* parecían indicar que el camino a seguir por el clasificador era otro. Es interesante ver como la distribución de pesos según esta medida apoya las suposiciones a las que se había llegado al finalizar el apartado de análisis inicial. A modo de resumen de todos los casos expuestos anteriormente, a continuación, se muestra como evoluciona la tasa de fallos en cada una de las combinaciones según el valor del umbral en el rango comprendido entre 0.20 y 0.40 ya que para la mayoría de los casos no experimentan una mejora notable hasta que se alcanzan dichos valores a la hora de realizar el escalado.

La tabla 5.15 muestra el rango de umbral donde se producen los cambios más significativos en la tasa de fallos, pero para obtener una visión global de la calidad del clasificador según el valor de umbral se muestra la tasa de fallos para el rango entre 0 y 1 en pasos de incremento de 0.10, donde se va a ver que nos se producen cambios significativos en la calidad de clasificación a partir de los umbrales cercanos a los extremos tanto a 1 como a 0.

En la tabla 5.16 la distribución de pesos de las diversas ecuaciones etiquetadas como *Pesos 1* hasta *Pesos 7* se corresponde con la misma distribución que la expuesta en la tabla 5.1. Como se puede observar en la tabla con un valor de umbral de 0.40 en la mayoría de las combinaciones que no consideran a la propiedad de *joint degree* como la más importante de las 3 características tienen una tasa de fallos cercana al 0.90, mientras que en el resto se consiguen para ese umbral unos valores más aceptables, tal y cómo se ha desarrollado en los apartados anteriores.

En el caso de la característica *joint degree* es la que mejores resultados ofrece a valores altos de umbral por ser la característica con los valores de distancia o diferencia entre imágenes más altos de todas las características, mientras que en el resto de casos es necesario reducir más el valor de umbral para que se eliminen conexiones erróneas y de esta forma reducir el número de errores. También hay que observar que a valores de umbral intermedios tienen mejores resultados otras funciones de combinación.

clasificador/umbral	0	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80	0.90	1
Grado	0.0539	0.0538	0.5643	0.7866	0.9082	0.9410	0.9412	0.9412	0.9412	0.9412	0.9412
Circularidad	0.0539	<b>0.0420</b>	0.2451	0.6681	0.8732	0.9380	0.9412	0.9412	0.9412	0.9412	0.9412
Joint Degree	0.0539	0.0539	0.0536	0.0524	0.0608	0.1771	<b>0.3989</b>	<b>0.7376</b>	<b>0.9311</b>	0.9412	0.9412
Máximos	0.0539	0.0539	0.0536	0.0523	<b>0.0589</b>	<b>0.1760</b>	<b>0.3989</b>	<b>0.7376</b>	<b>0.9311</b>	0.9412	0.9412
Media	0.0539	0.0537	0.0472	0.1212	0.7239	0.9368	0.9412	0.9412	0.9412	0.9412	0.9412
Media cuadrática	0.0539	0.0538	0.0520	0.0569	0.3970	0.9031	0.9412	0.9412	0.9412	0.9412	0.9412
Pesos 1	0.0539	0.0530	<b>0.0301</b>	0.3576	0.8609	0.9400	0.9412	0.9412	0.9412	0.9412	0.9412
Pesos 2	0.0539	0.0445	0.0905	0.5629	0.8740	0.9403	0.9412	0.9412	0.9412	0.9412	0.9412
Pesos 3	0.0539	0.0534	0.0401	0.5587	0.8629	0.9408	0.9412	0.9412	0.9412	0.9412	0.9412
Pesos 4	0.0539	0.0529	0.1975	0.7383	0.8938	0.9410	0.9412	0.9412	0.9412	0.9412	0.9412
Pesos 5	0.0539	0.0539	0.0529	<b>0.0475</b>	0.1955	0.6340	0.9383	0.9412	0.9412	0.9412	0.9412
Pesos 6	0.0539	0.0539	0.0533	0.0514	0.0852	0.3106	0.6985	0.9372	0.9412	0.9412	0.9412
Ratio de Fisher	0.0539	0.0492	0.0666	0.5575	0.8991	0.9412	0.9412	0.9412	0.9412	0.9412	0.9412

Cuadro 5.16: Tabla con la tasa de fallos de todas las ecuaciones de combinación con el rango de umbral 0-1 en pasos de 0.10 unidades

# Capítulo 6

## Conclusión

### 6.1. Conclusiones

En el tiempo empleado para realizar el proyecto, se ha conseguido en primer lugar adquirir conocimientos sobre las redes complejas. Aprendiendo de forma general los diversos tipos en los que se clasifican y cuáles son sus principales características y aplicaciones en el mundo real.

En cuanto a la metodología empleada para el desarrollo del proyecto aunque ha llevado más tiempo su realización, también ha permitido definir mejor las líneas generales de desarrollo y una mejor revisión de cada una de las piezas de las que se compone, gracias a las sucesivas reuniones con los tutores. Esta metodología también ha proporcionado la posibilidad de evaluar diversos caminos o alternativas al objetivo principal del proyecto, lo que ha permitido profundizar y comprender mejor las dificultades que surgen al llevar a cabo labores de clasificación.

También se ha intentado comprender de forma muy general, otros de los muchos de los clasificadores que hay disponibles, entre ellos, las redes neuronales artificiales, de las cuales como ya se comentó en la sección correspondiente se intentó realizar un acercamiento a ellas, mediante diversos experimentos, los cuales al final no tuvieron los resultados esperados, siendo esta una más de las vías que se ha explorado a lo largo del desarrollo del proyecto.



Otro aspecto en el que se ha ganado experiencia es en el análisis de los datos obtenidos, utilizando para ello la diversidad de herramientas que Matlab pone a disposición de sus usuarios. Del cual, tras un estudio más detallado y contrastando con diversos resultados, se han obtenido unas conclusiones que no se esperaban durante las primeras etapas del estudio.

Como ya se ha visto, durante el análisis inicial de las 3 características que se extraen de la red y después de evaluar sus respectivas matrices, la que proporciona un resultado más acertado es la característica de *circularidad*, en cuya representación visual se podía apreciar como en la diagonal se agrupan los bloques correspondientes a las imágenes de cada clase con un valor de distancia menor frente al resto de características.

Mientras que en los casos de *grado* y *joint degree*, representan los opuestos donde de forma general para el primero la distancia general en toda la matriz es muy baja y para el caso de *joint degree* el valor de distancia es de media más alto, aunque ésta última tiene un mayor grado de discriminación entre clases que para la propiedad de *grado*.

El siguiente paso en el análisis se da cuando se han fusionado todas las propiedades en un único bloque de datos, en el que todavía no se modifica la topología de las redes obtenidas, el estudio concluye que los casos que consiguen realizar una mejor clasificación son para las combinaciones de media, media cuadrática y la ecuación que aplica un 0.60 del peso total a la característica de *circularidad*, la cual era un resultado esperado, ya que previamente se había establecido que la *circularidad*, era la propiedad que de forma aislada aportaba mejores resultados. También se confirmaba en este apartado que el uso combinado de las tres características supone una mejora positiva frente a la utilización por separado de cada una de ellas, siempre y cuando el peso aplicado a cada característica no sea demasiado elevado, es decir, que no se aplique un peso superior a un 0.60 del porcentaje total a estas propiedades, excepto en el caso de la distribución en la que se aplica un 0.80 a la característica *joint degree*.

En la etapa final en la que se ha analizado la dinámica de las redes con un valor de umbral decreciente. Se ha llegado a la conclusión de que con un valor de umbral en el que empiezan a eliminarse conexiones, el caso que aporta la mejor clasificación es para la combinación basada en los máximos

donde consigue agrupar correctamente las imágenes de 14 clases de las 17 que se han utilizado en el estudio. En segundo lugar como mejor clasificador se encuentra la combinación mediante la distribución de pesos que aplica un 0.80 a la propiedad de *joint degree*. Estos mejores resultados se deben a que los datos generados por estas combinaciones tienen unos valores de distancia en general más elevados frente al resto, y aunque las distancias sean mayores, consiguen tener una diferencia mayor entre figuras de la misma clase que frente a figuras de otro tipo, por lo que permite agrupar de forma aceptable a la mayoría de clases utilizando un valor de umbral menos restrictivo lo que favorece a la clasificación debido a que se evita descartar imágenes de una clase con una distancia superior a la media.

Aunque se ha demostrado que tomar alguna de las 3 propiedades aisladas o en el caso de utilizar las ecuaciones de distribuciones de pesos de estas 3 características, no son buenos modelos para utilizar como clasificadores. Este hecho tiene como excepción a la ecuación de pesos a la que otorga el 0.80 total a la propiedad *joint degree*, en la que consigue agrupar correctamente 11 de las 17 clases. Este resultados se produce por los valores generados por esta propiedad lo que provoca el caso de una mayor separación en las distancias entre clases distintas.

Las otras combinaciones que sintetizan las 3 propiedades y que se pueden utilizar como clasificadores, son la media y la media cuadrática las cuales aunque no consiguen agrupar tantas clases perfectamente como los 2 casos anteriores, tienen una tasa global de fallos baja, pero para conseguir los mejores resultados, evitando falsos positivos, es necesario que se aplique un escalado con un valor de umbral inferior al de los casos anteriores, conllevando ciertos riesgos de descartar imágenes pertenecientes a una clase.

Por tanto el objetivo propuesto al inicio se ha cumplido. Se ha aportado un nuevo acercamiento al campo de la clasificación de imágenes, utilizando para ello las redes complejas y algunas de sus propiedades estadísticas de los cuales algunos de los resultados consiguen realizar un buen agrupamiento. Se deja como base una herramienta abierta a multitud de opciones para poder realizar clasificaciones basadas en otros aspectos de las redes complejas.

## 6.2. Líneas futuras de trabajo

El modelo de clasificación desarrollado es modular e independiente de las características utilizadas para realizar la labor de clasificación de imágenes. Estas propiedades son posibles, gracias al enfoque que se ha aplicado mediante las Redes Complejas. Como se ha podido observar en los apartados anteriores, gracias a este enfoque aplicado al proyecto se han conseguido unos resultados de clasificación con un porcentaje de error aceptable.

El modelo desarrollado en este proyecto puede ser ampliado de múltiples maneras, de las cuales, algunas de ellas se presentan a continuación:

- Características: Las Redes Complejas poseen otras propiedades aparte de las explicadas en este proyecto, por lo que sería interesante estudiar otras características y comprobar cómo afecta al rendimiento y la eficacia de los nuevos resultados obtenidos frente a los ya expuestos.
- Estudiar nuevas combinaciones: Existe una multitud de combinaciones disponibles para sintetizar el valor de la distancia entre dos imágenes que no han sido exploradas para este proyecto. También teniendo en cuenta el anterior punto, aparte de analizar nuevas funciones de combinación para las características existentes, también es una buena vía para ampliar este proyecto estudiar como afectan a los resultados de clasificación, la combinación de las características ya estudiadas frente a las nuevas que se añadan, teniendo una fuente de estudio bastante amplia con el gran número de combinaciones posibles.
- Trabajar con imágenes en escala de grises: Una evolución natural del modelo puede consistir en adaptarlo para que pueda trabajar con imágenes en escala de grises e incluso con imágenes en color, aunque se tendría que controlar el factor del crecimiento del volumen de datos, ya que por ejemplo, una imagen RGB consta de 3 matrices cada una asociada al nivel de intensidad de uno de los tres colores base correspondiente a los píxeles de la imagen.
- Ampliar el modelo de clasificación a otros tipos de datos: Un primer cambio en los datos que la aplicación puede manejar es el comentado en el punto anterior referente a imágenes en escala de grises y en color. Pero el modelo puede ampliarse para que pueda trabajar con bases de datos

que contengan otro tipo de información. Esta información puede estar recogida en ficheros de vídeo, e incluso puede ampliarse para trabajar con ficheros de audio para clasificar estos datos en diversas categorías según la información que contengan. Todo ello es posible, incluso si se continúa el proyecto utilizando Matlab porque proporciona todas las herramientas necesarias para trabajar con estos tipos de datos.

- Análisis de posibles herramientas utilizando Redes Complejas: Las Redes Complejas se encuentran en un punto en el que se pueden explotar de múltiples formas y en muchos campos, no sólo respecto a los asociados a Visión Artificial, como puede ser el reconocimiento facial [4] o el análisis del contorno de los objetos [5], también pueden aplicarse las características y fundamentos de construcción estudiadas en este proyecto a estudios de otras ramas de la ingeniería.

Analizando todos los puntos citados, se llega a la conclusión de que el proyecto desarrollado no es ni mucho menos cerrado, sino que se puede considerar como el punto de partida para una gran variedad de proyectos, incluso en disciplinas que no están directamente relacionadas con el campo en el que se ha orientado este estudio. También se puede tomar el camino de ampliar el modelo con otra serie de características para conseguir un modelo más completo o eficiente.

# Bibliografía

- [1] W. Abd-Almageed, C. Smith, “Hidden Markov Models for Silhouette Classification”, Automation Congress, 2002 Proceedings of the 5th Biannual World. 13:395-402(2002).
- [2] R. Albert, A.L. Barabasi, “Emergence of scaling in random networks ”, Science 286 (5439), 509-512(1999).
- [3] D.R. Amancio, L. da Fontoura Costa , M.G.V. Nunes, O.N. Oliveira, T.A.S. Pardo, “On the concepts of Complex Networks to evaluate machine translation”, Physica A-statistical Mechanics and Its Applications, vol 390, n ° 1, 131-142(2011).
- [4] J. de Andrade, O. Martinez, W. Nunes, “A Rotation Invariant Face Recognition Method Based on Complex Network ”, CIARP 2010 (64119), 426-433(2010).
- [5] A. R. Backes, D. Casanova, O. Martinez, “A complex network-based approach for boundary shape analysis ”, Pattern Recognition 42, 54-67(2009).
- [6] B. M. de Brío, “Redes Neuronales y Sistemas Borrosos ”, Ra-Ma (2006).
- [7] P. Erdos, A. Renyi, “On random graphs ”, I. Publ. Math. Decebren 6 290-297(1959).
- [8] L.M. Escudero, “Epithelial organisation revealed by a network of cellular contacts”, Nature communications, 8 de nov 2011.
- [9] L. da Fontoura Costa, M.G.V. Nunes, O.N. Oliveira, T.A.S Pardo, “Using Complex Networks for Language Processing: The Case of Summary Evaluation”, International Conference of Communications, Circuits and Systems-ICCCAS(206).

- [10] S. Gold, A. Rangarajan, “A graduated assignment algorithm for graph matching”, PAMI, 18(4), 377-388(1996).
- [11] Página web de la base de datos utilizada en el proyecto,  
  
`www.lems.brown.edu/vision/researchAreas/SIID`  
  
, último acceso: 28 de noviembre de 2012
- [12] Página web de MathWorks : <http://www.mathworks.com/>, último acceso: 28 de noviembre de 2012
- [13] D. Sharvit, J. Chan, H. Tek, B.B. Kimia, “Symmetry-based Indexing of Image Databases”, LEMS, Division of Engineering Brown University, Providence RI 02912
- [14] S.H. Strogatz, D.J. Watts, “ Collective dynamics of small-world networks ”, Nature 393(6684), 440-442(1998)

# Apéndices

# Figuras de la BBDD

En el presente apéndice se muestran todas las figuras de cada clase de la BBDD [11] que se ha utilizado durante el desarrollo de todo el proyecto.



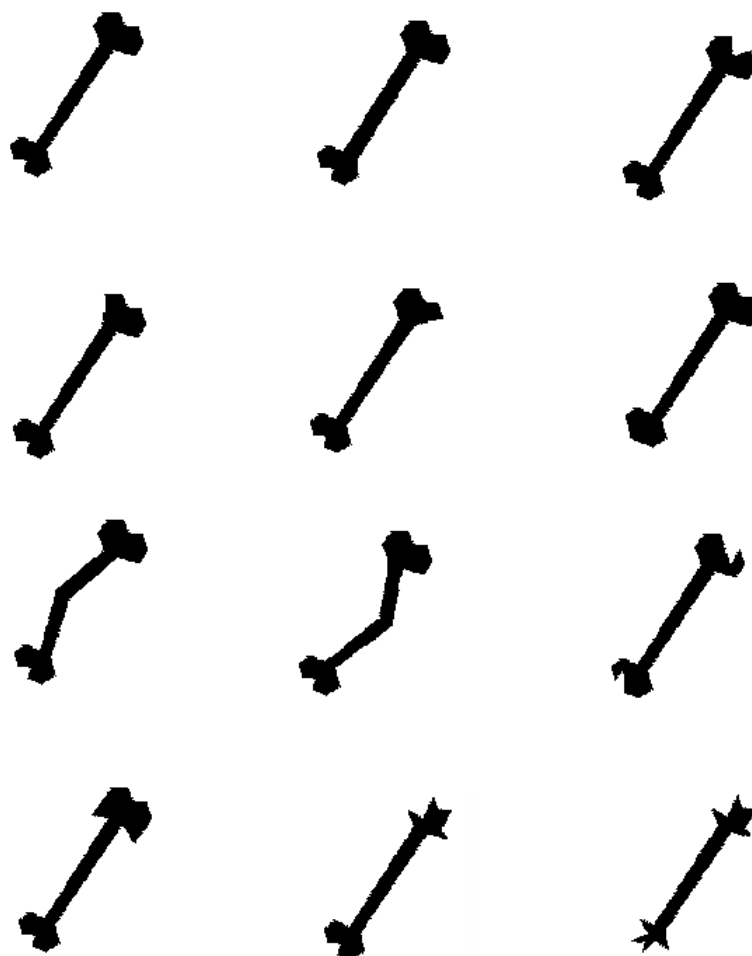


Figura 1: Imágenes de la clase 1 etiquetada como "Bone"

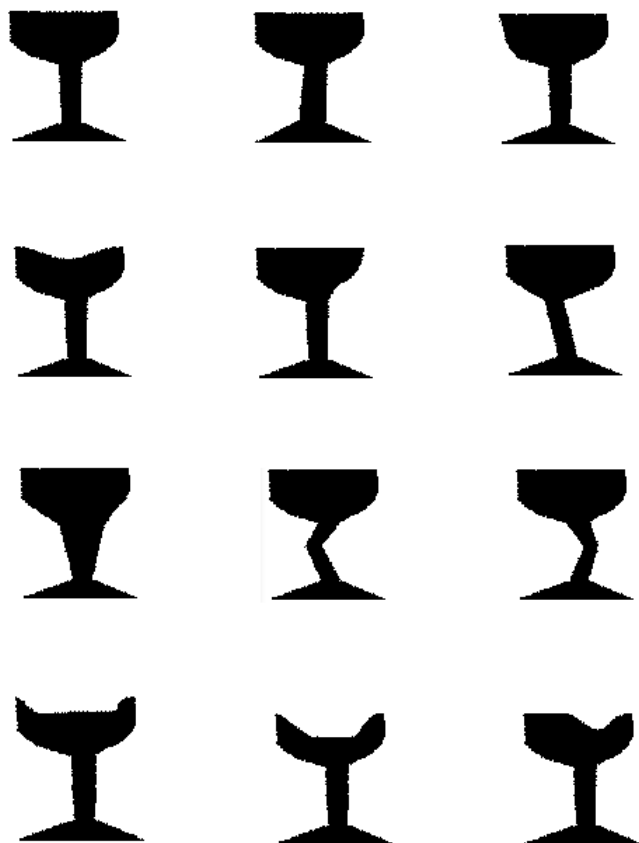


Figura 2: Imágenes de la clase 2 etiquetada como "Glass"

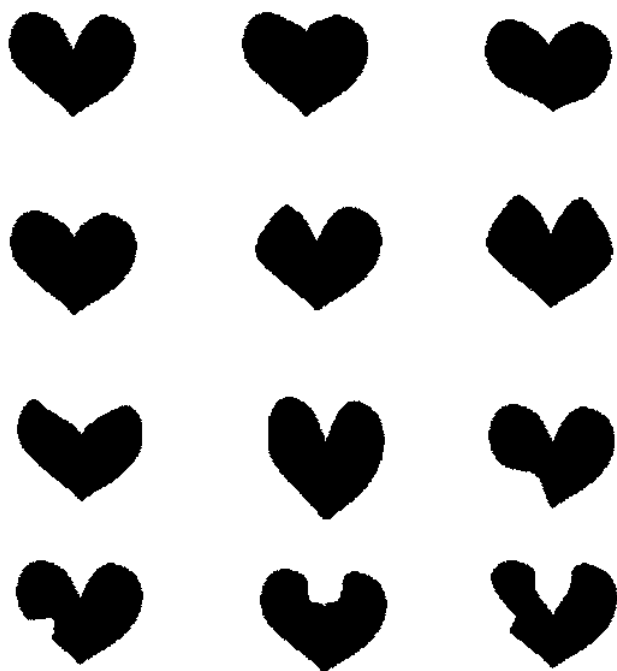


Figura 3: Imágenes de la clase 3 etiquetada como "Heart"

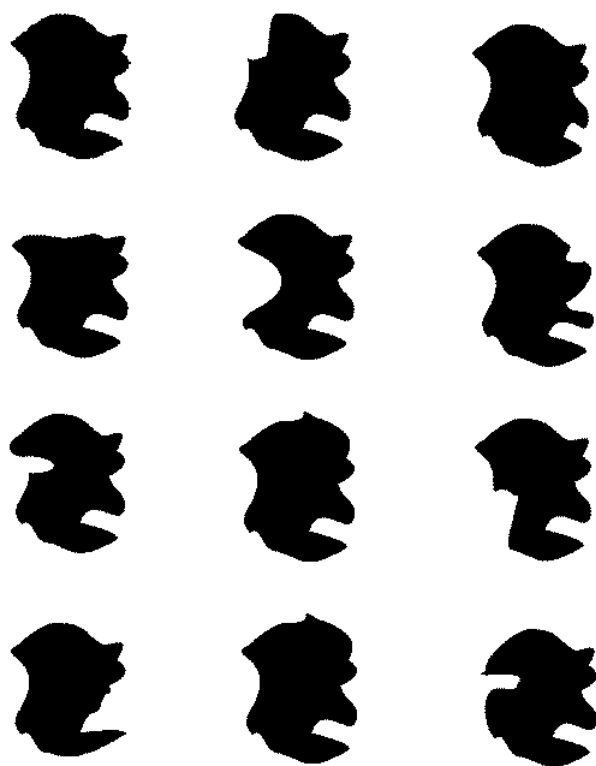


Figura 4: Imágenes de la clase 4 etiquetada como "Misk"

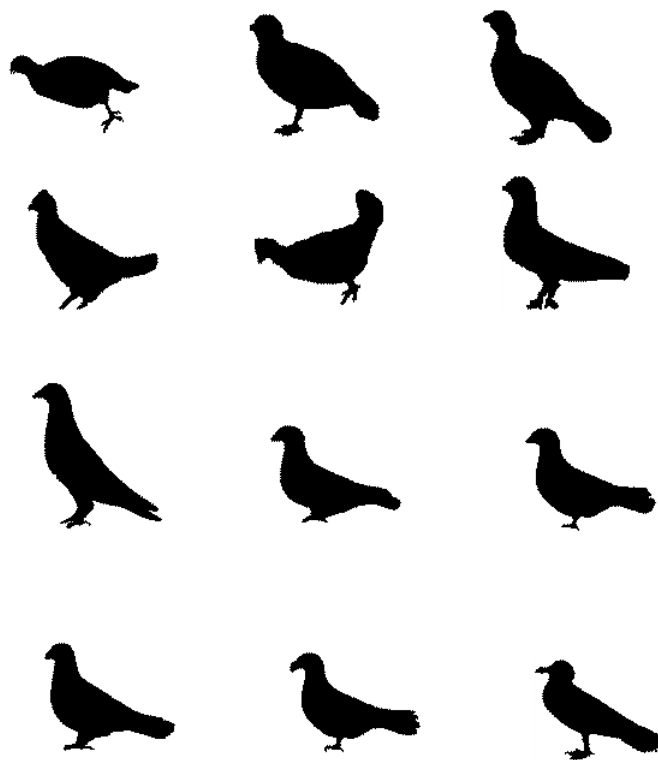


Figura 5: Imágenes de la clase 5 etiquetada como "Bird"

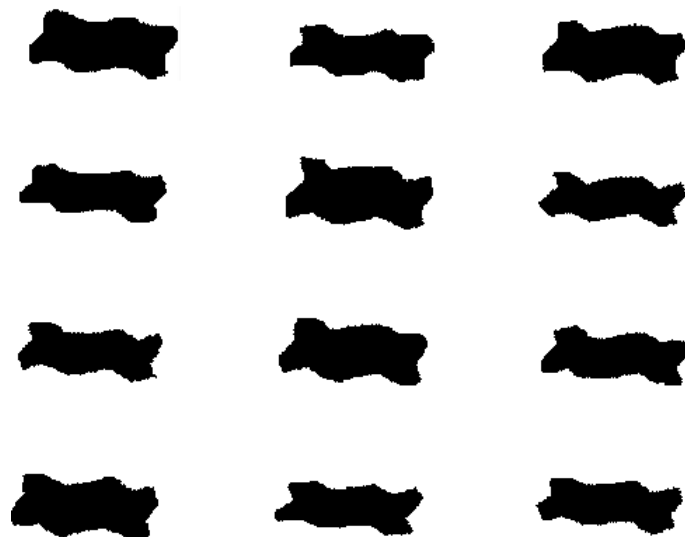


Figura 6: Imágenes de la clase 6 etiquetada como "Brick"

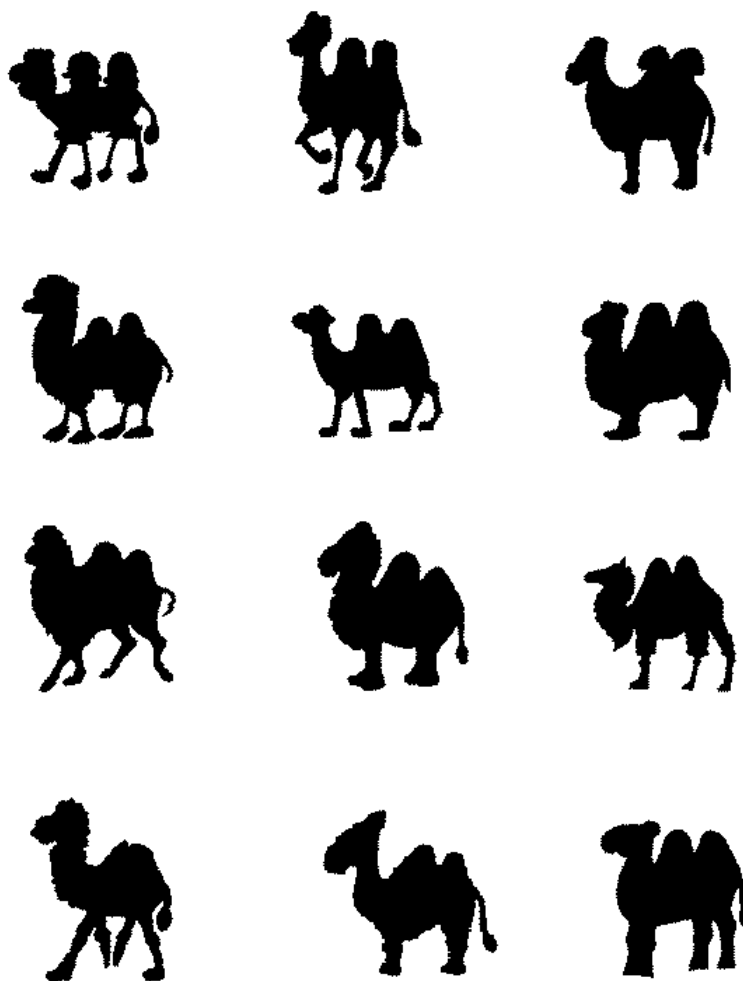


Figura 7: Imágenes de la clase 7 etiquetada como "Camel"

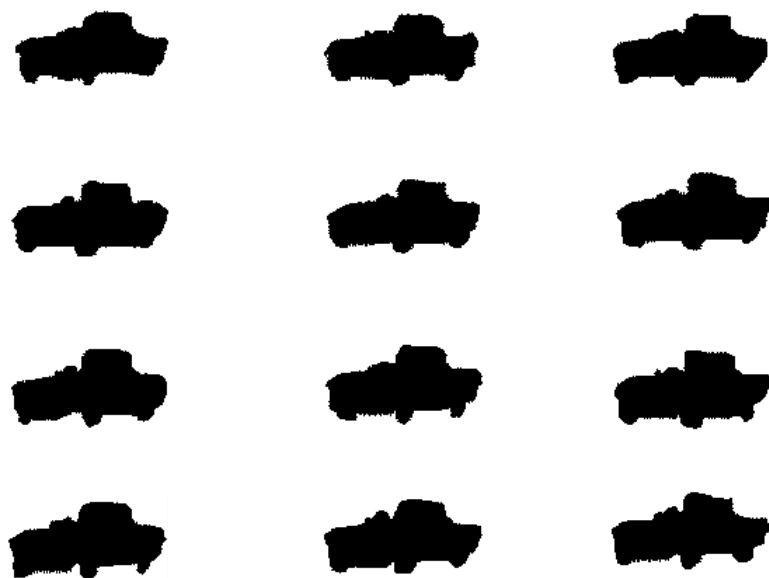


Figura 8: Imágenes de la clase 8 etiquetada como "Car"



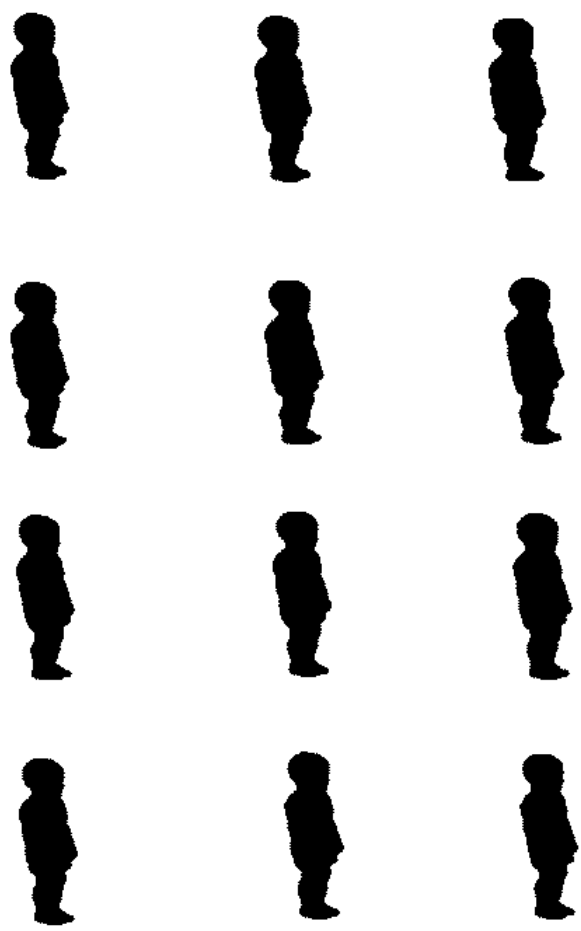


Figura 9: Imágenes de la clase 9 etiquetada como "Child"

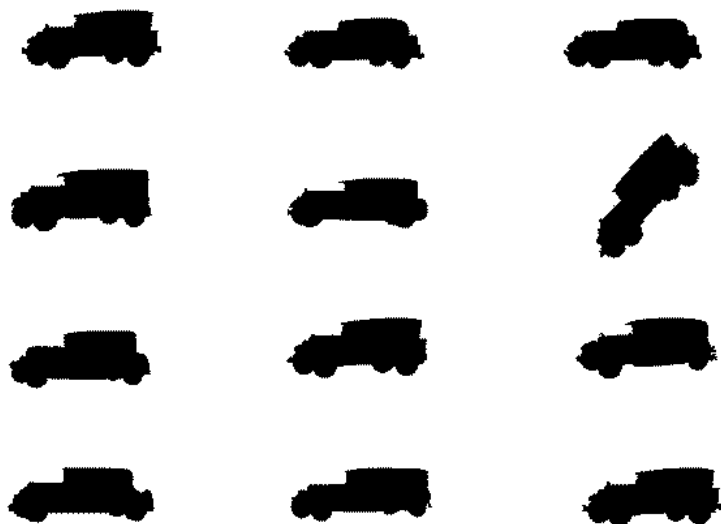


Figura 10: Imágenes de la clase 10 etiquetada como "Classic"

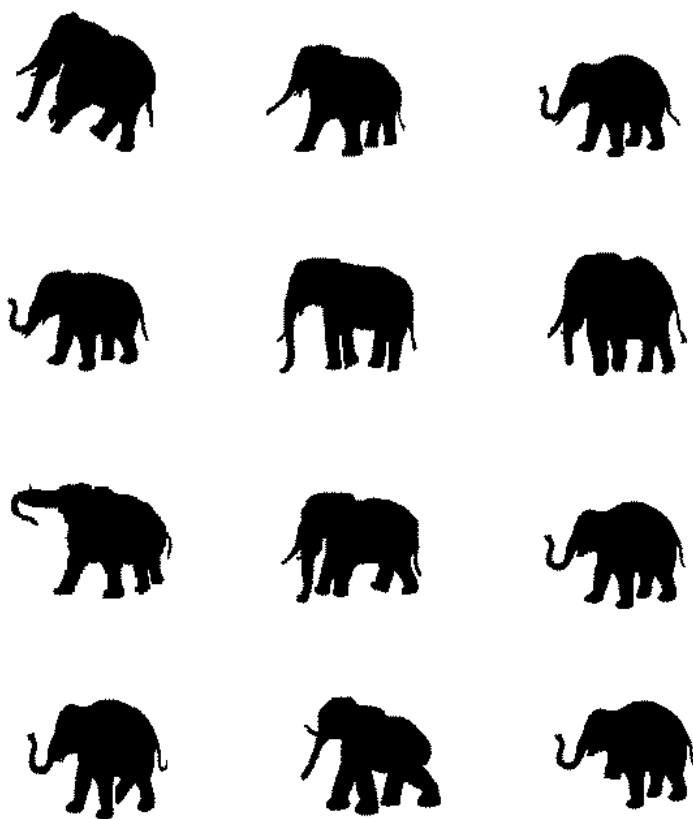


Figura 11: Imágenes de la clase 11 etiquetada como "Elephant"

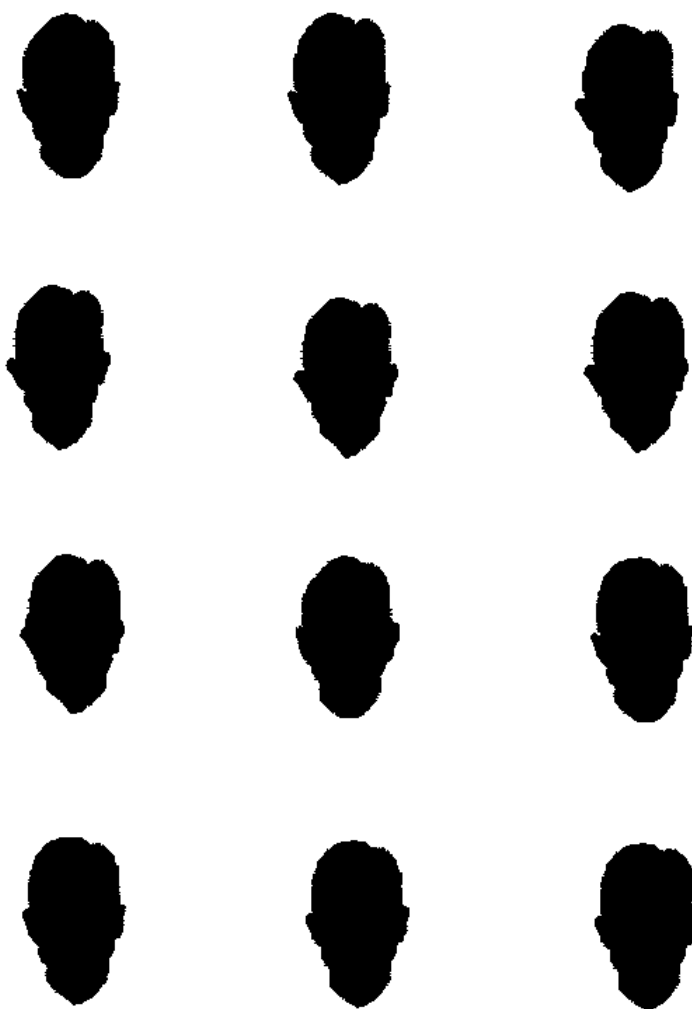


Figura 12: Imágenes de la clase 12 etiquetada como "Face"

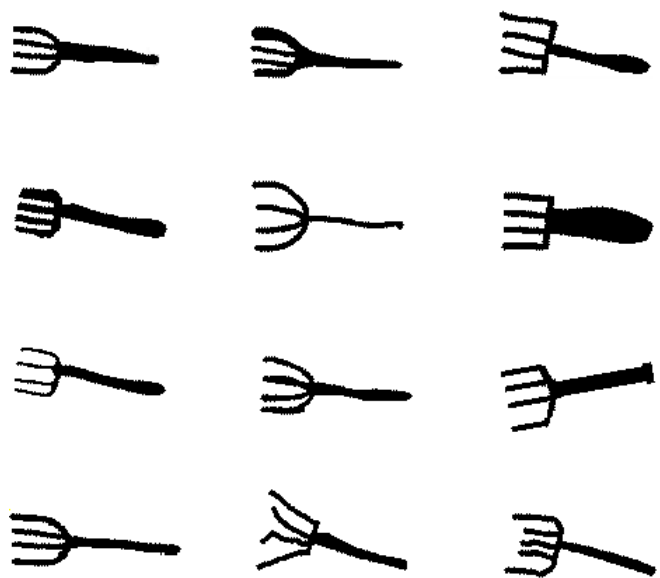


Figura 13: Imágenes de la clase 13 etiquetada como "Fork"

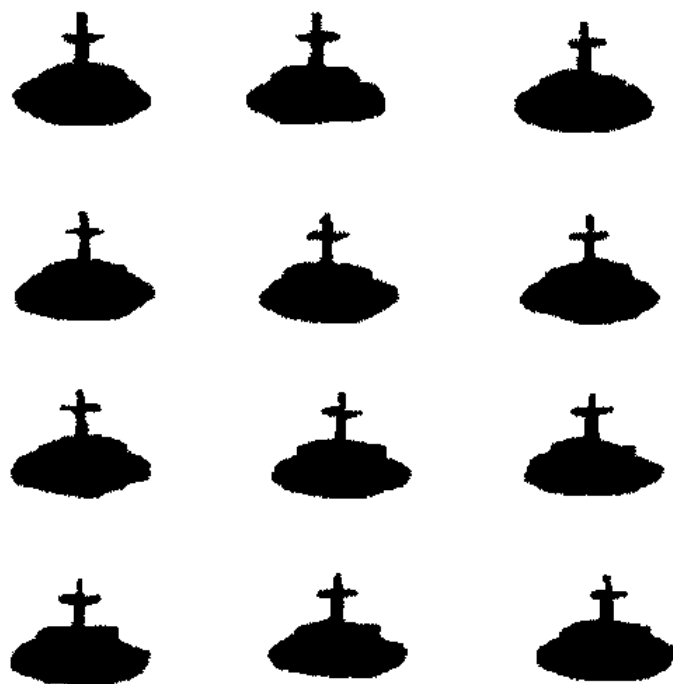


Figura 14: Imágenes de la clase 14 etiquetada como "Fountain"

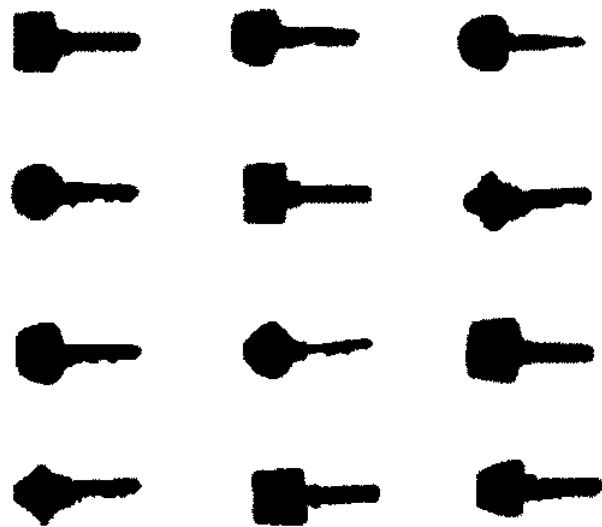


Figura 15: Imágenes de la clase 15 etiquetada como "Key"

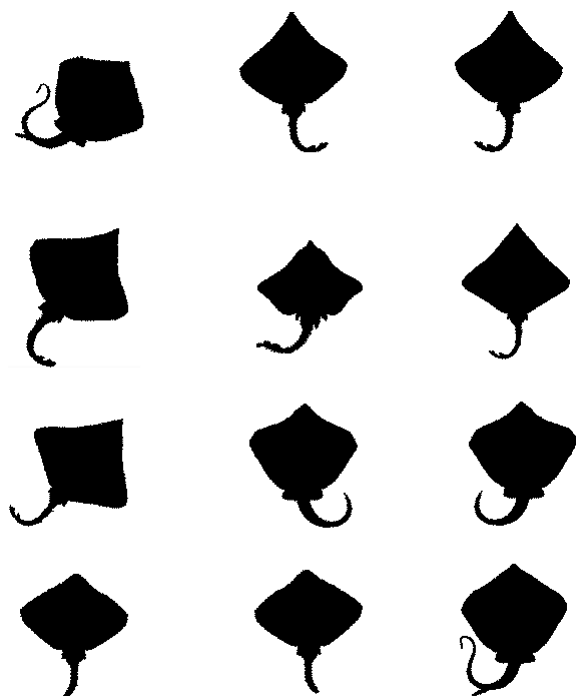


Figura 16: Imágenes de la clase 16 etiquetada como "Ray"



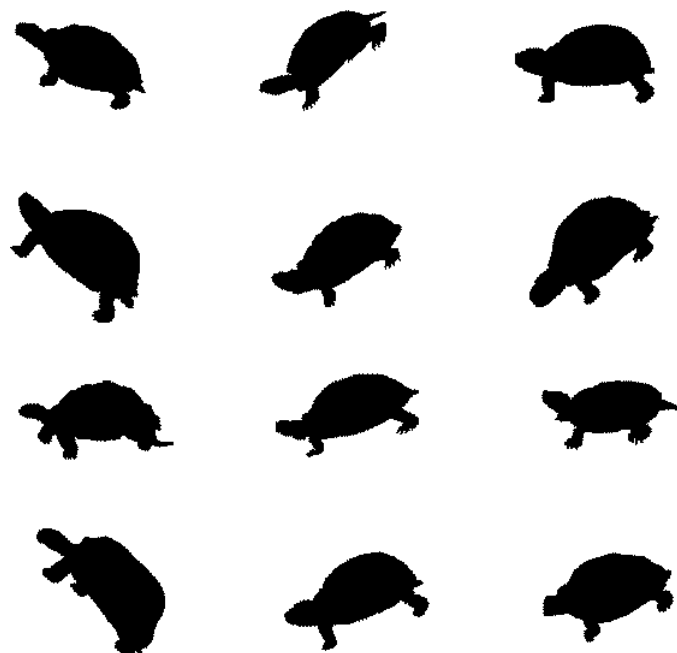


Figura 17: Imágenes de la clase 17 etiquetada como "Turtle"