

*UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR*



***ANÁLISIS DE IMÁGENES FACIALES
BASADO EN PROPIEDADES
ESPECTRALES DE GRAFOS***

***PROYECTO FIN DE CARRERA**
INGENIERÍA SUPERIOR DE TELECOMUNICACIÓN*

AUTOR: JULIO DANIEL PÉREZ ORR

TUTOR: CARLOS BOUSOÑO CALZÓN

JUNIO 2009

Título: ANÁLISIS DE IMÁGENES FACIALES BASADO EN EL ESPECTRO DE GRAFOS

Autor: Julio Daniel Pérez Orr

Tutor: Carlos Bousoño Calzón

EL TRIBUNAL

Presidente: Fernando Díaz de María, Teoría de la Señal y Comunicaciones

Secretario: Harold Molina Bulla, Teoría de la Señal y Comunicaciones

Vocal: Isaac Seoane Pujol, Ingeniería Telemática

Realizado el acto de defensa del Proyecto Fin de Carrera el día 18 Junio 2009 en Leganés, en la Escuela Politécnica Superior de la Universidad Carlos III de Madrid, acuerda otorgarle la CALIFICACIÓN de:

Fdo. Presidente

Fdo. Secretario

Fdo. Vocal

Agradecimientos

Quisiera agradecer a todas las personas que, de un modo u otro, han formado parte de este proyecto.

En primer lugar, agradecer a mi tutor, Carlos Bousoño, la ayuda y confianza para que se pudiera llevar a cabo este proyecto desde la distancia. A su vez, quiero mencionar y agradecer la ayuda de Juan Santaella, quien me ha echado mil manos cuando eran necesarias. Por otro lado, agradecer a Fernando Díaz de María, la cesión de la base de datos con imágenes faciales, fundamental para el desarrollo del proyecto.

Gracias a mi familia, por su apoyo constante desde el principio de la carrera hasta el final. A mis padres y a mi hermana, gracias por entenderme y haberme soportado en los buenos y malos momentos.

Mención especial para Miriam, quién siempre ha estado a mi lado para sacar lo mejor de mí y ha sido un apoyo imprescindible para sacar el proyecto adelante. El ánimo, apoyo y cariño siempre han estado presentes y son de agradecer.

No quisiera olvidarme de los componentes de los Luigis, equipo formado en la universidad y que se ha transformado en un grupo de amigos inigualables y para toda la vida.

Por último, a todos aquellos que habéis formado parte de mi carrera y de mi vida en estos años, muchas gracias!



Descripción

La proliferación de diferentes fuentes de información de texto, imágenes y video en formato digital hace interesante la investigación de tecnologías de procesado de datos que puedan aplicarse independientemente del tipo de información permitiendo, eventualmente, un procesado integrado de múltiples fuentes de una manera que cupiera denominar *semántica*; por su nivel de abstracción y su potencial acercamiento a la forma en que el ser humano maneja la información. Este proyecto fin de carrera hace un análisis preliminar del empleo del análisis espectral de grafos definidos a partir de imágenes con el propósito anterior. Los resultados de esta investigación muestran que los espectros de los grafos definidos son mayormente degenerados, imposibilitando su uso para la categorización de imágenes, y que es necesaria una reorientación en la definición de los mismos.

ÍNDICE

Agradecimientos.....	4
Descripción.....	6
ÍNDICE.....	8
ÍNDICE FIGURAS	10
ÍNDICE TABLAS	12
1. INTRODUCCIÓN.....	13
2. ESTADO DEL ARTE	15
2.1 Estado del Arte en Minería de Textos	15
2.1.1 Algoritmos.....	16
2.1.2 Aplicaciones	17
2.2 Semántica	21
2.2.1 Aplicaciones	23
2.3 Estado del Arte en Sistemas de Reconocimiento Facial.....	27
2.4 Estado del Arte : Combinación de ambos campos	39
2.5 Extracción de Características.....	43
2.6 Descriptores Locales	47
3. HERRAMIENTAS DE TRABAJO	49
3.1 Espectro de Grafos	49
3.2 Base de Datos	52
4. DESARROLLO.....	53
4.1 Las Imágenes.....	53
4.1.1 Corpus de trabajo.....	55
4.2. Tratamiento de la imagen	56
4.2.1 Obtención de la imagen formato ppm a jpg	56
4.2.2 Preprocesado.....	57
4.2.3 Extracción de Características	59
4.3 Tratamiento del Grafo	63
4.3.1 Definición de Palabras.....	63
4.3.2 Construcción del Grafo.....	65
4.3.2.1) CASO 1: Construcción directa del grafo a partir de los puntos.....	65
4.3.2.2) CASO 2: <i>Palabras visuales</i> para toda la imagen.....	68
4.3.2.3) CASO 3: Segmentación de la imagen en Celdas	69
4.3.3 Espectro de un Grafo	70
4.3.4 Comparativa de espectros.....	73
5. RESULTADOS	74
CASO 1: Construcción directa del grafo a partir de los puntos característicos	74
Prueba I. Diferencias entre características.....	74
Prueba II. Comparación entre Grupos (G1 vs. G2).....	75
Prueba III. Diferencias entre géneros	76
Prueba IV. Mismas comparativas con grafos aleatorios.....	77
Prueba V. Variaciones del umbral.....	78
Prueba VI. Combinaciones de los algoritmos de extracción de características.....	78
Prueba VII. Comparativas individuales en lugar de por grupos.....	78
CASO 2: Construcción del grafo usando <i>palabras visuales</i> para toda la imagen..	79
Prueba VIII. Comprobación definición de <i>palabras</i>	79

Prueba IX. <i>Palabras visuales</i>	80
Prueba X. Espectro del Grafo con Palabras Visuales.....	80
Prueba XI. Variaciones en el Espectro	81
CASO 3: Construcción del grafo usando <i>palabras visuales</i> con la imagen segmentada en celdas.....	82
Prueba XII. Segmentación de la imagen	82
Prueba XIII. Espectro del Grafo con Segmentación de la Imagen en celdas.....	83
6. CONCLUSIONES.....	85
7. LÍNEAS FUTURAS.....	87
8. BIBLIOGRAFÍA.....	88
ANEXO	93

ÍNDICE FIGURAS

<i>Figura 1: Ejemplo extracción de hechos.....</i>	17
<i>Figura 2: El gráfico de la derecha muestra las posibles estructuras de una red semántica, mientras que el diagrama de bloques de la izquierda muestra las etapas de extracción y clasificación del conocimiento.</i>	22
<i>Figura 3: Ilustración de la sociedad del Conocimiento [18]</i>	23
<i>Figura 4: Ejemplo de aplicación académica [17]</i>	24
<i>Figura 5: Interfaz Gráfica de la aplicación [17].....</i>	25
<i>Figura 6: Esquema de un ejemplo de ontología médica. [19].....</i>	25
<i>Figura 7: Diagrama de bloques [19].....</i>	26
<i>Figura 8: Diagrama de clases de técnicas [25]</i>	29
<i>Figura 9. Ejemplo de Rostros proyectados sobre distintos subespacios extraídos con Matlab de un conjunto de rostros de FERET.</i>	30
<i>Figura 10. Ejemplo de coordenadas del conjunto de rostros sobre los 2 autovectores más relevantes. Cada asterisco representa la coordenada de un rostro.</i>	31
<i>Figura 11. Ejemplo de los resultados de LDA donde vemos mayor separación entre los distintos clusters, diferentes individuos, y mínima separación dentro de la misma clase o cluster, distintas poses de un individuo. [30]</i>	32
<i>Figura 12. Un sistema de reconocimiento facial con subespacio LDA. [33]</i>	32
<i>Figura 13: Esquema de funcionamiento de EBGM [34].....</i>	34
<i>Figura 14: Representación de la malla para reconocimiento [34].....</i>	34
<i>Figura 15: Plantilla 3D de un rostro [36]</i>	35
<i>Figura 16: Diagrama de Flujo [36].....</i>	36
<i>Figura 17: Ejemplos de formas canónicas de rostros[37].....</i>	38
<i>Figura 18: Tabla con las aplicaciones típicas de sistemas de reconocimiento. [33].....</i>	39
<i>Figura 19. Etapas del Sistema.</i>	41
<i>Figura 20: Ejemplos de resultados donde las elipses rojas equivalen a topicos de primer plano mientras que los puntos verdes son el tópico “fondo”. Vemos los resultados para las iteraciones 1,6, 12 y 26.</i>	41
<i>Figura 21: Esquema de trabajo propuesto. [48]</i>	42
<i>Figura 22: Ejemplo de colección de imágenes y el hSO extraído.....</i>	42
<i>Figura 23: Ejemplo de la extracción de candidatos y posterior reducción de los puntos[42].....</i>	44
<i>Figura 24: Ejemplo de la extracción de la orientación en los puntos clave.[42].....</i>	44
<i>Figura 25: Conjunto de bases wavelets[43]</i>	45
<i>Figura 26: Representación del espectro de un grafo incorrelado [50]....</i>	50
<i>Figura 27: Representación del espectro aumentando la probabilidad (0, 0.01, 0.3, 1)</i>	51
<i>Figura 28: Espectro de un grafo de Orden Libre</i>	51

Figura 29: Distintas vistas de un mismo individuo. Enmarcadas en rojo aquellas vistas frontales con las que se ha desarrollado el proyecto... 55

Figura 30: Gráficas con los porcentajes de las poblaciones divididas en grupos y por sexo..... 56

Figura 31: Resultados de la primera segmentación, a la izquierda la imagen con información relevante y a la derecha parte de la imagen que se considera el fondo con información irrelevante. 57

Figura 32: Imágenes resultantes tras la segunda segmentación donde se separan las características del rostro de la piel..... 58

Figura 33: Puntos de interés obtenidos usando el algoritmo de Harris para la imagen preprocesada (drcha.) y con la imagen completa (izq.) 60

Figura 34: Puntos característicos usando el algoritmo LoG para la imagen preprocesada y para la imagen completa. 61

Figura 35: Puntos de interés obtenidos usando el algoritmo de Gilles para la imagen preprocesada (drcha.) y con la imagen completa (izq.)..... 62

Figura 36: Selección final de los puntos característicos utilizados combinando los algoritmos de Harris y LoG. Comparativa en el uso de la imagen completa (dcha.) y la imagen preprocesada (izq.)..... 63

Figura 37: Imágenes representando las agrupaciones de nodos de una misma característica, bigote y orejas prominentes. 66

Figura 38: Distribuciones de grafos de orden libre y grafos aleatorios. (A.Barabasi [50])..... 66

Figura 39: Distribución de enlaces entre nodos para el grafo usado que cumple la forma de exponencial inversa, por tanto es un grafo de orden libre..... 67

Figura 40: Matriz de Adyacencia de un grafo genérico. 67

Figura 41: Ejemplo gráfico de la combinación de grafos de diferentes individuos para formar el grafo total por clase..... 69

Figura 42: Ejemplo gráfico del proceso empleado en la construcción del grafo al segmentar una imagen en celdas. 70

Figura 43: Ejemplos de distintos casos analizados de los espectros. 72

Figura 44: Ejemplo del espectro del tercer tipo de caso analizado. 73

Figura 45: Representación de la evolución del número de palabras nuevas a medida que se procesan imágenes. Se ha probado para dos umbrales de distancia entre nodos que forman una esquina..... 79

Figura 46: Espectro del grafo usando palabras visuales como nodos para la categoría Mujeres..... 81

Figura 47: Espectro obtenido disminuyendo la distancia entre puntos a 10 píxeles y por tanto reduciendo el número de nodos en el grafo..... 82

Figura 48: Distribución de los enlaces por nodos para la nueva definición del grafo, segmentando la imagen en celdas. 83

Figura 49: Espectro del grafo obtenido con la imagen segmentada en celdas para la categoría Hombres..... 84

ÍNDICE TABLAS

<i>Tabla 1: Número de individuos por cada grupo.....</i>	<i>56</i>
<i>Tabla 2: Número de individuos por característica y por grupo.....</i>	<i>56</i>
<i>Tabla 3: Resultados de la comparativa entre características en un mismo grupo usando etapa de preprocesado. (En rojo los valores mínimos que presentan la mayor similitud entre ellas). Nota: Resultados expresados en % de disimilitud/diferencia entre espectros.....</i>	<i>74</i>
<i>Tabla 4: Resultados de la comparativa entre características en un mismo grupo para imágenes completas. (En rojo los valores mínimos que presentan la mayor similitud entre ellas).</i>	<i>75</i>
<i>Tabla 5: Resultados de la comparativa entre características entre grupos distintos usando preprocesado. (En rojo los valores mínimos que presentan la mayor similitud entre ellas, en verde los campos que debían ser mínimos).Nota: Resultados expresados en % de disimilitud/diferencia entre espectros.....</i>	<i>76</i>
<i>Tabla 6: Resultados de la comparativa entre características entre grupos distintos usando imagen completa. (En rojo los valores mínimos que presentan la mayor similitud entre ellas, en verde los campos que debían ser mínimos).</i>	<i>76</i>
<i>Tabla 7: Resultados comparativa entre géneros de distintos grupos. (En verde valores mínimos que representan mayor similitud). Nota: Resultados expresados en % de disimilitud/diferencia entre espectros.</i>	<i>77</i>
<i>Tabla 8: Resultados comparativa entre géneros de distintos grupos. (En rojo valores mínimos que representan mayor similitud, en verde los campos que debían ser mínimos).....</i>	<i>77</i>

1. INTRODUCCIÓN

La proliferación de fuentes de información de texto, imágenes y video en formato digital provenientes de la *Web*, redes sociales en Internet, cámaras digitales personales o cámaras de seguridad -por citar algunos ejemplos- hace interesante la investigación de tecnologías de procesamiento de datos que puedan aplicarse independientemente del tipo de información (texto, audio, imagen o video) permitiendo, eventualmente, un procesamiento integrado de múltiples fuentes de una manera que cupiera denominar *semántica*; por su nivel de abstracción y su potencial acercamiento a la forma en que el ser humano maneja la información.

El desarrollo de este proyecto tiene como referencia de partida otro proyecto fin de carrera (PFC) en el que se aplicaba el análisis espectral de grafos al procesamiento de textos abstrayéndose del idioma y permitiendo identificar características de alto nivel, como estilos de escritura (véase [1]). A la vista de los prometedores resultados obtenidos en el análisis de textos, la motivación de esta nueva investigación no es otra que la de hacer una validación preliminar y una discusión de la pertinencia de la aplicación de estos métodos al análisis de imágenes.

Para abordar el estudio, se ha elegido una base de datos de imágenes faciales para extraer conceptos que permitan clasificar, distinguir o incluso reconocer rostros humanos. Aunque en la actualidad, existen un gran número de sistemas de Reconocimiento Facial que utilizan como herramienta las cualidades biométricas de las personas, este proyecto pretende extraer cualidades ocultas en imágenes las cuales son difícilmente reconocibles por los sistemas actuales. Se trata de cualidades sutiles o percepciones que se pueden obtener con una descripción textual, como por ejemplo, poder distinguir si una persona es guapa o fea.

Por otra parte, y dado que las herramientas de la Minería de Datos permiten la extracción de información útil, es decir, de características invariantes de distintos tipos de datos (por ejemplo, textos, imágenes o audio), la idea de este proyecto es combinar el Reconocimiento Facial y Minería de Datos con el fin de crear un proceso de análisis facial basado en las características extraídas invariantes de la imagen de un rostro. Para ello se emplearán, como herramientas básicas, algoritmos de tratamiento de imágenes para la extracción de características y herramientas para el análisis espectral de grafos.

El proyecto analiza, en primer lugar, el estado del arte de las áreas a investigar para poder apoyarnos, en el desarrollo del mismo, en las últimas líneas de investigación. El objetivo es demostrar que la idea de este PFC trata uno de los temas que, a día de hoy, se encuentra en pleno desarrollo e intensa actividad de investigación.

La contribución específica de este PFC se desarrolla en dos etapas bien diferenciadas. La primera de estas etapas consiste en la extracción de características de una imagen, en nuestro caso, de una toma frontal de un rostro, con el fin de conseguir un grafo único. Se definen *palabras visuales* para poder almacenar el contenido de la imagen y poder relacionarlas con el objetivo de formar un grafo. Una vez obtenido dicho grafo, en una segunda etapa, se analizará el espectro del mismo y, en función de

las propiedades o características encontradas, se evalúa la posibilidad de distinguir el género del individuo, o la posesión de alguna otra característica facial.

Para concluir, se discute los resultados obtenidos y se plantean alternativas de desarrollo o prueba futuros.

2. ESTADO DEL ARTE

Para comenzar, se van a introducir los dos principales campos de acción para situar tanto el punto de partida de ambos, así como el lugar en el que se encuentran hoy en día a nivel de investigación y desarrollo.

2.1 Estado del Arte en Minería de Textos

La sociedad usa el lenguaje natural para comunicarse entre sí, para organizar información para el futuro, para archivar información pasada por lo que se estima que el 80% de la información útil está guardada en formato de texto. Por ello, se han desarrollado desde principios de los 80 avances tecnológicos en el campo de la minería de textos para poder utilizar esta información en formato textual. Anteriormente, ya existía la minería de datos, para procesar datos estructurados de forma automática, pero la diferencia con el texto es justamente esa, que los textos carecen de esa estructura.

Emails, páginas web, memorias, transcripciones de centros de llamadas, encuestas, notas, casos legales, descripciones de patentes, artículos de investigación, incidencias, etc, todas estas fuentes aportan extractos de conocimiento donde se pueden descubrir patrones, tendencias y anomalías que pueden ayudar a predecir situaciones futuras. El problema es que el análisis manual, es decir humano, es ineficiente a la hora de procesar volúmenes grandes de texto además de ser lento, y vulnerable a errores. Por otro lado, el análisis estadístico no encuentra relaciones semánticas ocultas por lo que se utilizan las técnicas de minería de textos que engloban un poco de cada campo de análisis.

Como definición se usa:

La minería de textos es un área multidisciplinaria basada en la recuperación de información, minería de datos, aprendizaje automático, estadísticas y la lingüística computacional. [21]

La minería de textos (text mining) dentro del acceso, recuperación y organización de información es un conjunto de técnicas que permiten extraer información relevante y desconocida de forma automática dentro de grandes volúmenes de información textual, normalmente en lenguaje natural y no necesariamente estructurada. [22]

A su vez, el ser humano al leer crea conexiones con hechos que no parecen estar relacionados creando nuevas ideas o hipótesis, pero dado el enorme crecimiento de información, sería imposible estar al tanto de todos los avances en todos los campos, por lo que la minería de textos es una solución para reemplazar o mejor, suplementar al lector humano usando sistemas automáticos. De esta forma se descubriría información previamente desconocida de grandes volúmenes de texto.

2.1.1 Algoritmos

La minería de textos encapsula varias técnicas procedentes de distintos campos como son Information Retrieval (IR)(Recuperación de Información), Natural Language Processing (NLP)(Procesamiento Natural de la Lengua), Information Extraction (IE-Extracción de Información) y Minería de Datos. Todas estas etapas se pueden agrupar en un único proceso formando lo que sería la forma de trabajo de la minería de textos. A continuación se explica la función da cada uno de los subprocesos:

Los sistemas de **Information Retrieval (IR)** identifican los documentos en una colección que coinciden con la petición del usuario. Con estos sistemas reducimos el conjunto de documentos que son relevantes y por tanto aceleramos el análisis general.

Natural Language Processing (NLP) es uno de los problemas más complicados en el campo de la inteligencia artificial y consiste en analizar el lenguaje humano para que los ordenadores puedan comprenderlo de la misma forma que los humanos. Este campo sigue en vías de desarrollo quedando mucho para que se cumpla este objetivo, aunque si que existen ciertos tipos de análisis que funcionan correctamente. El rol de NLP en la minería de textos es de dotar a los sistemas que extraen la información de datos lingüísticos como fronteras de frases, etiquetas de “part-of-speech” para discernir nombres, verbos y adjetivos, etc.

El proceso de **Information Extraction (IE)** es aquel en que se obtiene de forma automática información estructurada a partir de documentos no estructurados de lenguaje natural. Esta información estructurada vendría a denotar un concepto, la unidad semántica básica que permite expresar una idea. A menudo, este proceso implica definir las estructuras de la información que se busca para que sirvan de guía en los procesos de extracción. Además, estos sistemas tienen una fuerte dependencia con la información generada por los sistemas de NLP.

Algunas de las tareas que puede llevar a cabo un sistema IE son:

- Análisis de términos, que identifica términos en un documento formados por una o varias palabras.
- Reconocimiento de Nombres y Entidades, tarea que identifica nombres, tanto de personas como organismos, en un documento así como fechas, expresiones temporales, cantidades y asociar unidades y porcentajes.
- Extracción de hechos: Es el proceso que identifica y extrae hechos complejos de un documento. Estos hechos pueden ser relaciones entre entidades o eventos. Estos procesos se pueden ver a continuación en un breve ejemplo (figura 6), en el cual se ve una estructura y cómo se rellena con una frase. El sistema IE es capaz de identificar un tipo de interacción, la palabra “*bind*”, con dos nombres de proteínas “*myosin*” y “*actin*”. Este tipo de información puede ser archivada en un diccionario o una ontología y posteriormente será exportada a una base de datos para el procesado de la última etapa, la minería de datos.

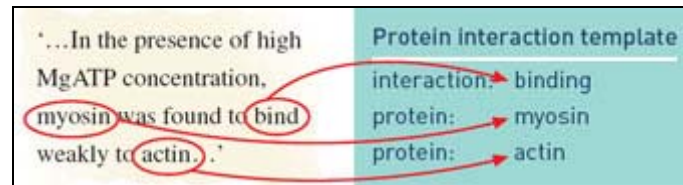


Figura 1: Ejemplo extracción de hechos.

La **Minería de Datos** (DM también conocida como Knowledge Discovery) es el proceso que identifica patrones en conjuntos grandes de información. El objetivo es descubrir conocimiento desconocido y útil. En minería de textos, el proceso DM se aplica a los hechos generados en la etapa de extracción de la información. Estos hechos se almacenan en una base de datos donde posteriormente se buscan patrones entre ellos. Por último, estos hallazgos se vuelven a almacenar preparados para peticiones del usuario en busca de información y con las respuestas a las peticiones se pueden presentar los resultados gráficamente como relaciones entre los hechos.

2.1.2 Aplicaciones

Una vez conocido el origen y evolución de la minería de textos así como su funcionamiento básico dividido por subprocesos, nos adentramos en las aplicaciones en las que se puede desarrollar esta técnica y las ramas futuras donde se está investigando.

EMPRESAS – BUSINESS INTELLIGENCE

- Algunas empresas utilizan ya sistemas de minería de textos para identificar el contenido de los e-mails que les envían sus clientes y redirigirlos a los departamentos apropiados. En otros casos, si el sistema es capaz de identificar el contenido de una consulta frecuente en un e-mail, envía una respuesta estándar a la consulta, sin necesidad de intervención humana. Otros de los servicios que existen en las empresas que usan la minería de textos son:
 - Análisis automático y mejorado de respuestas a sondeos.
 - Generación de informes mediante el análisis de transcripciones en centros de llamadas.
 - Determinación de las causas principales de problemas mediante análisis de informes de incidencias.
- Donde quizá lleva más tiempo utilizándose esta tecnología es en la Vigilancia Tecnológica y en la "**Business Intelligence**" para bucear en las bases de datos textuales y seguir la evolución de la competencia.
- Se apunta también la posibilidad de usar esta técnica en la investigación de mercados en la Web, a base de recoger estadísticas sobre la utilización de determinados conceptos y/o temas en la red con el objetivo de estimar la demografía y las curvas de demanda de productos asociados a los mismos.

BIOMEDICINA - BIOLOGÍA CONCEPTUAL

Uno de los avances más significativos se ha dado con la definición de la biología conceptual como complementaria a la biología empírica. Esto es debido al gran número de recursos digitales disponibles como por ejemplo redes de bases de datos, libros digitales, etc. Con los descubrimientos de Swanson, y basándose en sistemas de descubrimiento de hipótesis, se ha creado la biología conceptual. Dicha biología conceptual ha sido defendida en un artículo de la revista Nature[24] como un complemento perfecto a la biología empírica debido al enorme incremento de datos digitales sobre todo en los campos de biología molecular y celular. Una vez aceptada en la comunidad biológica, se ha servido de las investigaciones de Swanson en minería de textos.

Ideas de Swanson

La base de las ideas de Swanson viene resumida por la siguiente cita: “Conocimiento público sin descubrir”. Con ello se quiere expresar el conocimiento publicado pero que está “escondido” al estar en campos separados al que se está investigando. De esta forma, Swanson explicó la generación de conocimiento o hipótesis procedentes de publicaciones en principio sin relación entre ellas o como él denominaba, disjuntas. Esta metodología está implementada en un programa llamado ArrowSmith, donde podemos observar los resultados de dicho método. Este método incluye filtros para controlar la confusión y para introducir inteligencia humana al sistema así como criterios de parada basados en el número de términos o estadísticos como la frecuencia relativa o restricciones categóricas. Esta metodología se ha denominado “Swanson Linking (SL)”.

Existen muchos métodos diferentes de minería de datos donde se buscan las mismas citas en diferentes documentos como nombres de autor, nombres de revistas, palabras, frases, bases de datos de pacientes, secuencias de genes, de proteínas... Además se añaden aproximaciones estadísticas, lingüistas o visuales. Esta minería añade valor a la información recuperada y se encapsula en otro proceso más abstracto que se denomina Knowledge Discovery in Databases (KDD) que consiguen formas con mayor significativo. Ambas unidas han sido denominadas como Knowledge Discovery in text (KDT) que consiste en tres etapas:

- i. Recuperación de la información de los documentos relevantes
- ii. Extracción de información (entidades, relaciones, eventos...)
- iii. Minería de textos para hallar patrones “no triviales, implícitos y previamente desconocidos”.

A partir de las investigaciones de Swanson se han seguido desarrollando metodologías en este campo. Estas nuevas investigaciones han sido fieles a las ideas iniciales, pero han aportado diversos cambios como la sistemización de los primeros métodos, la mejora de la automatización del descubrimiento de hipótesis o la minería de entidades en vez de títulos.

WEB

- Un campo importante de aplicación es la Web. Existen diferentes tipos de minería de textos web, la de contenido, la de estructura y la de uso:
 - **Minería de Web de Contenido:** Dentro de la Web nos encontramos con gran cantidad de documentos heterogéneos, ya sean hipertexto, documentos de texto, documentos en formato pdf, así como imágenes o vídeos, lo que dificulta su clasificación. La minería de contenido del web trata de extraer información relevante sobre el contenido de la web de manera que pueda ayudar clasificarlo, aumentando la organización de ese contenido, para posteriormente mejorar el acceso y la recuperación de la información en él contenida.
 - **Minería Web de estructura:** Sirve para saber cómo está organizada una Web, cómo está estructurada y cómo es la navegación a través de ella.
 - **Minería de uso del Web:** Trata de extraer patrones por parte de los usuarios. Para ello utiliza los logs de los servidores Web de forma que aplicando minería de textos sobre ellos se pueda extraer información útil. Este tipo de minería tiene 2 objetivos principales: uno es sacar patrones generales de uso de un sitio web de manera que se pueda reestructurar para que sea más fácil de utilizar y mejore el acceso por parte de los usuarios. El otro uso es obtener perfiles de los distintos tipos de usuarios a través de su comportamiento y navegación, para poder atender de forma más personalizada.

Estos tres tipos de minerías son el punto de partida de la **Web semántica**, término que se utiliza para la World Wide Web una vez añadidos metadatos semánticos. Este campo será analizado posteriormente con más detalle una vez que profundicemos en la extracción de conocimiento o significado.

SOFTWARE COMERCIAL

Otro claro ejemplo que demuestra el desarrollo de estas técnicas es la gran variedad de programas o aplicaciones comerciales que existen en la actualidad:

- Anderson Analytics - proveedor de análisis de texto y de contenido relacionado con el comportamiento del consumidor.
- Attensity - grupo de soluciones de minería de textos para diversas industrias.
- Autonomy - software de minería de textos, clustering y categorización.

- Clarabridge - aplicaciones de minería de textos, categorización para clientes, servicios de salud y analítica investigativa.
 - Clearforest - software de minería de texto para extraer el significado de varias formas de información textual.
 - Cortex Intelligence - proveedor de análisis de contenido de Web.
 - Crossminder - empresa de minería de textos con búsqueda multilingüe y aproximación semántica.
 - IBM Intelligent Miner for Text - software de minería de textos comercial.
 - Inxight - proveedor de tecnologías de análisis de texto, búsqueda y visualización.
 - Island Data – Inteligencia de Mercado en tiempo real de retroalimentación no estructurada del cliente.
 - Nstein Technologies - proveedor de text analytics, y asset/web content management technologies (media, e-publishing, online publishing).
 - PolyAnalyst - software de minería de textos.
 - SAS Enterprise Miner - software de minería de textos.
 - SPSS - proveedor de TextSmart, SPSS Text Analysis for Surveys y Clementine, productos que se pueden utilizar con otros de SPSS.
 - TEMIS - TEMIS es un productor de software en el área de inteligencia de información.
 - TextAnalyst - software de minería comercial.
 - Textalyser - herramienta de análisis en línea para ver las estadísticas de textos.
 - Topicalizer - una herramienta en línea para generar estadísticas de páginas web y otros textos.
 - The "Ultimate Research Assistant" - una herramienta para gerencia de conocimiento que usa una combinación de búsqueda tradicional y técnicas de minería para facilitar la búsqueda en línea de temas complejos.
- Se le presta cada vez un mayor interés a la minería de textos multilingual: la habilidad de ganar información en otros idiomas.

APLICACIONES DE SEGURIDAD

Una de las aplicaciones más amplias de minería de textos es el sistema de vigilancia ECHELON [22,23]. Es un sistema inteligente para vigilar comunicaciones electrónicas usado por la NSA (National Security Agency) en Estados Unidos. Este sistema captura grandes volúmenes de tráfico provenientes de comunicaciones satélite, radiofrecuencia y microondas, celular y fibra óptica desde y hacia Estados Unidos. Toda esta información se filtra buscando ciertas palabras clave o frases.

LIMITACIONES

El principal factor limitante es la imposibilidad, a día de hoy, de una interpretación total de un texto así como la falta de un 20% de la información que no se encuentra almacenada en formato textual. Otra de las limitaciones que han creado

polémica es en aspecto moral del análisis de información en cuanto al riesgo de transgredir la propiedad privada en el caso de la aplicación Echelon.

2.2 Semántica

Según la Real Academia de la Lengua Española, la definición de semántica es: “Estudio del significado de los signos lingüísticos y de sus combinaciones, desde un punto de vista sincrónico o diacrónico”. Uno de los campos donde más se ha investigado es en la extracción de dicha semántica o conocimiento de datos en formatos textuales. Para ello, primeramente se deben utilizar las herramientas de minería de textos para extraer unas unidades de significado mínimas y luego poder clasificarlas. Este proceso se resume en la formación de ontologías las cuales forman una red semántica donde se alberga todo el contenido semántico de una determinada información analizada.

El concepto de ontología proviene de la filosofía definiéndose como “la ontología o *Metafísica general* es el estudio de lo que es en tanto que es y existe”. Por ello, es llamada *la teoría del ser*, es decir, el estudio de todo lo que es: qué es, cómo es y cómo es posible. La Ontología se ocupa de la definición del ser y de establecer las categorías fundamentales o modos generales de ser de las cosas a partir del estudio de sus propiedades. Por contra, el término ontología en informática hace referencia al intento de formular un exhaustivo y riguroso esquema conceptual dentro de un dominio dado, con la finalidad de facilitar la comunicación y poder compartir la información entre diferentes sistemas. Es decir, una ontología es una especificación formal y explícita de una conceptualización compartida en un dominio de interés [21] teniendo que estar especificada en un lenguaje que contiene semántica formal para poder ser interpretada por máquinas.

Estas ontologías se pueden especificar de varias formas según el campo de acción en el que se trabaje. El objetivo final es conseguir una ontología estructurada por capas con el fin de obtener un significado único. Este hecho es complicado con el lenguaje natural por eso se utilizan lenguajes formales que siguen la siguiente estructura:

$$\theta := \{C, R, H^c, rel, A^o\}$$

Donde θ es la ontología, C es el conjunto de conceptos, R son las relaciones entre dichos conceptos, H^c es el concepto de jerarquía, es decir el nivel o capa del concepto, rel es la función y A^o es el conjunto de axiomas.

Una vez definida la estructura principal de la ontología se puede organizar de diversas formas dependiendo de la estructura jerárquica. De estas formas componen grafos que pueden tener formas de árbol, formas totalmente arbitrarias o formas mixtas entre las dos como es el caso de ontologías estructuradas por capas donde en cada capa se relacionan de manera arbitraria teniendo posteriormente forma de jerarquía de árbol según el nivel. Podemos ver a continuación un ejemplo gráfico de estas situaciones:

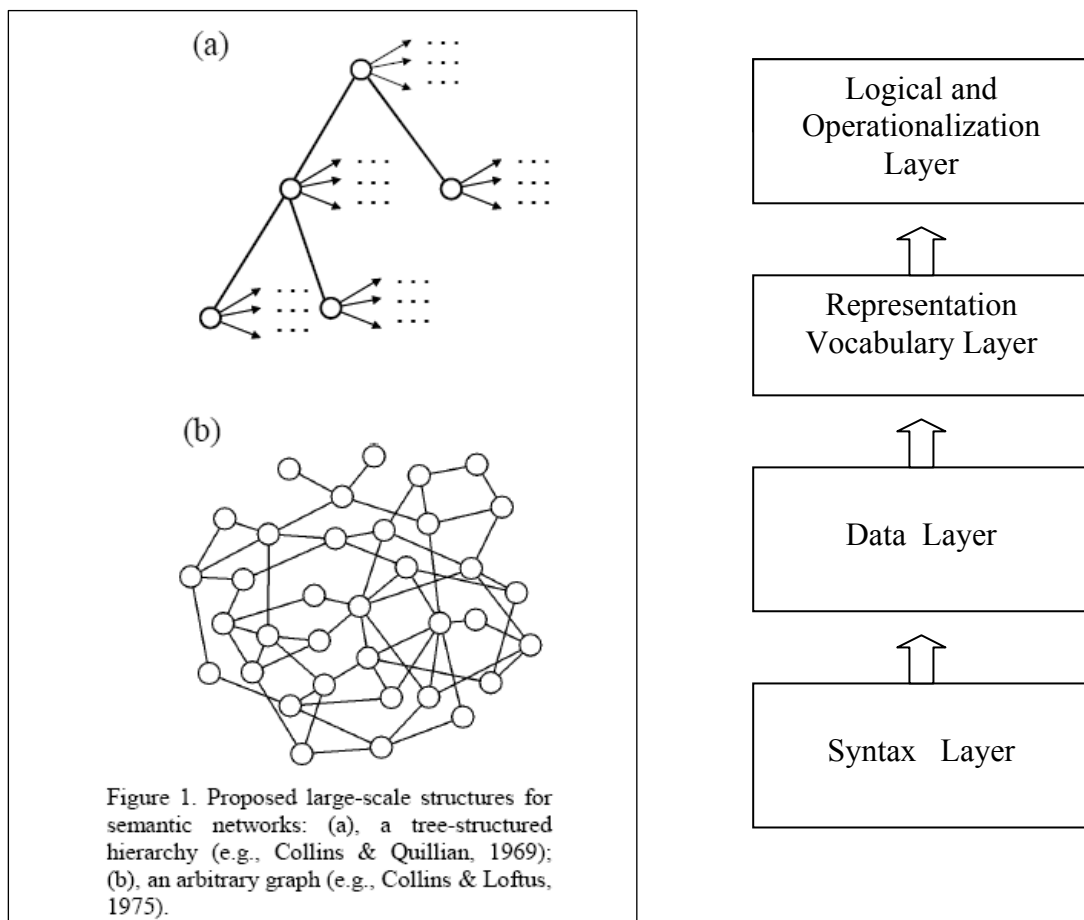


Figura 2: El gráfico de la derecha muestra las posibles estructuras de una red semántica, mientras que el diagrama de bloques de la izquierda muestra las etapas de extracción y clasificación del conocimiento.

Actualmente no existe un estándar concreto para las ontologías y de hecho se está produciendo una “guerra semántica y ontológica debido a la competencia entre estándares” para poder tener un dominio comercial en este campo.

2.2.1 Aplicaciones

Una vez comprendida la estructura de una ontología podemos adentrarnos en las aplicaciones donde se están utilizando hoy en día y podremos ver la amplia diversidad de aplicaciones donde se están desarrollando.

WEB SEMÁNTICA

Uno de los campos donde más se ha desarrollado es en el manejo de aplicaciones sobre el conocimiento como la Web semántica sobre todo en aplicaciones de extracción de información, minería de textos y respuesta de peticiones. Dado su amplia utilización y lo costoso y tedioso que consiste desarrollar una ontología, hoy en día el interés existe en el aprendizaje automático de ontologías basado en texto. El fin buscado son estructuras conceptuales más expresivas y concisas y en ello trabajan actualmente diversas comunidades. El problema es que cada comunidad parece trabajar de forma independiente cuando el fin es el mismo para todos, el aprendizaje de las estructuras conceptuales, pero unas comunidades buscan “concept hierachies” (jerarquías de concepto), otras “heterarchies”, otras “conceptual patterns” (Patrones conceptuales) o por último ontologías. En función de lo visto, la tendencia actual es que cada comunidad trabaja por separado pudiendo alcanzar mejores resultados trabajando conjuntamente por lo que existen talleres donde se intenta fusionar las distintas ramificaciones hacia la misma dirección. Esta misma dirección se podría llamar la sociedad del conocimiento y como podemos ver en el siguiente gráfico se puede descomponer en distintos sectores que lo forman y hacia donde están dirigidas las áreas de aplicaciones.

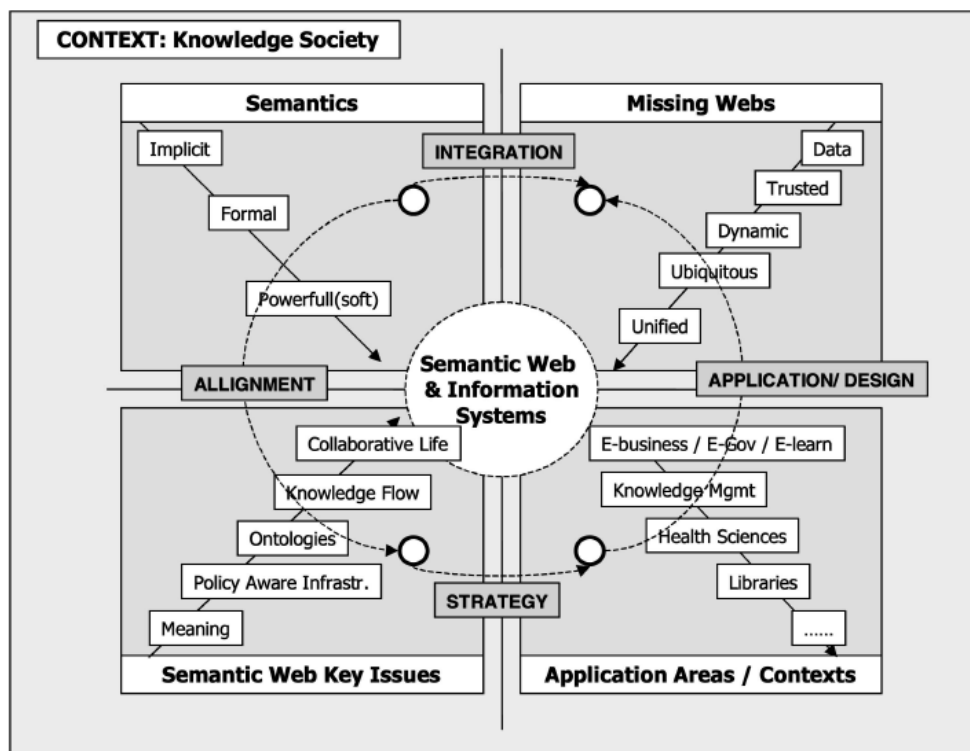


Figura 3: Ilustración de la sociedad del Conocimiento [18]

Ejemplo Aplicación: Área Académica

Una de las aplicaciones que utiliza la Web Semántica es un curso online. De esta forma los profesores obtienen retroalimentación de parte del uso de los alumnos para posteriores mejoras del curso y los alumnos obtienen diferentes elementos relacionados para el aprendizaje.

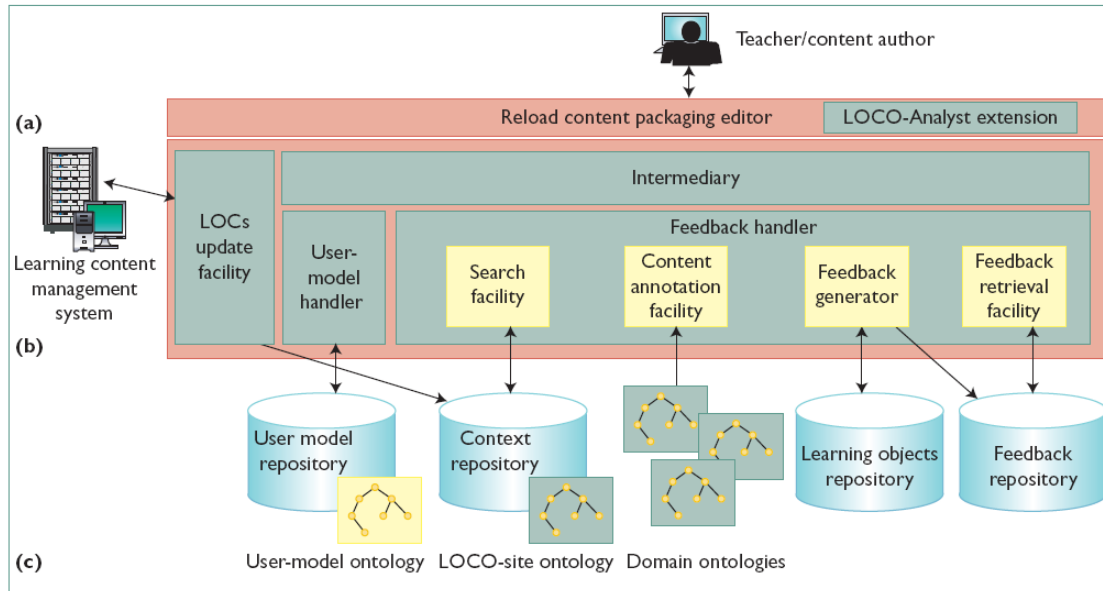


Figure 1. The LOCO-Analyst's three-tier architecture. (a) The user interface is built on the Reload editor with our extension for learning feedback. (b) The application logics is the core part of the tool responsible for semantic data integration and analysis. (c) The semantic repositories include different kinds of learning-related data stored in ontology-based data repositories.

Figura 4: Ejemplo de aplicación académica [17]

En la figura podemos observar la estructura de la aplicación y como depende de una arquitectura de tres niveles que tiene en la base las diferentes ontologías en las que se basa. A su vez podemos observar el programa en sí y cómo interactúa el estudiante con él así como se extraen de forma automática los estadísticos del uso para su posterior análisis.

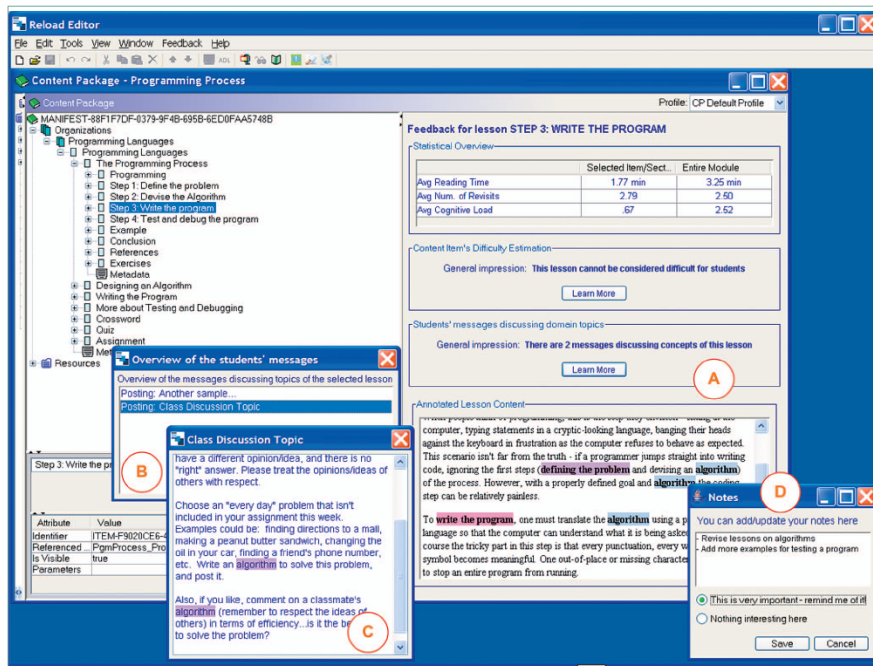


Figure 2. Feedback for the lesson "Step 3: Writing the program." The left half of the screen comes from the Reload editor and shows the lesson plan and some general content packaging data. The right half shows the generated feedback.

Figura 5: Interfaz Gráfica de la aplicación [17]

Ejemplo Aplicación: Medicina

En el caso de la medicina [19], también vemos el uso de ontologías para estructurar la información. Una ontología médica es un modelo de conocimiento que contiene conceptos relevantes relacionados con la diagnosis, el tratamiento, los procedimientos clínicos y los datos del paciente. Todo esto se agrupa para obtener razonamientos y decisiones basadas en el conocimiento. Un ejemplo de cómo quedaría formada la ontología la podemos apreciar a continuación donde vemos como las clases o conceptos poseen atributos y están relacionadas con otras clases.

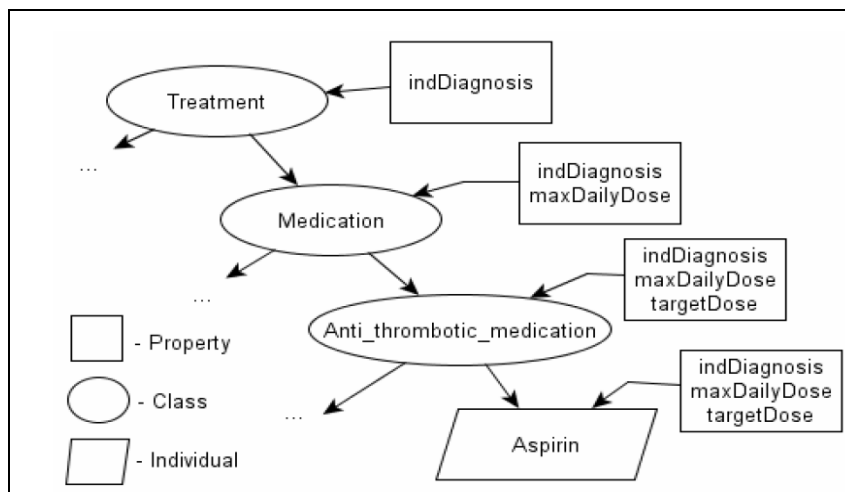


Figura 6: Esquema de un ejemplo de ontología médica.[19]

A su vez debemos tener en cuenta que la ontología en sí tan sólo es una base de conocimiento y que para razonar con ella se debe implementar un sistema de soporte para decisiones. Para ello se presenta el siguiente gráfico donde podemos observar cómo funciona todo el sistema. Inicialmente, ocurre un evento que se le pasa a una interfaz la cual tiene asociada un sistema de control que extrae los datos del paciente de la base de datos. Estos datos son transformados al formato de la ontología y preparados para razonar con ellos como un conjunto de hechos. A continuación se pasa al proceso de razonamiento que sigue un conjunto de normas, y las conclusiones se devuelven nuevamente a la ontología, la cual posee un intérprete que las devuelve al usuario a través de otra interfaz.

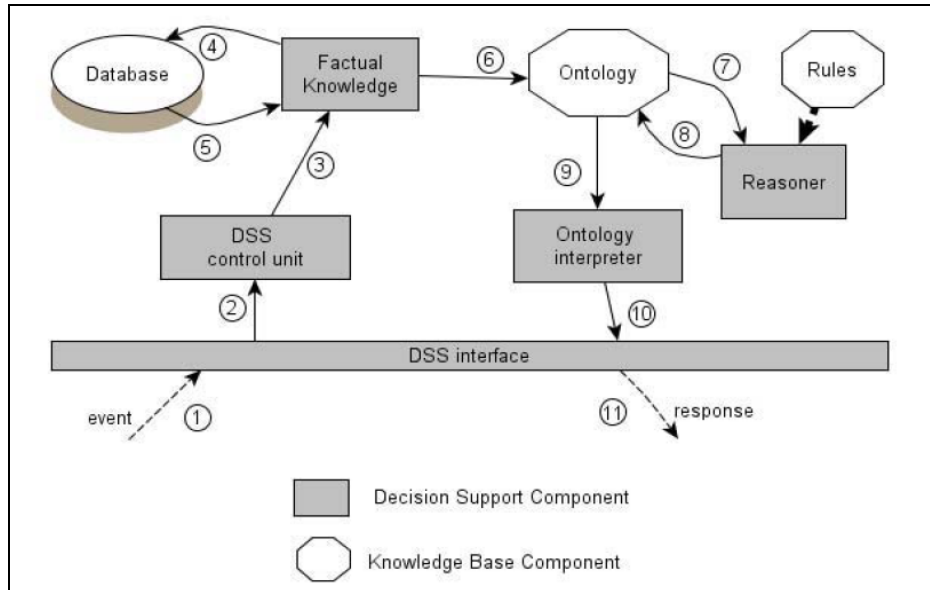


Figura 7: Diagrama de bloques [19]

La ventaja de usar una ontología para representar el conocimiento se encuentra en la estandarización de términos médicos, tener conocimientos compartidos y el soporte de un razonamiento automático.

Ejemplo Aplicación: Militar

Otro campo donde se están usando y desarrollando las ontologías son en áreas militares [21]. En este ejemplo, se trata de obtener una infraestructura del conocimiento para situaciones de guerra. Esto consiste en transformar toda la información disponible en conocimiento y para ello es necesario construir ontologías militares de dominio, ontologías de situación y ontologías de normas. De esta forma se consigue automatizar las normativas a seguir en determinadas situaciones bélicas.

2.3 Estado del Arte en Sistemas de Reconocimiento Facial

2.3.1 Historia

El campo de tratamiento de imágenes para la extracción de semántica es muy extenso por lo que nos centraremos en nuestro proyecto que es el reconocimiento facial. El reconocimiento facial automatizado es un concepto relativamente nuevo que se empezó a desarrollar en los años 60. El primer sistema era semiautomático puesto que requería de un administrador para localizar rasgos (como ojos, orejas, nariz y boca) en las fotografías antes de que este calculara distancias a puntos de referencia en común, los cuales eran comparados luego con datos de referencia. En los años 70 Goldstein, Harmon, & Lesk [2], usaron 21 marcadores subjetivos específicos tales como el color del cabello y grosor de labios para automatizar el reconocimiento facial. El problema con estas soluciones previas era que se computaban manualmente. En 1988 Kirby & Sirobich aplicaron análisis de componentes principales, una técnica estándar del álgebra lineal, al problema del reconocimiento facial. Esto fue considerado algo así como un hito al mostrar que eran requeridos menos de 100 valores para cifrar acertadamente la imagen de una cara convenientemente alineada y normalizada [3]. En 1991 Turk & Pentland utilizando las técnicas Eigenfaces, el error residual podía ser utilizado para detectar caras en las imágenes [4] – un descubrimiento que permitió sistemas automatizados de reconocimiento facial en tiempo real fiables.

La tecnología inicialmente capturó la atención del público a partir de la reacción de los medios en una prueba de implementación en la Super Bowl de la NFL en enero de 2001, la cual capturó imágenes de vigilancia y las comparó con una base de datos de foto archivos digitales. Hoy la tecnología de reconocimiento facial está siendo utilizada para combatir el fraude de pasaportes, soporte al orden público, identificación de niños extraviados y minimizar el fraude en las identificaciones en otras muchas aplicaciones.

2.3.2 Algoritmos y Técnicas

A lo largo de la historia del reconocimiento facial se han utilizado muchas y diversas técnicas para tratar las imágenes, pero el punto de partida es la imagen en sí. En los tiempos actuales, se trabaja y se investiga con imágenes digitales, de distintos formatos y características, pero nosotros no queremos adentrarnos en el mundo de la imagen digital sino usarla como “materia base” de trabajo y beneficiarnos de esas cualidades y propiedades para tratar con ellas.

Para empezar, debemos separar dos de las ramas con las que se está trabajando en la actualidad en el reconocimiento de imágenes referido a la forma de la imagen, es decir, 2-Dimensiones o 3-Dimensiones. Hoy en día, y gracias a los avances tecnológicos, se pueden obtener imágenes en 3D que a simple vista, al contener más información, se suponen mejores, aunque también tienen sus desventajas, por ello se ha

elaborado un cuadro comparativo para ver las ventajas y desventajas de cada uno de los formatos:

2-Dimensiones

Ventajas:

- Disponibilidad de grandes colecciones de imágenes 2D
- Dispositivos de captura más económicos.

Desventajas:

- No se puede manejar variaciones en las poses.
- Sensible a variaciones de iluminación, sombras, etc.

3-Dimensiones

Ventajas:

- Capaces de manejar variaciones de poses.
- Menos sensibles a variaciones de luz.
- Mayor precisión en el reconocimiento.

Desventajas:

- Dispositivos caros.
- Elevado consumo de tiempo para reconstruir los modelos.
- Falta de colecciones grandes de imágenes en 3D.

Una vez conocida la diferencia entre los posibles formatos de las imágenes a tratar nos adentramos en las técnicas usadas en el reconocimiento facial. Para ello vemos el siguiente diagrama en el que se diferencian tres ramas principales.

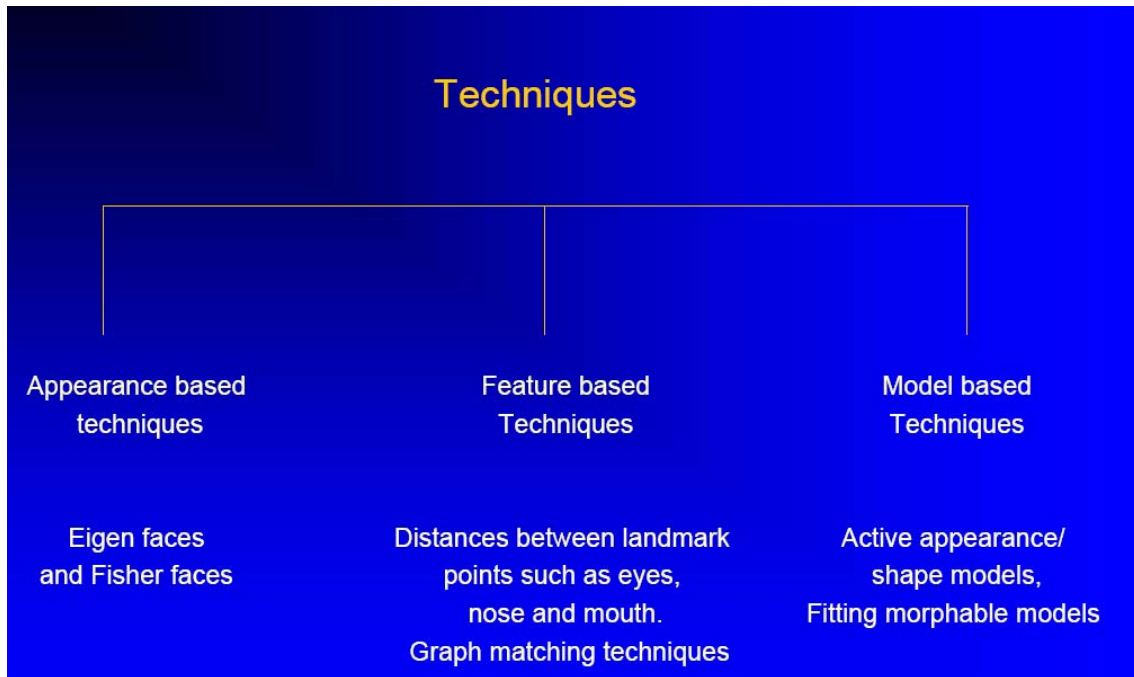


Figura 8: Diagrama de clases de técnicas [25]

Se pueden estructurar las técnicas en tres ramas principales:

- Técnicas basadas en apariencia: Aquellas que usan la apariencia general del rostro y suelen ser técnicas algebraicas de reducción de bases o del espacio para obtener componentes principales o información relevante. Por ejemplo están las técnicas de Eigen Faces y Fisher Faces.
- Técnicas basadas en Características: Son técnicas que usan la extracción de características como su principal función.
- Técnicas basadas en modelos: Usan modelos o “templates” para hallarlos en las caras ya sea para la extracción de características así como para la búsqueda de propiedades.

A continuación, explicaremos las técnicas que se utilizan para el reconocimiento facial, inicialmente explicando el funcionamiento de los más conocidos y más adelante viendo las principales novedades que se están desarrollando así como las últimas técnicas investigadas y desarrolladas en este campo.

2.3.2.1 Técnicas basadas en Apariencia

PCA (Principal Component Analysis)

Dicha técnica es una de las más extendidas en el mundo de reconocimiento facial y se basa en encontrar un subespacio de dimensión inferior al espacio de partida, la imagen, cuyos vectores base se corresponden con la dirección de máxima varianza del espacio de la imagen original. Esta técnica, conocida como Eigen Análisis (también conocido como EigenFaces), se aprovecha que las caras contienen muchas características, la mayoría de ellas comunes a todas las caras y algunas con información muy discriminante. Por lo tanto, la imagen se proyecta en un espacio donde son más relevantes esas características discriminantes y a su vez es un espacio de dimensiones menor. Por lo tanto, este método extrae las componentes principales llamadas así ya que indican las dimensiones que tienen una varianza alta [26, 27, 28].

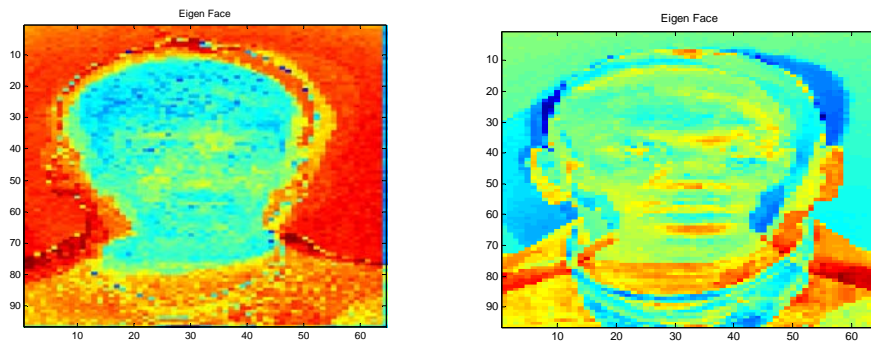


Figura 9. Ejemplo de Rostros proyectados sobre distintos subespacios extraídos con Matlab de un conjunto de rostros de FERET.

Una ventaja de esta técnica, a la hora de identificar un rostro en una base de datos radica, es la facilidad para conseguir esa identificación ya que supone decidir entre distancias de coordenadas en un espacio. A continuación mostramos un ejemplo donde se puede observar las coordenadas de distintos rostros sobre un espacio de 2D (escogido para simplificar el ejemplo).

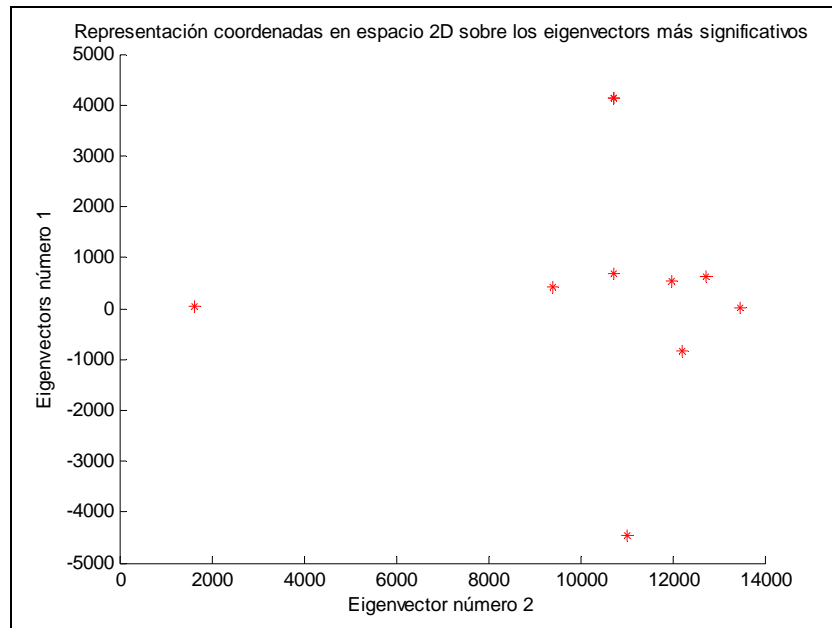


Figura 10. Ejemplo de coordenadas del conjunto de rostros sobre los 2 autovectores más relevantes. Cada asterisco representa la coordenada de un rostro.

LDA (Linear Discriminant Analysis)

Esta técnica se basa conseguir discriminar de forma lineal entre las características de los rostros. Para ello, primero obtiene las características que contienen mayor información relevante en las caras para su posterior clasificación. Esto se hace mediante una extracción de características basadas en proyección, al encontrar el espacio subyacente que mejor discrimina entre las clases y posteriormente se clasifica. Por lo que el objetivo es minimizar la separación dentro de una misma clase a la vez que se maximiza la separación entre clases diferentes. Con esta técnica se obtiene un buen clasificador pero se ha demostrado que se obtienen mejores resultados si se usa en combinación con PCA [29,30].

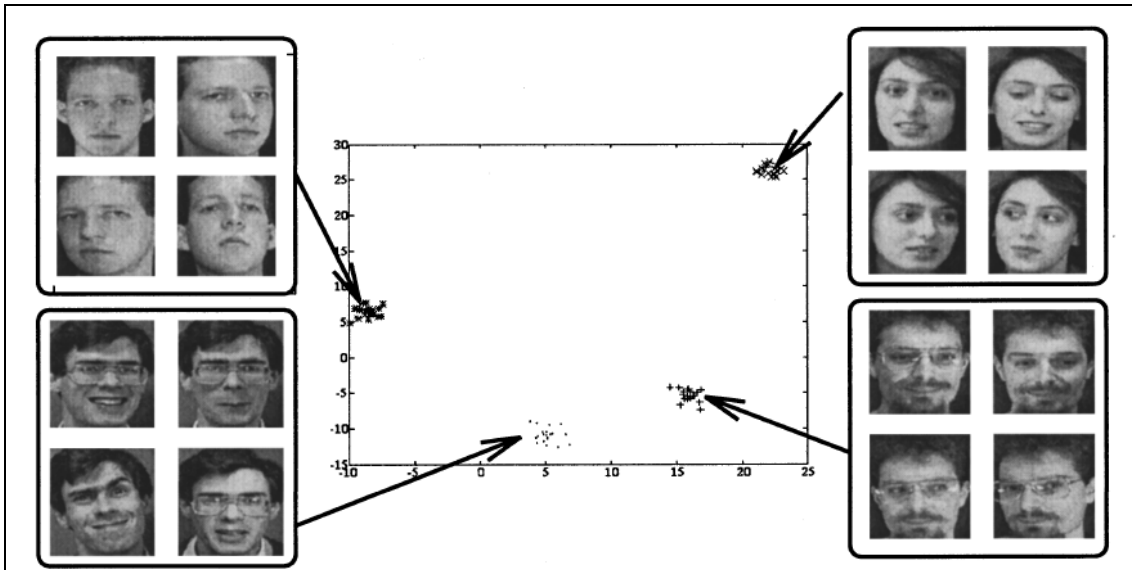


Figura 11. Ejemplo de los resultados de LDA donde vemos mayor separación entre los distintos clusters, diferentes individuos, y mínima separación dentro de la misma clase o cluster, distintas poses de un individuo.[30]

A continuación vemos un ejemplo de una combinación de las técnicas LDA y PCA:

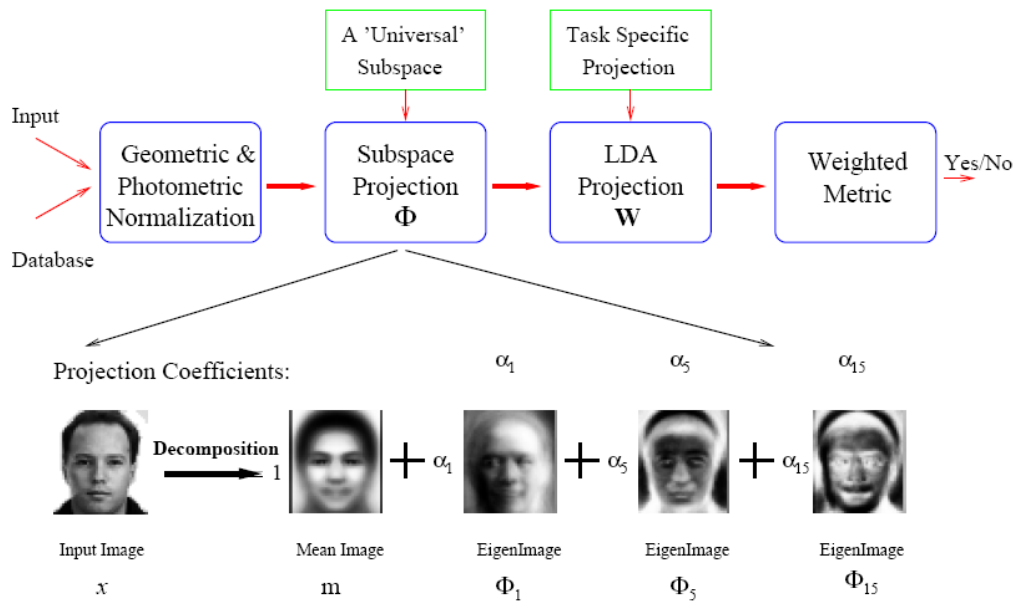


Figura 12. Un sistema de reconocimiento facial con subespacio LDA.[33]

ICA (Independent Component Analysis)

El algoritmo ICA es una generalización de PCA, el cual halla un conjunto de bases de forma que minimiza las dependencias de órdenes superiores para conseguir que la información proyectada sobre dicha base sea estadísticamente independiente. La principal diferencia es que ICA toma en cuenta los estadísticos de orden superior con lo

cual se obtiene una representación de los datos más independiente y por tanto, potente. [31,32].

EP (Evolutionary Pursuit)

Esta forma de trabajo se basa en espacios Eigen de forma adaptativa buscando el mejor conjunto de bases para maximizar una función de evaluación midiendo a su vez la precisión de clasificación y la habilidad de generalización. Debido a que el espacio de solución es muy amplio, se usan este tipo de algoritmos genéticos. [35]

2.3.2.2 Técnicas basadas en Características

Este tipo de técnicas fueron las primeras utilizadas al medir la distancia entre ciertos puntos de la cara, los ojos, nariz y boca. La debilidad de esta técnica es su poca robustez frente a cambios en la pose así como la identificación de los puntos clave.

GRAPH MATCHING

Existen varias técnicas que manejan grafos como herramienta en los algoritmos. Esta tipo de técnicas se asemejan al desarrollo en este PFC al identificar puntos característicos y formar un grafo, pero la diferencia radica en las relaciones entre los nodos, la forma de extracción de características así como en la definición y elección de dichas características. Por eso esta técnica ha sido muy utilizada dado la gran combinación de posibilidades que existen a la hora de seleccionar unos criterios de diseño que obtienen distintos resultados.

- **3D GRAPH MATCHING**

Esta técnica utiliza una malla en 3D para identificar los puntos clave hallando los puntos de máxima curvatura en los perfiles. Dichos puntos y su relación entre ellos son luego representados en un grafo.

- **EBGM (Elastic Bunch Graph Matching)**

Este algoritmo se apoya en la técnica de grafos en 2D usando grafos en estrella, ya que los rostros humanos comparten una estructura topológica similar, cuyos nodos son puntos clave (ojos, narices,..) y las aristas las distancias entre ellos. Cada nodo contiene unos coeficientes de wavelets para distintas escalas y orientaciones y finalmente el reconocimiento se basa en grafos etiquetados. [34]

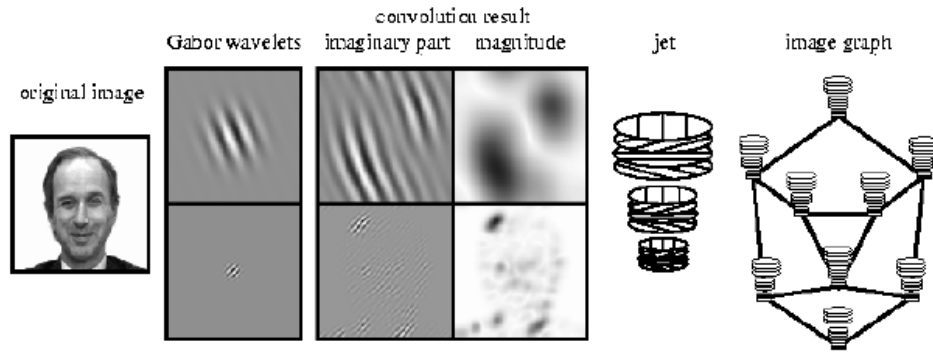


Figure 1: The graph representation of a face is based on the Gabor wavelet transform, a convolution with a set of wavelet kernels. These have the shape of plane waves restricted by a Gaussian envelope function. We compute 40 coefficients (5 frequencies \times 8 orientations). Coefficient phase varies approximately with wavelet frequency (see imaginary part), magnitude varies slowly. The set of 40 coefficients obtained for one image point is referred to as a jet (for clarity, only 3 frequencies and 4 orientations are represented in the figure). A sparse collection of such jets together with some information about their relative location constitutes an image graph, used to represent an object, such as a face.

Figura 13: Esquema de funcionamiento de EBGm [34]

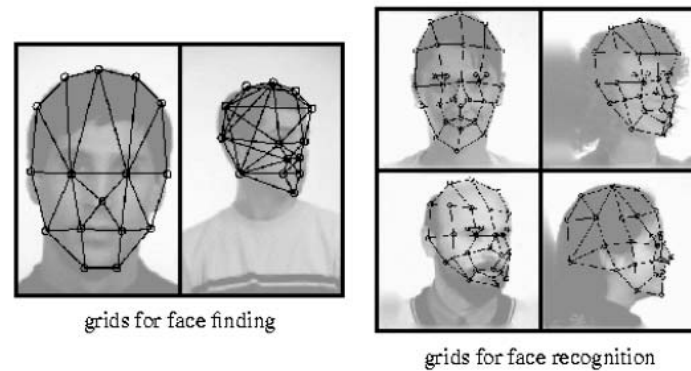


Figure 4: Object-adapted grids for different poses. The nodes are positioned automatically by elastic graph matching against the corresponding face bunch graphs. The two images on the left show originals with widely differing size and grids, as used for the normalization stage with many nodes on the outline for reliable face finding. The images on the right are already rescaled to normal size. Their grids have more nodes on the face, which is more appropriate for recognition (The grids used in Section 3.2 had about 14 additional nodes, which, for simplicity, are not shown here). One can see that, in general, the matching finds the fiducial points quite accurately. But mismatches occurred, for example, for the bearded man. The chin was not found accurately; the leftmost node and the node below it should be at the top and the bottom of the ear respectively.

Figura 14: Representación de la malla para reconocimiento [34]

2.3.2.3 Técnicas basadas en Modelos o Plantillas

Estas técnicas combinan modelos/plantillas o “templates” de formas y texturas para componer un modelo de apariencia. Los modelos de formas se construyen identificando las posiciones de ciertas características o puntos clave, mientras que los modelos de textura representan la intensidad en la escala de grises. De esta combinación y de forma iterativa se consigue un reconocedor facial obteniendo la mejor combinación de ambos que se asemeje a la imagen.

3-D Morphable Model

El fundamento de esta técnica radica en el hecho que los rostros humanos son superficies en 3D, por lo que se ha extraído un modelo que beneficiará el reconocimiento frente a variaciones de pose, iluminación, etc. En concreto, para 3D se crea el modelo de la cara teniendo en cuenta la forma y la textura para posteriormente recubrirla con una sola imagen. [36]



Fig. 1. Generation of the 3D model. The top images were used to compute the 3D model. The bottom images are synthetic images generated from the 3D model.

Figura 15: Plantilla 3D de un rostro [36]

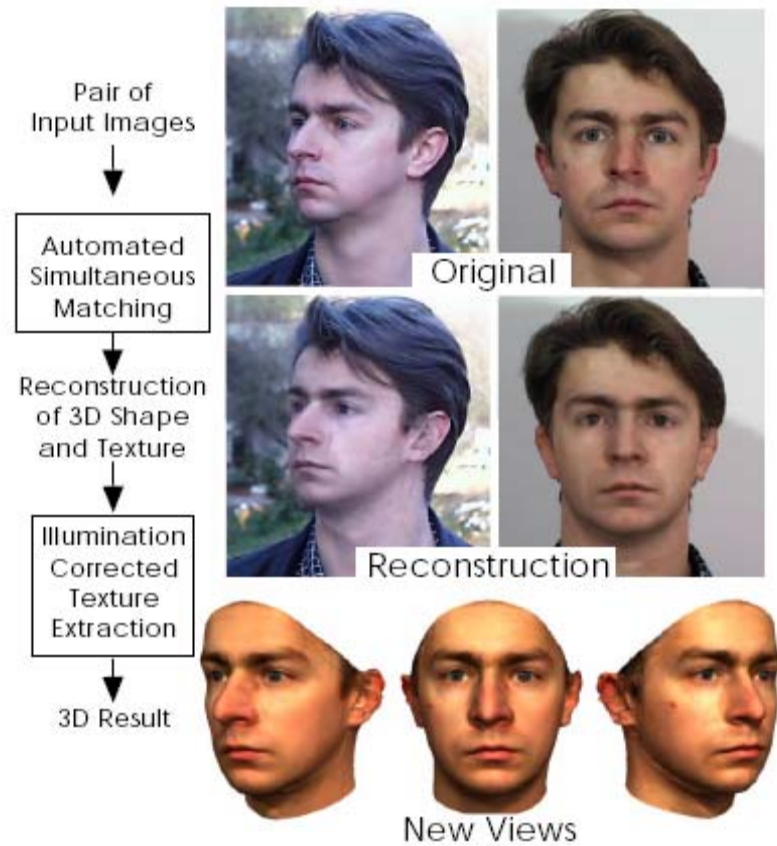


Figure 5: Simultaneous reconstruction of 3D shape and texture of a new face from two images taken under different conditions. In the center row, the 3D face is rendered on top of the input images.

Figura 16: Diagrama de Flujo [36]

A parte de estas técnicas resumidas, existen muchas más técnicas, que por su naturaleza, no se pueden ubicar directamente en estas clases, pero el objetivo es conseguir una visión general de las tendencias de las técnicas en reconocimiento facial. Una vez acabado el repaso de todas las técnicas, se van a analizar las líneas actuales de investigación.

2.3.3 Estado del Arte en Reconocimiento Facial

Una vez vistas las principales técnicas usadas en el reconocimiento facial, se van a presentar, a continuación, las nuevas propuestas e investigaciones en este campo. Estas novedades, aún estando en un mismo campo, el reconocimiento facial, se pueden separar por especialidades ya que existen muchas direcciones de investigación. Se investiga por separado la forma de extraer características, la definición de dichas características, su relevancia, su invarianza, las relaciones entre ellas, las distintas formas de presentación de la información a tratar, ya sea en 2D o 3D, tipos de algoritmos, estadísticos, algebraicos, espectrales... Por ello, no se puede definir una única dirección de investigación en este campo si no fuera porque se busca el mismo fin común.

A día de hoy, muchas de las técnicas se centran en imágenes 3D comparando las superficies de forma independiente de las expresiones faciales. Para ello se deben conseguir tanto el rango de la imagen como la textura de la cara para luego preprocesar el rango eliminando elementos que dificulten el reconocimiento como puede ser el pelo. Vemos que estas características que anteriormente se consideraban claves a día de hoy han perdido su relevancia centrándose más en la forma general. De esta forma, se consigue una forma canónica de la superficie del rostro, insensible la orientación o expresión del rostro [37].

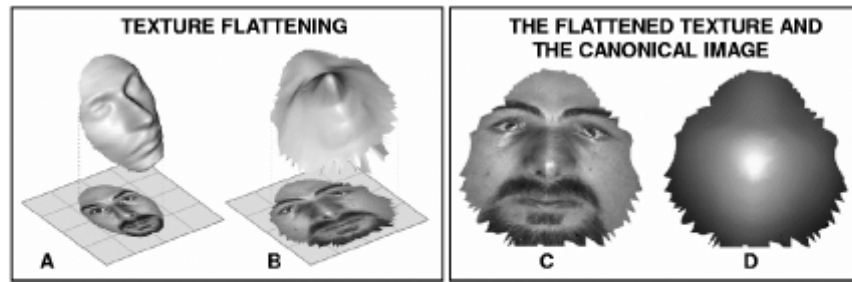


Fig. 1. Texture flattening by interpolation onto the X^1X^2 plane: texture mapping on the facial surface (A) and on the canonical form (B); the resulting flattened texture (C) and the canonical image (D)

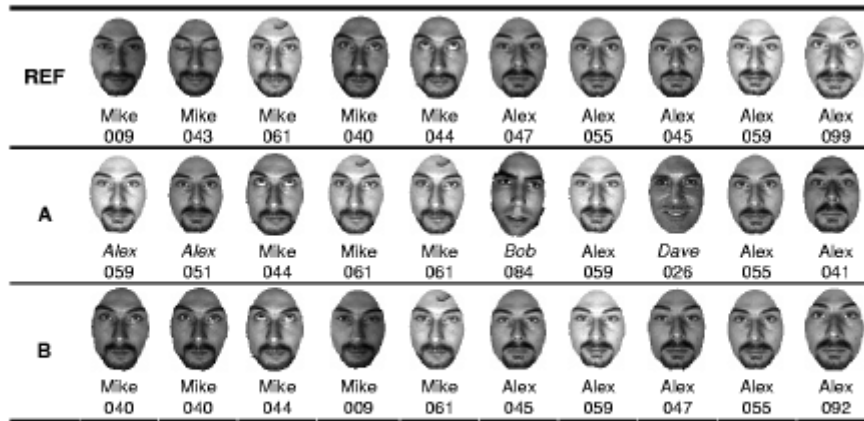


Fig. 2. The closest matches, obtained by eigendecomposition of range images (A) and canonical images (B). Wrong matches are italicized

Figura 17: Ejemplos de formas canónicas de rostros[37]

Análisis Espectral: Otra de las ramas investigadas es en el uso de características espectrales para la extracción de características. El SFA (Spectra Feature Analysis)[18] se aprovecha de las últimas investigaciones que afirman que las imágenes de caras residen en un subespacio no lineal. Las ventajas de este método de extracción son que no se ven afectados por tamaños pequeños de las muestras, además de poder extraer la información más relevante de los datos pudiendo usar LDA a pesar de la falta de linealidad.

Aplicaciones

Para finalizar, se van a presentar algunos de los modos de aplicación así como aplicaciones donde se usan los sistemas de reconocimiento facial. Simplificando el “scope” de los modos de aplicación, se pueden clasificar en tres grupos principales:

- Verificar la identidad: Esta aplicación sirve para autenticar la identidad que la persona afirma ser.

- Identificación: Una vez extraídas las características se compara con una base de datos y mediante cierto algoritmo se toma una decisión de la identidad.
- Búsquedas: Sigue el mismo algoritmo que la identificación con la diferencia de necesitar un umbral de acierto .

A su vez, estos tres modos de aplicación se pueden ver hoy en día desarrollados y usados en diversos campos en aplicaciones específicas como podemos ver en la siguiente tabla:

Areas	Specific Applications
Biometrics	Drivers' Licenses, Entitlement Programs
	Immigration, National ID, Passports, Voter Registration
	Welfare Fraud
Information Security	Desktop Logon (Windows NT, Windows 95)
	Application Security, Database Security, File Encryption
	Intranet Security, Internet Access, Medical Records
	Secure Trading Terminals
Law Enforcement and Surveillance	Advanced Video Surveillance, CCTV Control
	Portal Control, Post-Event Analysis
	Shoplifting and Suspect Tracking and Investigation
Smart Cards	Stored Value Security, User Authentication
Access Control	Facility Access, Vehicular Access

Table 1: Typical applications of face recognition.

Figura 18: *Tabla con las aplicaciones típicas de sistemas de reconocimiento. [33]*

2.4 Estado del Arte : Combinación de ambos campos

Para finalizar, se comentarán unas investigaciones que se han desarrollado últimamente donde se han juntado ambos campos, tanto el tratamiento de imágenes como la minería de información extraída de dichas imágenes. Este tipo de desarrollos se están extendiendo para abarcar prácticamente el resto de la información restante que existe en el mundo digital. Como se expreso al comienzo del proyecto, la mayoría de la información se puede encontrar en formato textual, pero existe la parte restante que se almacena en formas de imágenes, multimedia o audio. Este es el caso que nos interesa por lo que a continuación veremos las últimas tendencias en esta combinación de campos.

Existe un campo denominado CBIR (Content Based Image Retrieval) que engloba la combinación de ambos dominios. Este campo intenta extraer el contenido de una imagen apoyándose en distintas técnicas. El problema es que en la mayoría de los

algoritmos desarrollados, es necesaria la intervención humana introduciendo con ello una parte subjetiva al algoritmo puesto que se asume una parte semántica inherente en el uso humano. Por ello, existe un vacío entre las características que se extraen de forma automática, denominadas “low-level”, y el significado semántico que se busca en ellas, denominadas “high-level semantics”. Para reducir este vacío, se están investigando diversas técnicas desde aprendizaje activo, enlazado de métrica semántica, anotación oculta, etc, con la finalidad de propagar la semántica de unos objetos a otros. A continuación se va a resumir un par de ejemplos de estas investigaciones.

El campo que más interés está generando es la detección automática de objetos, es decir sin supervisión ni intervención humana. Una de las investigaciones que se ha desarrollado en este tema se denomina “Semantic-Shift” [46] y se apoya en una modificación de PLSA (Probabilistic Latent Semantic Analysis), una técnica de aprendizaje automática, usada para el reconocimiento de objetos. El propósito es conseguir un entendimiento automático de una imagen y para ello un reconocimiento de los objetos que existen en ella. De primeras, es un propósito complicado dado que la disposición de los objetos en una imagen puede ser compleja así como lo subjetivo que es el entendimiento de una imagen en sí. Cada individuo tiene su propia apreciación de la realidad. Por lo que se utiliza un método de aprendizaje automático denominado PLSA que ha sido probado anteriormente en el campo textual y lingüístico. La base de esta técnica radica en que un documento contiene una mezcla de “tópicos”, y cada uno de estos “tópicos” esta compuesta a su vez por una mezcla de palabras. Por tanto el potencial está en el hecho de poder encontrar un “tópico” en un documento gracias a parejas documento-palabras. Si trasladamos esta forma de trabajo al dominio de las imágenes, los documentos serían las propias imágenes y las palabras serían palabras visuales como características o puntos clave. Habiendo definido esto, se podría aplicar PLSA a un conjunto de imágenes y extraer distintos tópicos. Pero la aplicación de esta técnica directamente tiene un problema al no tomar en cuenta la estructura espacial en una imagen por lo que se ha desarrollado una variante denominada “Semantic Shift” en la cual se realiza de forma iterativa y se añade una componente espacial. Con esto se actualiza después de cada iteración la probabilidad de que cada “palabra” pertenezca a un tópico.

Para comenzar, se usa el detector de aristas de Canny y se muestrean los puntos de cada una de las aristas. Estos puntos se denominan, puntos de interés y se calculan los descriptores SIFT, descritos con anterioridad, alrededor de estos puntos. Por último, se obtiene un conjunto finito de “palabras visuales” aplicando el algoritmo de clustering K-medias que formarán el diccionario con el que luego se decidirá en función de las distancias a estas “palabras visuales”.

Para finalizar vemos un diagrama de flujo del algoritmo, aplicando PLSA para localizar y estimar el objeto en primer plano y un ejemplo de los resultados obtenidos aplicando “Semantic-Shift” y viendo como evolucionan los resultados a medida que avanzan las iteraciones.

Step 1 Compute $P(z|d, w)$, $P(w|z)$, and $P(z|d)$ from training data using PLSA. The first one is used in Step 2 and 3, and the other two are used in Step 4 to initialize Semantic-Shift.

Step 2 Foreground topic identification.

Step 3 Compute location and scale estimate for each image to initialize $P(x|d, z)$ in Semantic-Shift.

Step 4 Compute $P(w|z)$ from training data using Semantic-Shift.

Step 5 Run Semantic-Shift on test data.

Figura 19. Etapas del Sistema.

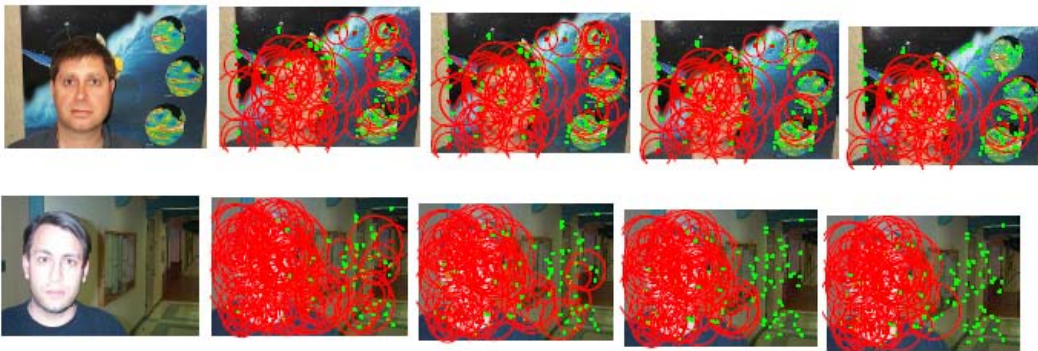


Figure 4. Test (inference) stage of Semantic-Shift.

Figura 20: Ejemplos de resultados donde las elipses rojas equivalen a topics de primer plano mientras que los puntos verdes son el t3pico "fondo". Vemos los resultados para las iteraciones 1,6, 12 y 26.

Como se ha podido observar, este m3todo obtiene una detecci3n de forma autom3tica de objetos invariante a traslaci3n y escalado que mejora a su vez investigaciones previas usando PLSA b3sico.

Otra t3cnica que se ha investigado en los 3ltimos a3os ha sido la propagaci3n de forma autom3tica de la sem3ntica para a3adir a los sistemas de extracci3n de im3genes precisi3n, velocidad y mejora del rendimiento. Uno de los estudio sobre este tema usa estructuras jer3rquicas de sem3ntica a dos niveles, uno a nivel local y otra a nivel global [48]. Este m3todo propaga la sem3ntica de una imagen con aquellas im3genes relacionadas basada en la similitud visual y en las relaciones sem3nticas, actualizando las relaciones entre las im3genes y la red sem3ntica. Lo que se propone es usar m3todos CBIR junto con redes sem3nticas como se aclara en el esquema siguiente:

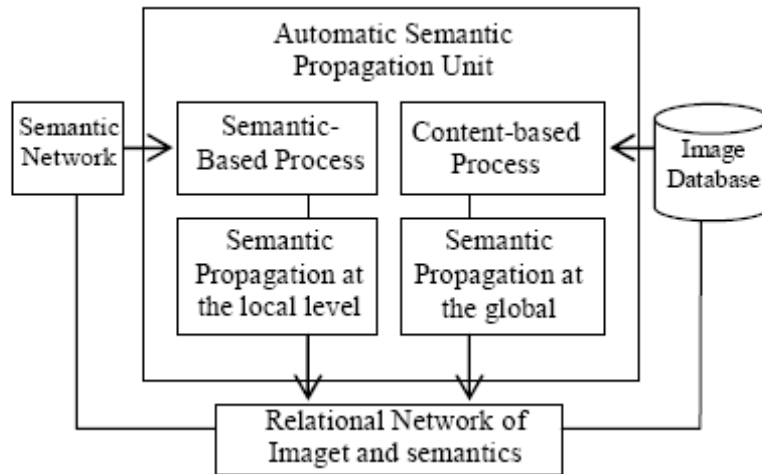


Figura 21: Esquema de trabajo propuesto. [48]

La forma de trabajo es por un lado trabajar con la información local de cada imagen y en paralelo con la información global de todo el conjunto de imágenes. Inicialmente, se crea una red semántica de forma jerárquica donde los nodos son sinónimos y las relaciones son generalizaciones o especializaciones. Luego cada imagen se relaciona con distintos nodos asignándoles un peso. Inicialmente ese peso es igual para todos y con las actualizaciones, los pesos varían hasta lograr el óptimo que mejor les describa. De esta forma se puede obtener la extracción y clasificación de imágenes según un mismo tema de forma automática o como respuesta a peticiones explícitas.

Como último ejemplo de las investigaciones sobre estos campos, se va a describir otra metodología denominada Hierarchical Semantics of Objects (hSO) [49]. El fundamento de esta metodología radica en que existen ciertos objetos que se dan en la misma situación o emplazamiento y que por tanto, están relacionados semánticamente.

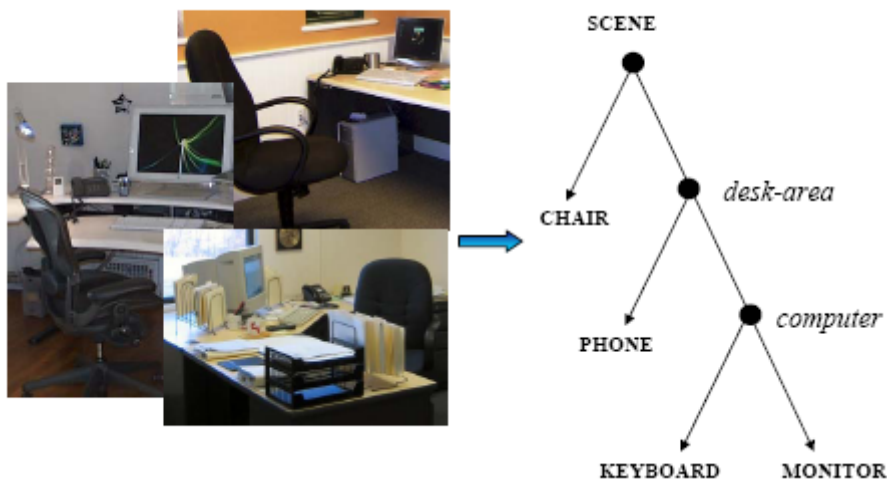


Figura 22: Ejemplo de colección de imágenes y el hSO extraído.

En la figura previa vemos como de un conjunto de imágenes se consigue extraer una red semántica jerárquica. De esta forma, este algoritmo puede identificar los objetos

importantes en una escena, descubrir las relaciones entre estas y otras partes de la imagen obteniéndose un buen contexto para detectores robustos de objetos.

2.5 Extracción de Características

En el campo general de tratamiento de imagen, la extracción de características es uno de los mayores puntos de interés y más aun en el reconocimiento facial. La forma de extraer un punto en concreto, es decir, una característica, se puede realizar de muchas formas, por ejemplo, partiendo de algoritmos de búsqueda que localizan la característica en la imagen, y una vez localizados, segmentándolos y extrayendo la información. Los algoritmos de búsqueda pueden ser a su vez muy diversos, desde algoritmos simples que usan plantillas y buscan mediante correlación con todos los píxeles de unas imágenes hasta algoritmos más complejos que extraen características espectrales, algebraicas, invariantes,...

Uno de los temas más estudiados en la actualidad son los denominados puntos o características invariantes. Estos puntos se caracterizan por permanecer invariantes frente a rotaciones, traslaciones o incluso escalados de una imagen. El origen del estudio de estos puntos se debe a las debilidades de otras técnicas como aquellas basadas en modelos, puesto que se debe tener modelos a priori, y basadas en apariencia, en las que el acierto decrece cuando existen oclusiones o bajo distintas condiciones de iluminación o pose. Debido a estas debilidades surgieron técnicas que combinan características geométricas locales con características de apariencia. Estas técnicas se usan no sólo en el reconocimiento facial sino también en el reconocimiento de objetos, dado que características de apariencia en un rostro se pueden tomar como objetos independientes con una cara como fondo.

Una de las técnicas más extendidas en cuanto a extracción de características invariantes son los descriptores SIFT (Scale Invariant Feature Transform) [42]. El objetivo de este método es la extracción de características que son invariantes a la rotación, el escalado y parcialmente invariantes a cambios de iluminación. El resultado de aplicar dicha técnica es la obtención de características distintivas. A grandes rasgos se va a describir el algoritmo usado:

El primer paso es la búsqueda de las características candidatas que se repiten bajo distintas condiciones de escalado. Esto se consigue usando una función continua conocida como “Scale Space” [47] que consiste a su vez en el uso de diferencias de gaussianas y la comparación de cada punto con sus vecinos tanto en su escala como en escalas siguientes y anteriores. De esta forma se extraen los candidatos. El siguiente paso consiste en eliminar aquellos candidatos que poseen poco contraste ya que son sensibles al ruido y ciertos bordes.

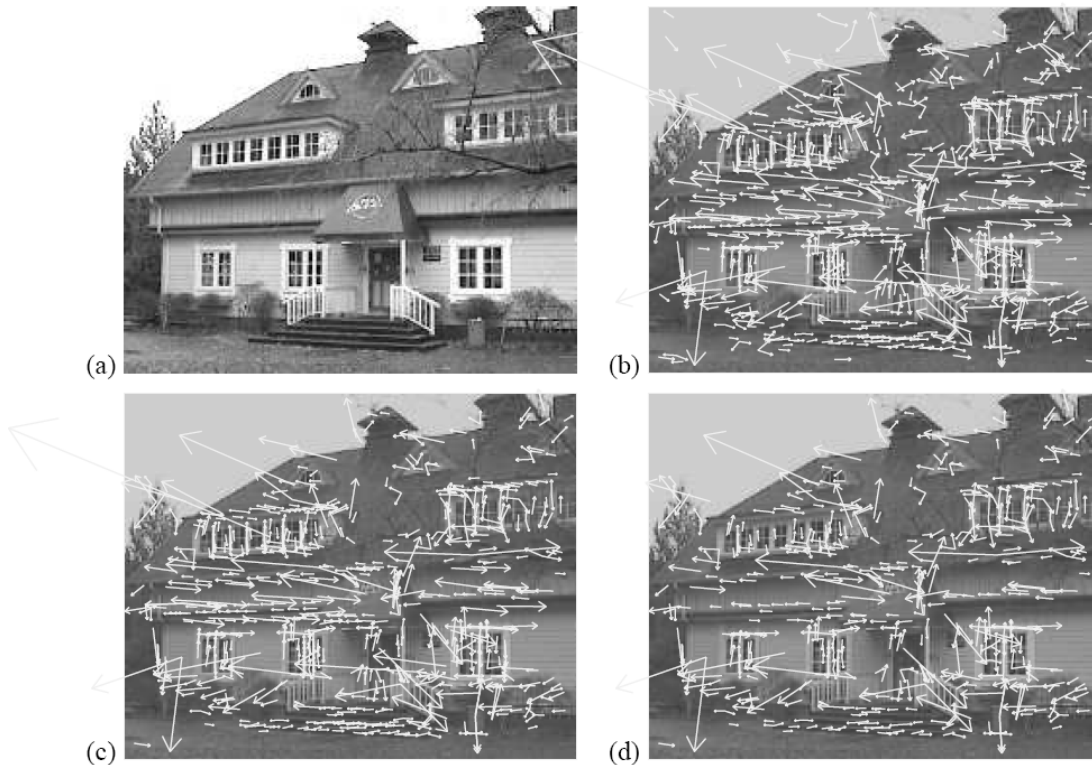


Figure 5: This figure shows the stages of keypoint selection. (a) The 233x189 pixel original image. (b) The initial 832 keypoints locations at maxima and minima of the difference-of-Gaussian function. Keypoints are displayed as vectors indicating scale, orientation, and location. (c) After applying a threshold on minimum contrast, 729 keypoints remain. (d) The final 536 keypoints that remain following an additional threshold on ratio of principal curvatures.

Figura 23: Ejemplo de la extracción de candidatos y posterior reducción de los puntos[42].

El siguiente paso es la asignación de orientación a cada punto para conseguir la invarianza frente a la rotación. Esto se consigue usando un histograma de orientación que se forma con la orientación de los gradientes de los puntos alrededor de cada uno de los “keypoints”, puntos clave. Estos gradientes, usados con una campana gaussiana centrada en el punto clave, proporcionan la dirección dominante local que se le asigna al punto clave analizado.

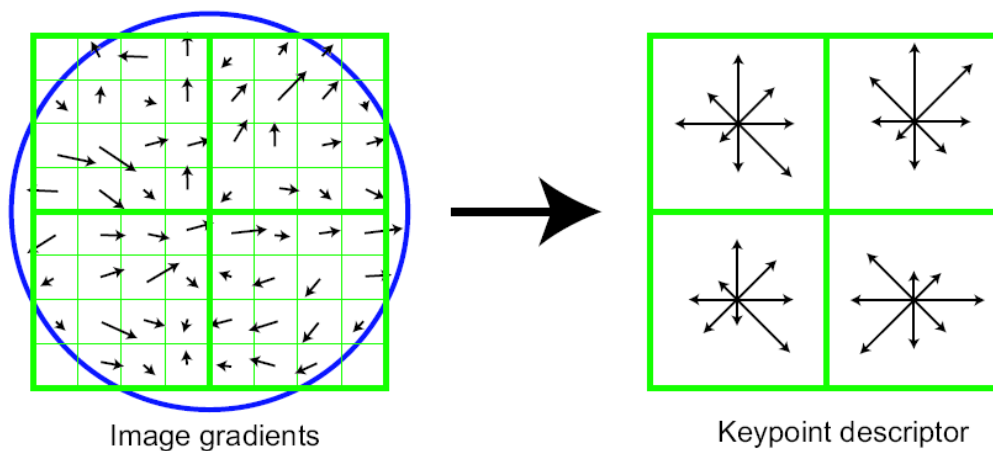


Figura 24: Ejemplo de la extracción de la orientación en los puntos clave.[42]

Finalmente se obtiene por cada punto clave una posición, una escala y una orientación por lo que conocidos estos parámetros se tiene a su vez un descriptor invariante.

Pero los descriptores SIFT no son la única investigación en el campo de la extracción de características invariantes. En la UPC-CSIC de Barcelona [43], también se ha realizado un estudio sobre características invariantes a la rotación desde otro enfoque. La principal diferencia es la selección de características multiclase pero con un fuerte interés en el cálculo rápido de clasificadores débiles invariantes a orientación [43] con el fin de conseguir un sistema de reconocimiento de múltiples objetos invariantes a rotación. Para la selección de características, las que mejor discriminan un objeto se obtienen gracias a la convolución de muestras de imágenes positivas con un conjunto simplificado de funciones bases *wavelets* a diferentes escalas y orientaciones. De esta forma, se consiguen regiones representados por contornos, puntos y líneas, las cuales de forma conjunta pueden representar un objeto.

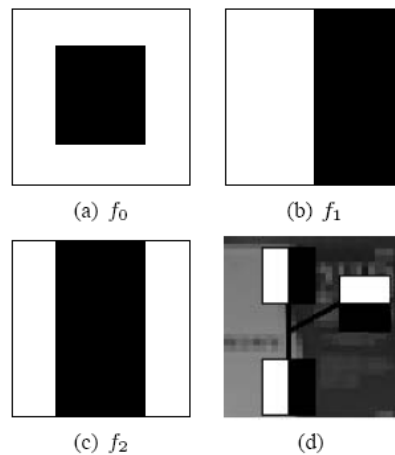


Figura 1: Conjunto de bases *wavelet* simplificado. a) punto b) contorno, c) línea, y d) características locales del objeto

Figura 25: Conjunto de bases *wavelets*[43]

En la figura vemos una representación de las bases utilizadas así como un ejemplo del conjunto de las características locales de un objeto.

Una vez obtenidos los objetos, se usa una estructura jerárquica de características para detectar las múltiples clases de objetos. Inicialmente, se encuentran las características comunes entre los objetos y posteriormente se distinguen las características diferenciativas. Para el cálculo de la orientación han usado filtros con bases de Haar y una imagen integral de orientación para obtener la orientación local de los puntos. Estos filtros tienen selectividad a la orientación espacial como también a frecuencia, y producen características que capturan el contraste entre regiones representando contornos, puntos y líneas.

Otra técnica es la denominada “Esquinas Harris”. Esta técnica desarrollada por Yokono y Poggio [44,45] usa las esquinas de Harris a varios niveles de resolución como puntos de interés, y de estas, selecciona como características aquellas que son más robustas a

filtros bajo rotación y escala. Los filtros usados son derivadas de Gaussianas orientadas, dado que las derivadas no son invariantes a rotación. Con esto se obtiene las características según la orientación local de gradientes alrededor del punto de interés.

Una vez vistas algunas de las técnicas usadas para la extracción de características invariantes se va a ver en que tipo de modelos se usa. Uno de los modelos donde se han aplicado estas técnicas es en ASM (Active Shape Models) [41]. La técnica inicial incorpora un conjunto reducido de características invariantes que usa como descriptores de imagen. Estas características son invariantes a transformaciones rígidas y se selecciona un conjunto de ellas para cada punto y para cada nivel de resolución mediante SFS (Sequential Feature Selection o Selección secuencial hacia delante). A continuación se resume los pasos de este algoritmo:

- I. Conjunto S sin ninguna característica y todas las N características marcadas como no elegidas.
- II. En cada iteración, se añade a S una característica de entre las no elegidas (la que minimiza el **error cuadrático medio** (MSE)):
 - (a) Elegir únicamente la mejor característica.
 - (b) Elegir el mejor conjunto de características de tamaño dos que incluye la característica (a).
 - (c) Elegir el mejor juego de características del tamaño tres que incluye el conjunto de características (b), etc.
- III. Después de N iteraciones, el S incluye todas las características, y la selección de paradas.
- IV. El resultado es N conjuntos de características no vacíos $S_1 \subset S_2 \subset \dots \subset S_N$. El mejor entre éstos es el conjunto que minimiza MSE.

Como ventajas principales este algoritmo tiene su implementación simple y su buen desempeño en la reducción de características mientras como desventajas tiende a ir al óptimo local más cercano y es incapaz de eliminar características que se hacen obsoletas después de la adición de otras características.

La unión de ASM con extracción de características mediante SFS ha obtenido mejoras en la precisión al segmentar comparados con modelos lineales de ASM.

Otra forma que se ha estudiado es la extracción de unos descriptores de la imagen invariantes. Esto es una alternativa a la extracción directa de características, sino lo que se contempla es mediante la transformada de Fourier-Mellin, se procesa el espectro tanto de amplitud como fase de la imagen obteniéndose unos descriptores iniciales. A estos resultados se les añade un algoritmo PCA con lo que se obtiene un descriptor completo que es una representación compacta que contiene la estructura esencial de los objetos en una imagen. [40]

2.6 Descriptores Locales

Un descriptor local, es una forma de almacenar información de una región en una imagen, y que cumple la propiedad de ser invariante frente a transformaciones. Se han implementado diversas técnicas para describir regiones locales en imágenes, desde la implementación más simplificada basada en un vector de píxeles de la imagen, con el inconveniente de poseer dimensiones elevadas, hasta implementaciones más complejas. Se distribuyen principalmente en las siguientes clases:

Descriptores basados en la Distribución:

Estas técnicas usan histogramas para representar las diferentes características de apariencia o forma. Un descriptor simple es la distribución de intensidad de píxeles representada por un histograma. Existen versiones mejoradas que introducen más dimensiones como la distancia al punto central a parte del valor de intensidad. Uno de los descriptores más usados es el descriptor SIFT [42] basado en transformaciones invariantes a escala para extraer características, el cual combina un detector de región invariante a escala y un descriptor basado en la distribución del gradiente de esa región. Dicho descriptor se representa con un histograma 3D con las posiciones de los gradientes y sus orientaciones. Estas posiciones llevan un peso asociado en función de la magnitud del gradiente. Todo ello conlleva, a un descriptor robusto frente a distorsiones geométricas. A su vez, histogramas geométricos y contextos de formas, métodos parecidos a SIFT, se basan en histogramas que describen la distribución de los contornos en una región, obteniendo resultados fiables para el reconocimiento de formas en dibujos.

Técnicas Espacio-Frecuenciales

Existen varias técnicas que describen la frecuencia del contenido de una imagen. La transformada de Fourier descompone el contenido de la imagen en sus bases. El mayor inconveniente es que dichas técnicas no son aplicables localmente. Sin embargo, la transformada de Gabor, soluciona estos problemas generales, pero requieren un elevado número de filtros de Gabor para capturar los cambios de frecuencia y orientación. Este tipo de técnicas se suelen usar para la clasificación de texturas.

Descriptores Diferenciales

Conjunto de derivadas de imágenes hasta un orden fijo aproximan el entorno de un punto. Estas técnicas son invariantes frente a rotaciones debido a ciertos filtros derivados.

Una vez vistas algunas de las categorías de los descriptores, se van a ver las distintas etapas involucradas en la implementación de un descriptor local:

Detectores de Regiones

Estos detectores proporcionan las regiones que se usan para crear el descriptor. Algunos de los detectores usados actualmente son:

- Puntos de Harris: Son invariantes a rotaciones. Se fija una región de 41x41 píxeles centrado en el punto de interés.
- Regiones Harris-Lapalce: Son invariantes a rotaciones y cambios de escala. Se detectan los puntos mediante una función de Harris adaptada a escala, obteniéndose estructuras tipo esquinas.
- Regiones Hessianas-Laplace: Son invariantes a rotaciones y cambios de escala. Los puntos son localizados como el máximo local del determinante Hessiano y en escala como el máximo de la Laplaciana de la Gausiana. Esta técnica extrae regiones con estructuras tipo “blob”.

Normalización de las Regiones

Los detectores proporcionan regiones circulares o elípticas de distintos tamaños, que depende de la escala del detector. Dichas regiones se mapean a una región circular de radio constante manteniendo las variaciones de señal del detector original. A su vez, para mantener la invarianza frente a rotaciones, las regiones se rotan en la dirección de la orientación del gradiente dominante. También se normaliza la iluminación de la región usando unas transformaciones de la intensidad de los píxeles.

3. HERRAMIENTAS DE TRABAJO

3.1 Espectro de Grafos

Una de las herramientas básicas para el desarrollo del PFC son los espectros de los grafos, así como sus propiedades. A lo largo del proyecto, se usará esta herramienta dado que nos proporciona información sobre la estructura del grafo, la cual usaremos para sacar resultados y conclusiones. Para ello, en la siguiente sección se van a definir las herramientas analíticas necesarias, las cuales también han sido usadas por J.Farkas, I. Derényi, A.Barabasi y T.Bishkek en su investigación resumida en el artículo *Spectra of “real-world” grapas: Beyond the semicircle law.*[50]

El espectro de un grafo, por definición, es el conjunto de autovalores de la matriz de adyacencia del grafo. Para hallarlo, se escribe cada componente del vector sobre el correspondiente vértice del grafo y luego, en cada vértice se escribe el número de todos sus vecinos, es decir, la suma.

$$\rho(\lambda) = \frac{1}{N} \sum_{j=1}^N \delta(\lambda - \lambda_j)$$

Por otro lado, la densidad espectral del grafo puede ser relacionada directamente con las características topológicas del grafo, siendo el k-ésimo momento del grafo. Esto es realizando una relación topológica entre el número directo de caminos que vuelven al mismo vértice después de “k” pasos.

Otra de las herramientas a utilizar, son los autovalores extremos. En un grafo incorrelado, definido en el apartado de grafos [sección 3], el autovalor principal muestra la densidad de arcos y el segundo puede ser relacionado como la conductancia del grafo (para el caso de considerar el grafo como una red de resistencias). En general, existe una propiedad que se cumple, la cual es que el autovector principal de la matriz de adyacencia es no negativo (ninguna de sus componentes tiene un valor inferior a cero). A su vez, si el grafo no tiene ningún vértice aislado, se afirma que todas las componentes del autovector principal son positivas. Por último, se cumple que todos los otros autovectores son perpendiculares al principal y por tanto, sus signos serán mezclados en signo.

Por último, se usará también como herramienta, la participación inversa de la razón de los autovectores, la cual se define para el j-ésimo autovector como:

$$I_j = \sum_{k=1}^N [(e_j)_k]^4$$

Una limitación numérica que existe usando estas herramientas es cuando el número de nodos excede los 5000 dado que hay que utilizar técnicas iterativas para extraer los

autovalores, pero durante el desarrollo de este proyecto, el tamaño de los grafos siempre será inferior a dicha limitación numérica.

Resultados del Espectro para tres tipos de grafos

Una vez definidas las herramientas que se van a utilizar, se pretende observar las propiedades más importantes que podemos identificar para los distintos tipos de grafos apoyándonos en sus espectros así como los autovalores y autovectores.

Grafos Aleatorios Incorrelados: Los resultados obtenidos para este tipo de grafos son que no todos los grafos aleatorios incorrelados obtienen como espectro el semicírculo de potencia. En el siguiente gráfico vemos la simulación cuando el número de nodos tiende a infinito:

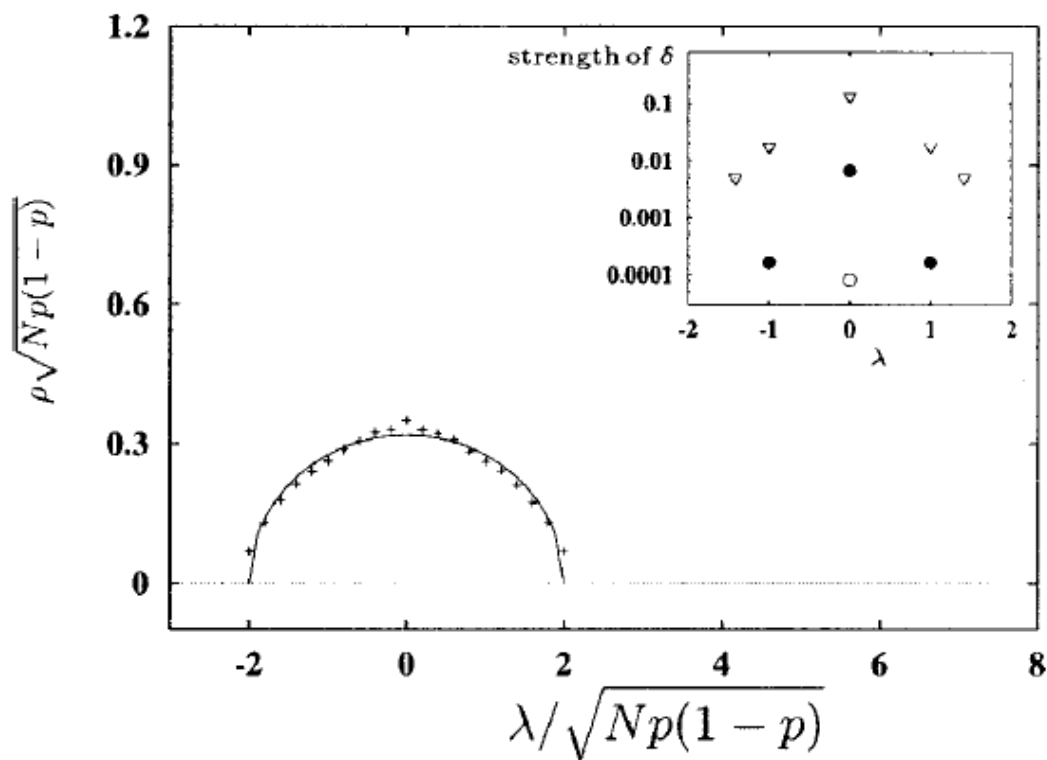


Figura 26: Representación del espectro de un grafo incorrelado [50]

Grafo del Mundo Pequeño: Como se ha visto anteriormente, para este tipo de grafos, existe un parámetro fundamental que es la probabilidad de recableado, la cual define la estructura interna del grafo. Este parámetro se puede observar claramente en el espectro de este tipo de grafos ya que para $p_r=0$, los espectros son picudos mientras que al aumentar la probabilidad, el espectro tiende al semicírculo de potencia, es decir, tiende a ser un grafo incorrelado.

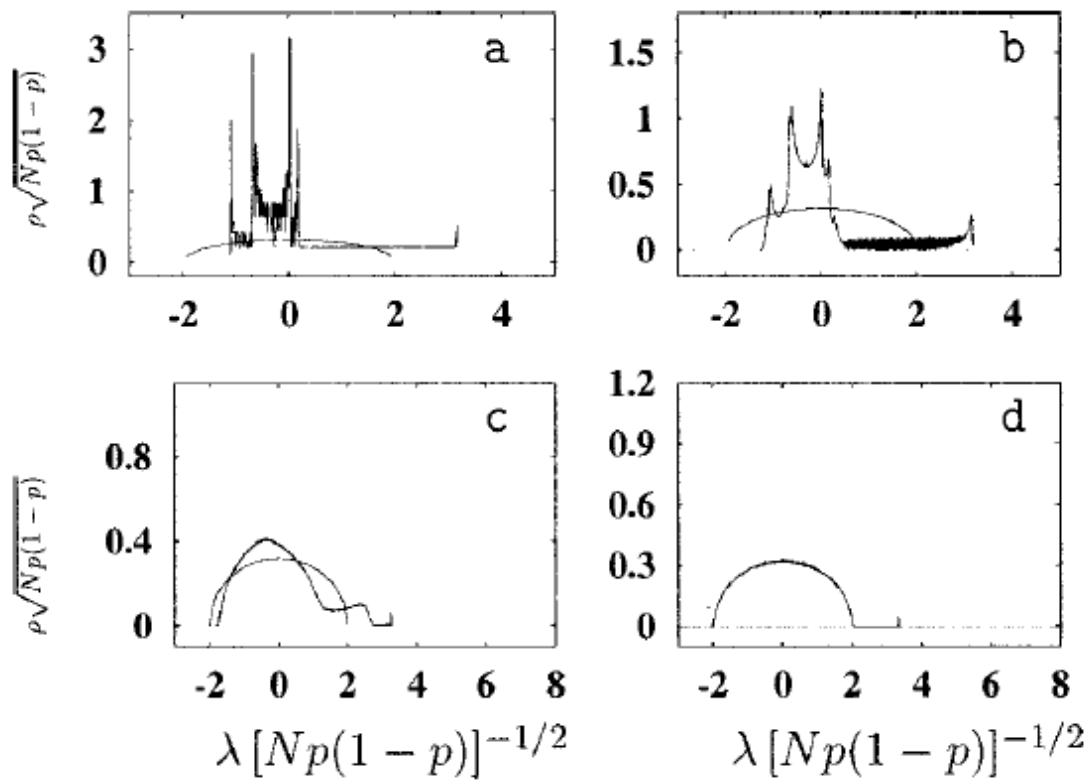


Figura 27: Representación del espectro aumentando la probabilidad (0, 0.01, 0.3, 1)

Grafos de Orden Libre: En este caso, el parámetro fundamental es el número de vértices asilados que posee. Cuando este valor es superior a la unidad, en el espectro se pueden distinguir varias partes ya que los autovalores distintos de uno convergen de manera simétrica teniendo forma de triángulo, como podemos observar en la siguiente gráfica:

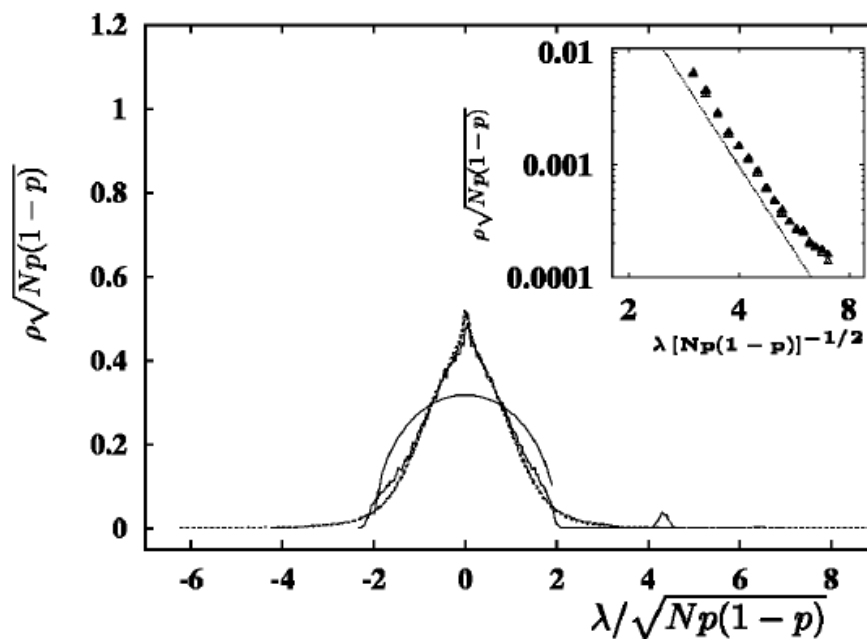


Figura 28: Espectro de un grafo de Orden Libre

Partiendo de estas herramientas, se va a utilizar el espectro de los grafos para recopilar información de la estructura del mismo y con esta información se pretende diferenciar entre grafos distintos que provienen de imágenes distintas, que para nuestro caso de estudio, coinciden con la imagen de un rostro.

3.2 Base de Datos

Para el desarrollo del proyecto se necesitaba un conjunto representativo de imágenes de rostros. Para solventar esta necesidad, se ha recurrido a la base de datos FERET [5] que posee una gran variedad de imágenes de individuos distintos.

Estas imágenes fueron tomadas inicialmente con una cámara de 35mm usando carretes Kodak Ultra film. Posteriormente, los carretes fueron procesados por Kodak y almacenados en formato digital en un CD-ROM usando una técnica multiresolución propia de Kodak para la digitalización y almacenamiento de imágenes.

La colección de imágenes se realizó en 15 sesiones fotográficas entre Agosto 1993 y Julio 1996. Cada una de estas sesiones tenía una duración de uno a dos días donde no se variaba la localización ni las condiciones. A su vez, estas sesiones consistían en conjuntos de 5 a 11 instantáneas de cada individuo donde se obtenían dos vistas frontales, una con expresión neutra, denominada “fa”, y la segunda con una expresión facial distinta, “fb”. Por el contrario, y exclusivamente en 200 individuos, se realizó una tercera vista frontal usando otra cámara y bajo unas condiciones de iluminación diferente, “fc”. El resto de las imágenes del conjunto de cada individuo son tomas laterales que incluyen los perfiles tanto derechos como izquierdos así como perfiles medio ladeados de ambos lados. Estas imágenes están digitalizadas a color pero también hay una versión en escala de grises de 8-bits.

Con todo esto, la base de datos de FERET se compone de 1564 conjuntos de imágenes teniendo un total de 14126 imágenes. De todo esta base de datos, 503 conjuntos han sido distribuidos para investigación mientras que el resto ha sido adquirido por el gobierno estadounidense para su evaluación [13].

Dado que para la realización del proyecto abarcar toda la base de datos supondría un coste tanto temporal como computacional elevado, se ha reducido el conjunto con el que se ha desarrollado la investigación. La primera pauta ha sido simplificar el conjunto de trabajo reduciendo el número de individuos y seleccionando solamente uno de las tomas. Por tanto como criterio de diseño se ha seleccionado exclusivamente las vistas frontales a color para poder extraer mayor número de características tanto por tener un área de trabajo mayor, comparado con una vista de perfil, así como la posibilidad de tener el color en distintas regiones y no solo trabajar con las tonalidades. En resumen, se ha escogido un conjunto reducido de 50 individuos para formar un corpus de trabajo lo más representativo posible. Por eso, el corpus de caras está formado por una mezcla equilibrada de individuos masculinos y femeninos además de haber intentado ser lo más neutral posible en la elección de personas de distintas razas, características, formas de caras, etc.

4. DESARROLLO

Para la consecución del objetivo de este proyecto se han seguido varias etapas de desarrollo práctico en las cuales, se han observado los resultados obtenidos y se ha trabajado sobre esos resultados variando en caso de ser necesario el enfoque. Para ello se ha usado una especie de plataforma con las distintas opciones barajadas para, finalmente, comparar los resultados y obtener unas conclusiones sobre el desarrollo.

Este proyecto se divide en tres fases bien diferenciadas:

- 1) Construcción del grafo directamente con los puntos característicos extraídos incluyendo la información geométrica de la imagen.
- 2) Procesado de los puntos característicos formando *palabras visuales* dotadas de significado, usando toda la imagen.
- 3) Segmentación de la imagen en celdas, para comparar una imagen con un texto.

Se ha trabajado, sobre todo inicialmente, en el tratamiento de la imagen, realizando un preprocesado, y posteriormente en el análisis de los grafos, tanto en el hecho de la construcción del grafo como en la normalización e interpretación de los espectros. A su vez, se han considerado distintas formas de definición de una palabra, así como distintas formas de relación entre ellas.

Este capítulo del proyecto será dividido en subapartados para la mejor comprensión de las fases usadas para el desarrollo del mismo.

4.1 Las Imágenes

La “materia prima” de este proyecto son las imágenes de los individuos con las que se trabaja. Estas imágenes, como se ha comentado con anterioridad, se han extraído de la base de datos *FERET [5]*, en las que para cada individuo existen fotografías tomadas desde varias perspectivas, de frente y de cada lateral, así como a color y en blanco y negro, además de presentarse a distintas escalas de tamaño. Para el desarrollo de este proyecto, se ha escogido trabajar con las imágenes de toma frontal, a color y en la escala más reducida, cuya dimensión es 382x256 píxeles.

La elección de dichas imágenes se basa en los siguientes argumentos:

- **Toma frontal:** Las características principales del rostro humano se observan mejor de frente, por lo cual al escoger dicha toma, incluimos mayor número de características que usando una imagen lateral. Dado que el objetivo del proyecto trata de extraer información de una imagen, está en nuestro interés obtener cuantas más características se puedan de la imagen de un individuo, por lo que se ha optado trabajar con dichas imágenes frontales.

- **Color:** La elección de trabajar con las imágenes a color se debe básicamente al mismo razonamiento planteado con anterioridad. Se busca obtener información de la imagen tratada por lo que si incluimos la variable, color en nuestro desarrollo, se trabaja con mayor información sobre el individuo. Esta variable, nos beneficiará también a la hora del preprocesado de la imagen además de proporcionar información en la extracción de características.
- **Dimensiones:** Una de las características que se buscan en este proyecto, es que los resultados obtenidos sean todo lo invariante posible, y para ello se ha decidido trabajar con una imagen reducida. Con esto se pretende, asegurar que al obtener las características en el “caso peor”, es decir, con la resolución más adversa, obviamente al tratar con una imagen más detallada los resultados serán mejores. A su vez, otra causa para este criterio de decisión se debe al coste computacional del mismo. Al trabajar con un espacio más reducido de puntos en la imagen, el tiempo del preprocesado así como el tamaño del grafo será también más reducido y el proceso total más rápido.

Todas estas razones han sido usadas para la elección del tipo de imagen, además de la necesidad de usar el mismo tipo de imágenes en general. Dado que se trata de implementar un proceso genérico para cualquier tipo de rostro, la necesidad de trabajar con el mismo formato de imagen es un requisito importante, y es la razón principal por la que se ha utilizado una base de datos, a parte del hecho de poseer una gran variedad de imágenes e individuos distintos.

A continuación se muestra para un individuo las distintas opciones existentes y la elección final, de un individuo de ejemplo de la base de datos:

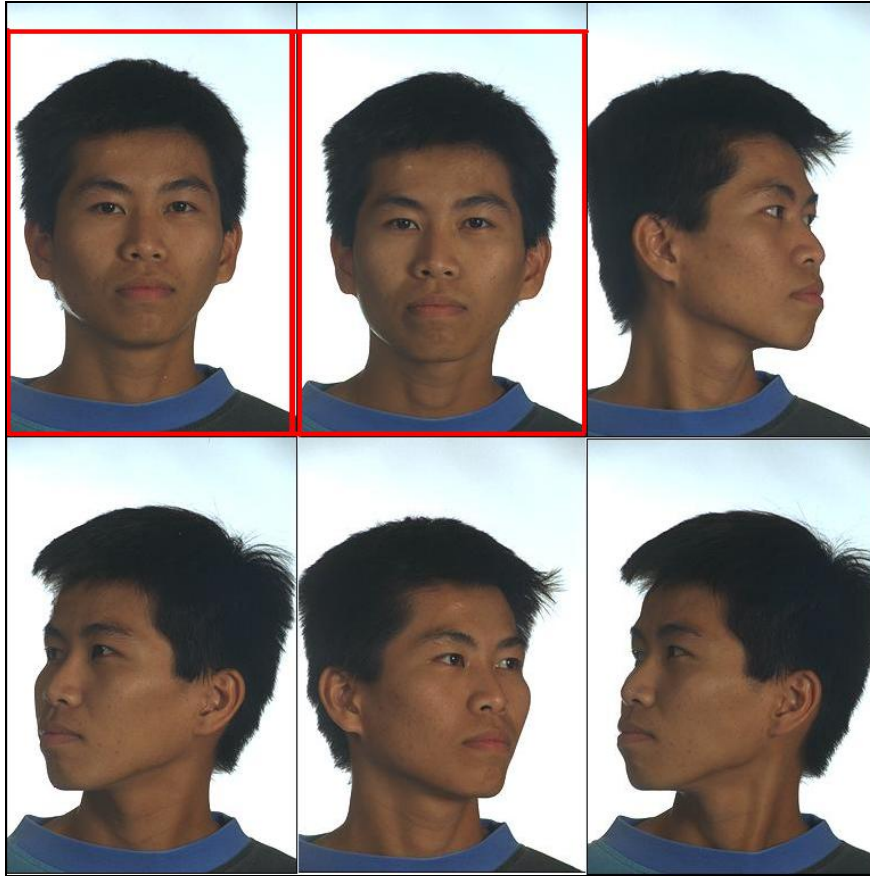


Figura 29: Distintas vistas de un mismo individuo. Enmarcadas en rojo aquellas vistas frontales con las que se ha desarrollado el proyecto.

4.1.1 Corpus de trabajo

Uno de los aspectos positivos con los que se va a trabajar es la base de datos *FERET*, la cual proporciona un corpus de trabajo muy extenso y variado. El objetivo que se busca es la identificación y extracción de información de los individuos, y con dicha base de datos se posee mucha información. Con ello se ha creado un corpus de trabajo muy extenso y variado pretendiendo que la información sea relevante sin perder su condición de generalidad. Para ello, se ha seleccionado un total entorno a los 100 individuos, con diversidad de sexos, raza y características con el fin de poder identificar esta información y ganar en generalización al usar promediados de varios individuos en vez de casos concretos o individuales que pueden alterar los resultados. Las distinciones básicas serán por sexo (hombres y mujeres), razas (blanca, asiática, de color e india) y por características (poseer bigote, barba, perilla, pelo largo, gafas, etc). A continuación se presenta una tabla con el número de sujetos para cada categoría:

Sexo: Se ha trabajado con unas poblaciones de individuos similares en cuanto a porcentajes para poder realizar las comparaciones de la forma más equitativa.

Las cantidades vienen mostradas en la siguiente tabla:

Hombres	Mujeres	
52 (61%)	33 (39%)	Grupo1
31 (56%)	24 (44%)	Grupo2

Tabla 1: Número de individuos por cada grupo.

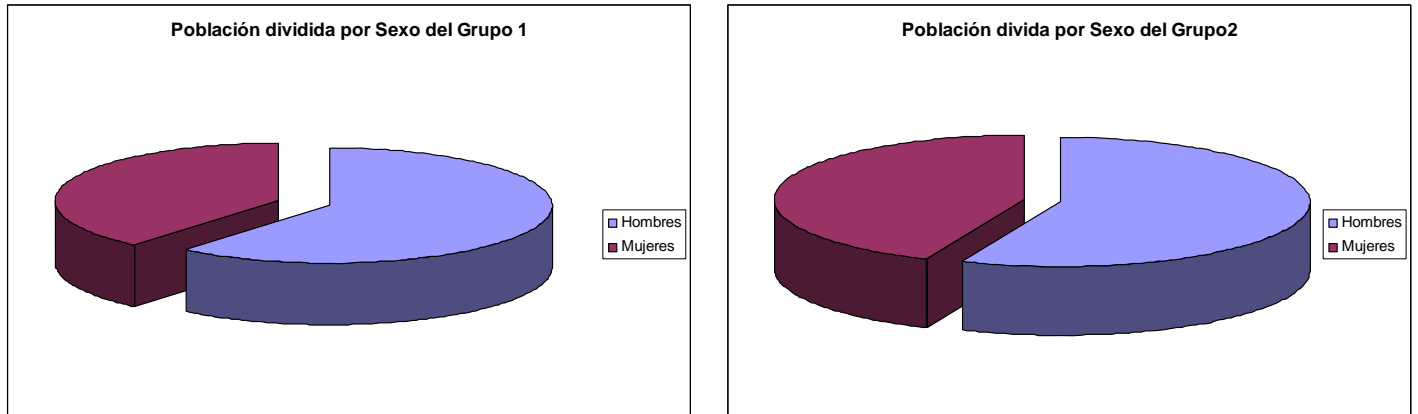


Figura 30: Gráficas con los porcentajes de las poblaciones divididas en grupos y por sexo.

Características:

	Bigotes	Barba	Orejas	Pelo Largo
Grupo 1	9	5	8	6
Grupo 2	16	5	5	4
Total	15	10	13	10

Tabla 2: Número de individuos por característica y por grupo.

Una vez descrito el corpus de trabajo, se continúa con el desarrollo del proyecto detallando las fases seguidas para la consecución de los resultados.

4.2. Tratamiento de la imagen

La primera etapa que se ha realizado es de tratamiento de la imagen y consiste en dos fases, una inicial de preprocesado de la imagen y una secundaria usada para la extracción de características.

4.2.1 Obtención de la imagen formato ppm a jpg

El primer proceso que se tuvo que realizar fue pasar las imágenes de un formato a otro, en concreto del formato de imagen *Portable Píxel Map* (.ppm) a *Joint Photographic Experts Group* (.jpg), formato más universal el cual no pierde propiedades de la imagen y sí se gana en ventajas de uso ya que la mayoría de los

visores de imágenes trabajan con este formato entre otros. Este cambio de formato se ha realizado con un simple script en Matlab recodificándose la imagen al formato deseado en destino, para nuestro caso, JPEG. Una vez, transformadas todas las imágenes al formato deseado, se ha continuado con el desarrollo del proyecto.

4.2.2 Preprocesado

Esta fase de preprocesado se ha realizado con el fin de tomar en cuenta sólo la información relevante eliminando así datos que no aportan información útil. Este fin se pensó inicialmente para la posterior selección de características, donde se pretende eliminar toda información que no tuviera que ver con las características típicas a extraer en un rostro humano. Con ello lo que se pretende es quedarse exclusivamente con el contorno de la cara, los contornos de zonas como el pelo, las características en sí, por ejemplo, los ojos, nariz, boca, orejas y cualquier otro rasgo que pudiese aportar información del individuo. Para realizar dicho preprocesado, se ha segmentado la imagen en dos etapas usando el algoritmo de agrupamiento *k-medias*. La primera etapa consta de dos centroides, para poder separar el fondo y el individuo en sí. Dado que las imágenes están tomadas sobre el mismo fondo azul claro, este hecho ha beneficiado esta primera segmentación pudiendo separar fácilmente el fondo de la imagen de la zona que contiene al individuo, para ello vemos el resultado de esta primera segmentación:

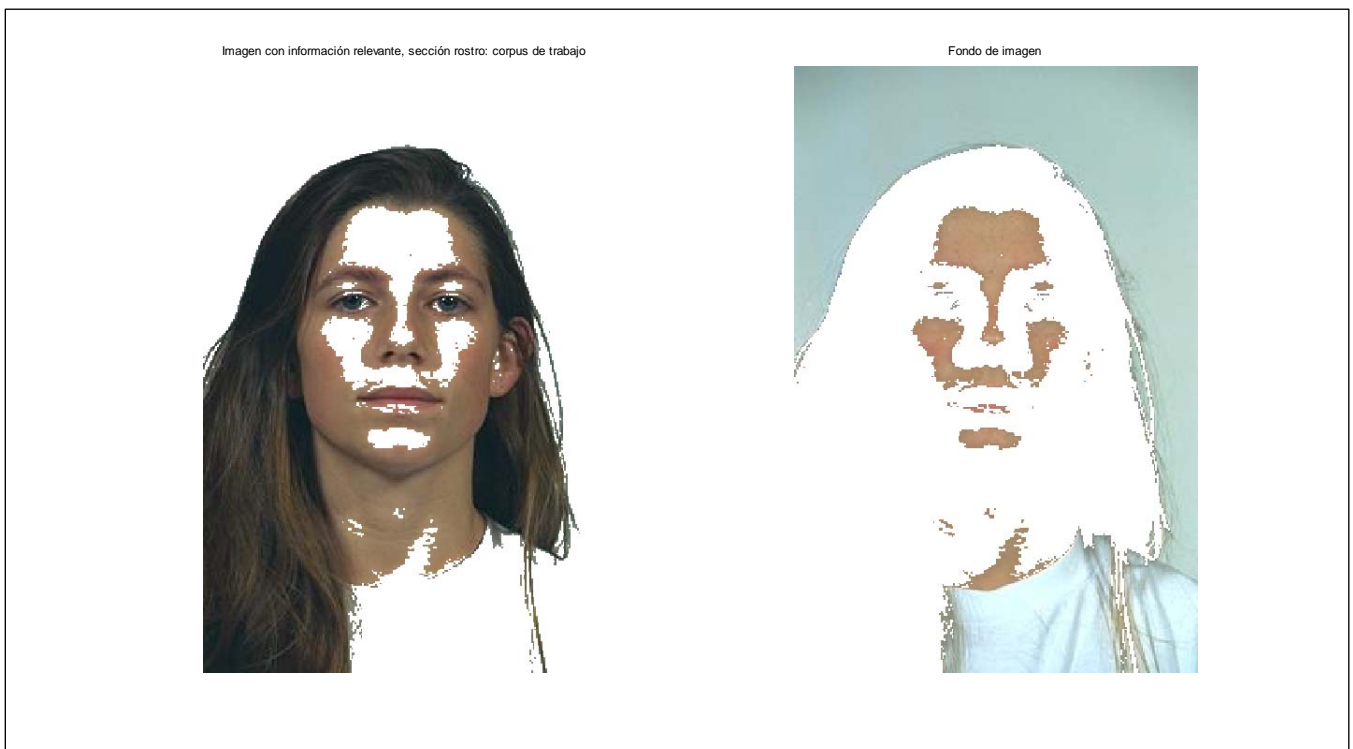


Figura 31: Resultados de la primera segmentación, a la izquierda la imagen con información relevante y a la derecha parte de la imagen que se considera el fondo con información irrelevante.

El propósito de la segunda segmentación ha sido separar por un lado las regiones y características de la imagen de las zonas que carecían, a nuestro parecer de información útil, zonas de la piel y la vestimenta. Esto se ha realizado usando de nuevo el mismo algoritmo de agrupamiento pero en esta ocasión usando tres centroides, uno para el fondo blanco sobre el cual se había vuelto a pegar la imagen útil, otro centroide que ganaba los píxeles que contienen la información menos útil y por último, el centroide que conservaba los píxeles más oscuros que coinciden con las características de la cara. Como se puede observar, este preprocesado se basa sobre todo en el color de la imagen, agrupándose los puntos más oscuros y guardándose esa imagen segmentada sobre la que posteriormente se van a seleccionar y extraer los puntos característicos. En la siguiente figura podemos observar las dos imágenes separadas, por un lado la imagen con las características y por otro lado, las zonas de la imagen descartadas para la posterior selección:

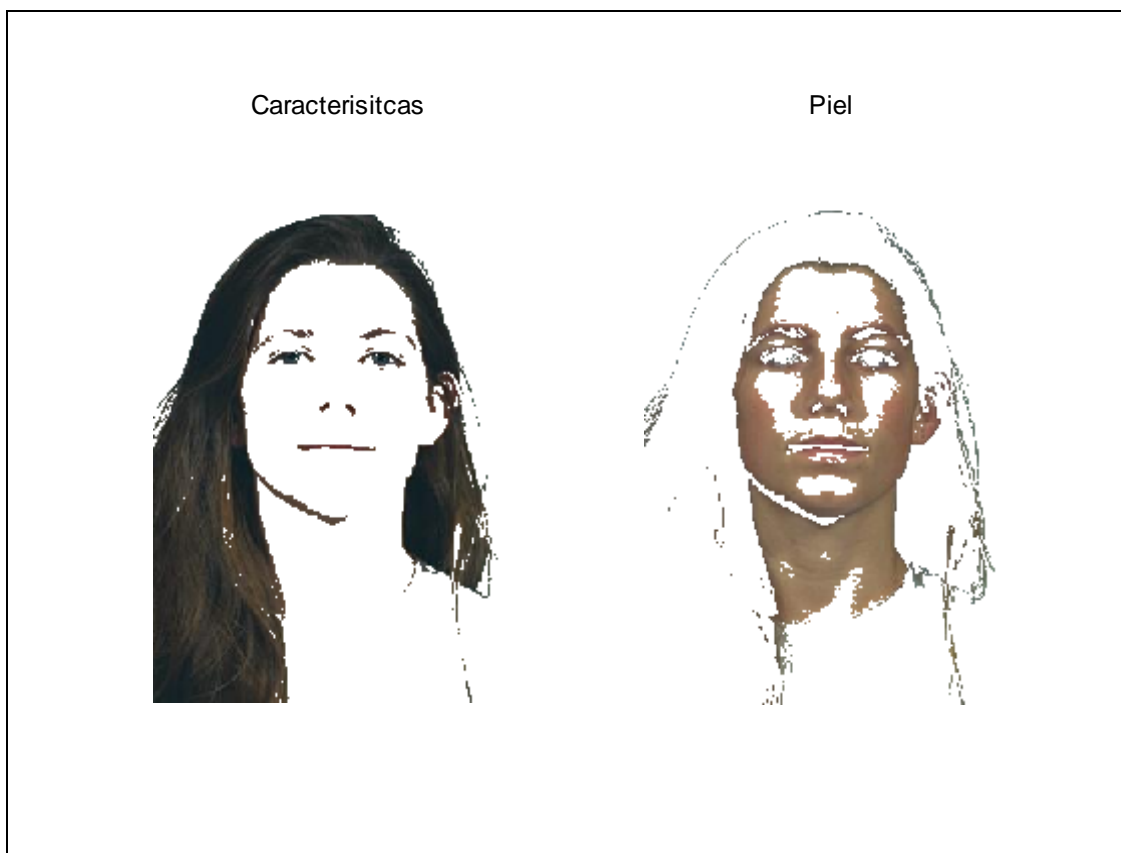


Figura 32: Imágenes resultantes tras la segunda segmentación donde se separan las características del rostro de la piel.

Este preprocesado también era necesario para eliminar cualquier tipo de ruido que pudieran tener las imágenes a pesar de que en su mayoría eran nítidas y correctas. Por último, también se ha probado el proyecto sin usar este tipo de preprocesado ya que al estar buscando representar la información de la imagen con el grafo, se consideró la posibilidad de que este preprocesado eliminara aspectos o información relevante que a priori pudiese parecer innecesaria, por lo que se ha intentado considerar todas las posibles opciones.

4.2.3 Extracción de Características

Uno de los aspectos clave para el desarrollo del proyecto es la selección y extracción de las características del rostro. Dicha extracción debe seleccionar aquellos puntos de una cara que contengan toda la información relevante de la misma, y a su vez, ser puntos invariantes, es decir, que no se vean afectados por cambios de color, iluminación o escalado. Para esta fase, se han buscado algoritmos en el campo del tratamiento de la imagen que tuvieran estas propiedades y finalmente se han barajado cuatro posibles algoritmos: Esquinas de Harris, Laplace de Gaussianas, método de Susan y algoritmo de Gilles. De estos algoritmos, algunos de ellos detectan esquinas y bordes lo que se ha considerado como positivo dado que estos puntos son más significativos y aportan más información que otro punto cualquiera de una imagen.

El objetivo de los puntos característicos que se obtienen con estos algoritmos es que cumplan las siguientes propiedades:

- i. Todas las esquinas y vértices se deben detectar.
- ii. No se deben detectar falsos puntos de interés.
- iii. Debe ser robusto frente al ruido.
- iv. La localización debe ser precisa.

A continuación se van a describir brevemente estos algoritmos:

- **Esquinas de Harris:**

Este algoritmo es un detector de esquinas y bordes, donde se define una esquina como la intersección de dos bordes o a su vez, como un punto donde existen dos direcciones de bordes diferentes. Estos puntos también se conocen como puntos de interés y deben estar siempre en una posición bien definida así como ser detectados de forma robusta. Con estas definiciones, un punto de interés engloba esquinas, puntos aislados de intensidad máxima o mínima, finales de líneas, o puntos donde la pendiente de una curva es máxima.

Este algoritmo se basa en el algoritmo de detección de esquinas de Moravec, el cual detecta esquinas mediante similitud de intensidad analizando los píxeles por bloques. Cuando una región presenta intensidad uniforme, estos bloques serán similares, mientras que cuando se analiza un píxel en un borde, los bloques en direcciones perpendiculares resultaran distintos. Con esto se obtiene la similitud entre bloques y el máximo local se considera un punto de interés. El algoritmo de Harris es una mejora del anterior, al usar el diferencial de la esquina con respecto directamente de la dirección y no usando un conjunto de parches. Con esto, un punto de interés se caracteriza por presentar grandes variaciones de intensidad en todas las direcciones. Dicho algoritmo obtiene buenos resultados para imágenes faciales, al detectar un conjunto bueno de puntos de interés y presentar un coste computacional no demasiado elevado.

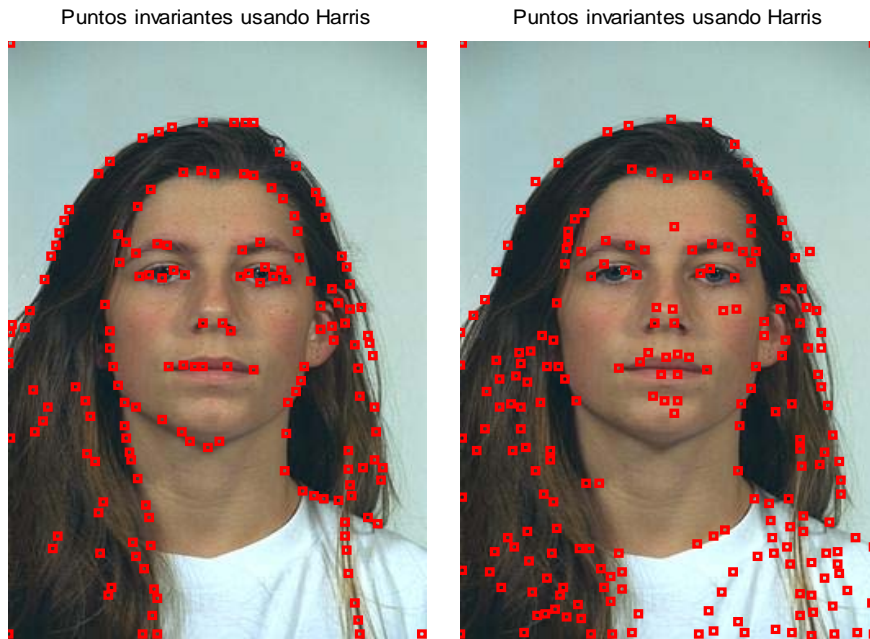


Figura 33: Puntos de interés obtenidos usando el algoritmo de Harris para la imagen preprocesada (drcha.) y con la imagen completa (izq.)

- **LoG (Laplaciana de Gaussianas):**

Este método es uno de los más comunes detectores y se basa en la Laplaciana de la gaussiana. Esto es, dada una imagen, se convoluciona con una gaussiana fijada una escala determinada y luego se computa la Laplaciana lo que conlleva a resultados positivos y negativos para “blobs” oscuros y claros, respectivamente. El principal problema de este algoritmo es la dependencia del escalado con el tamaño del “blob” detectado, pudiéndose obtener mejoras usando variantes automáticas para conseguir regiones multiescalado.

Los puntos resultantes se presentan a continuación donde podemos observar los puntos detectados usando el algoritmo LoG:

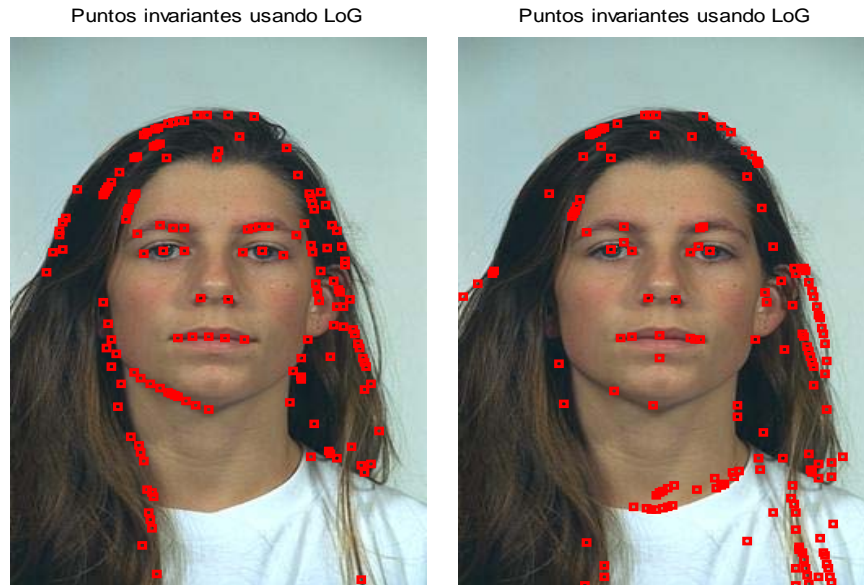


Figura 34: Puntos característicos usando el algoritmo LoG para la imagen preprocesada y para la imagen completa.

- **Método de SUSAN (Smallest Univalued Segment Assimilating Nucleus):**

Para la detección de características, este método emplaza una máscara circular sobre el píxel a analizar denominado, núcleo. Cada uno de los píxeles es comparado con el núcleo usando una función de comparación y mediante similitud, se agrupa a un segmento. Dicho algoritmo obtiene un conjunto de puntos muy limitado además de utilizar para su ejecución un tiempo elevado para la obtención de los puntos característicos.

- **Algoritmo de Gilles:**

Dicho algoritmo detecta puntos de una imagen que sean únicos denominados puntos salientes. Para ello usa medidas de rarezas maximizando la probabilidad de un objeto dada una medida con lo que obtiene puntos de mayor distinción. El problema de este algoritmo es que tiene una escala fija y es sensible al ruido, así como que produce un conjunto de puntos muy limitado para el propósito del proyecto. A continuación vemos los puntos resultantes:

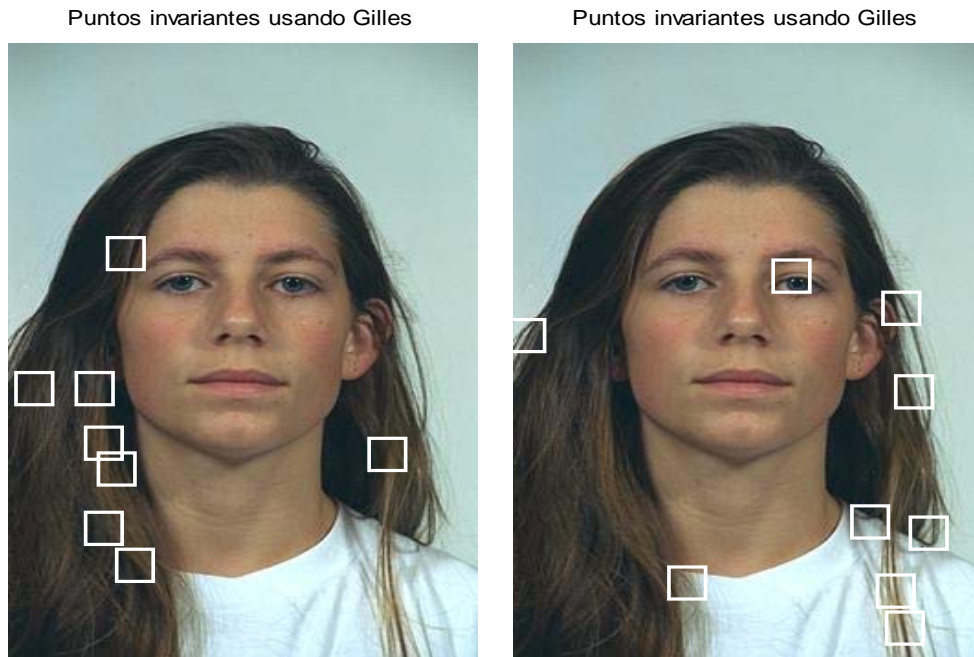


Figura 35: Puntos de interés obtenidos usando el algoritmo de Gilles para la imagen preprocesada (drcha.) y con la imagen completa (izq.)

A parte de todos estos algoritmos también se ha considerado la combinación de ellos consiguiendo así mayor información y más robustez dado que si alguno de los métodos no considera un punto como importante, es probable que otro sí lo tome en cuenta si verdaderamente lo es.

Una vez probados todos los métodos, se ha decidido descartar alguno de ellos en función de los resultados obtenidos y del tiempo de ejecución del mismo para optimizar el proceso total del proyecto. Los métodos descartados fueron el algoritmo de Gilles y de Susan. El primero debido a que obtenía un conjunto de puntos característicos demasiado reducido y el segundo debido al elevado coste computacional que conlleva un elevado tiempo de ejecución. De esta forma, se ha optado utilizar una combinación entre el método de Harris y Laplace de Gaussianas produciendo un conjunto grande de puntos que como se puede ver en las siguiente gráficas, están situadas en las características de un rostro que se estaban buscando.

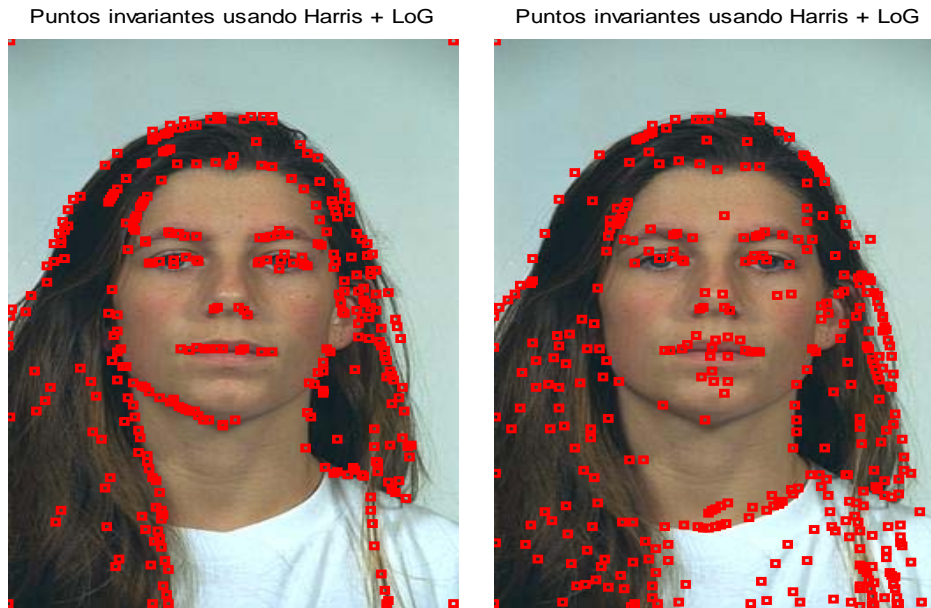


Figura 36: Selección final de los puntos característicos utilizados combinando los algoritmos de Harris y LoG. Comparativa en el uso de la imagen completa (dcha.) y la imagen preprocesada (izq.).

4.3 Tratamiento del Grafo

4.3.1 Definición de Palabras

Una de las claves de este proyecto es la definición de una palabra visual. Una *palabra visual* es el equivalente a una palabra de un texto en cuanto al significado que posee dentro de una imagen. Por tanto, esta característica visual debe poseer la semántica de lo que representa estando presente en todas las imágenes que posean esa misma característica visual. La forma de implementar *palabras visuales* es mediante un descriptor local (ver apartado 2.6).

Para ello debe cumplir unas propiedades imprescindibles para saber que se ha definido de forma correcta una palabra visual y estas son:

- **Invarianza:** Es una de las propiedad más importantes que debe cumplir una palabra visual. Esto se debe a que es necesario que se pueda identificar y extraer una *palabra* siempre que esté presente en la imagen. Para ello se han usado algoritmos de extracción de características invariantes (ver 4.2.3) así como la propia estructura usada para la definición como se explicará posteriormente.
- **Palabras Representativas:** Otra propiedad de interés es la importancia de definir palabras con fundamento, aquellas palabras que realmente aporten información de la imagen y sean representativas. Con esto se quiere asegurar, que no se trabaja con palabras que carecen de información. Al igual que en el lenguaje escrito, hay un conjunto finito de palabras en un diccionario, y esa propiedad se ha intentado simular evitando que el número de palabras crezca sin límite a medida que se analizan más imágenes.

- **Posean Información de la imagen:** Es condición necesaria que la definición de la palabra posea toda la información de la imagen que se pueda. Este punto supone encontrar un equilibrio entre la información almacenada y su ocupación dado que palabras muy pesadas, en cuanto a capacidad (número de dimensiones), influyen negativamente en el rendimiento del sistema total al ralentizar más el procesado.

Para cumplir estas propiedades, se ha optado por implementar un descriptor local con 3 dimensiones, un descriptor SIFT simplificado, que cumple con las propiedades anteriormente descritas. Para ello se ha usado la combinación de los algoritmos de extracción de características Harris y Laplace de Gaussianas, consiguiendo así una base invariante a la hora de seleccionar los puntos. Una vez obtenidos los puntos, se ha decidido usar conjuntos de tres puntos o esquinas, cuya distancia entre sí fuera inferior a un umbral. Dicho umbral se ha fijado posteriormente de forma práctica al comprobar que se cumplían también el resto de propiedades. Estas esquinas, o combinaciones de tres puntos invariantes, aseguran no estar seleccionando ruido, o puntos aislados que no aportan información haciendo la selección aún más invariante. A su vez, había que dotar a estas esquinas de más información por lo que se ha escogido tres conceptos de información visual:

- **Ángulo:** Valor del ángulo que se forma entre los tres puntos característicos. Dicho ángulo es siempre inferior a 180° , ya que se toma el ángulo complementario en caso de ser superior.
- **Longitud del perímetro:** Distancia total en píxeles entre los tres puntos.
- **Iluminación Media:** Valor entre 0-255 de la intensidad de iluminación del área que haya en el triángulo restante que forma la esquina.

Con estas tres dimensiones, se consigue identificar características de una imagen, que sean invariantes y se puedan obtener aunque estén situados en otro lugar e incluso si se trabaja a otra escala o con imágenes en blanco y negro.

Por último, estas *palabras* no poseen todo el contenido semántico sino que se incluye la semántica en la forma de relacionar las palabras entre sí y con las colecciones de imágenes previamente seleccionadas y categorizadas.

Comparativa de Palabras

A lo largo del proyecto se ha contemplado la necesidad de comparar las *palabras visuales* para obtener el grado de similitud o disimilitud entre ellas. Para ello, se ha usado la distancia de Mahalanobis fijando un umbral reducido que fijaría la igualdad entre palabras (dejando una tolerancia). De esta forma, se consigue descartar palabras repetidas ya que uno de los criterios de diseño ha sido no incluir bucles en el grafo.

$$Distancia = \sqrt{\left(\frac{\alpha_1 - \alpha_2}{\alpha_{TOTAL}}\right)^2 + \left(\frac{L_1 - L_2}{L_{TOTAL}}\right)^2 + \left(\frac{I_1 - I_2}{I_{TOTAL}}\right)^2}$$

Siendo: α_{TOTAL} la variación máxima 180°

L_{TOTAL} la variación máxima del perímetro fijada por el umbral usado.

I_{TOTAL} la variación máxima de intensidad de iluminación 255.

Se ha elegido esta distancia frente a la distancia Euclídea debido a que se normaliza por la varianza de cada componente de la palabra dotando del mismo peso a cada una.

4.3.2 Construcción del Grafo

Uno de los puntos clave para el desarrollo del proyecto es la correcta construcción del grafo. Para ello, se han probado diversas posibilidades para cada uno de los casos.

4.3.2.1) CASO 1: Construcción directa del grafo a partir de los puntos.

Para este primer intento se probaron diversas relaciones de forma experimental para finalmente quedarnos con una opción que cumplía los requisitos tanto de forma como de resultados del grafo. Para comenzar, se van a mencionar las diferentes construcciones probadas razonando el porqué de su elección.

El primer punto a tener en cuenta es que el grafo con el que se va a trabajar, tiene que ser un grafo que utilice los puntos característicos obtenidos con anterioridad y que se relacionen entre ellos de alguna forma. Para ello, nos hemos apoyado en el artículo "*Efficient Graph-Based Image Segmentation*" [53], el cual propone unir los puntos que se encuentren a una distancia inferior a un umbral y que a su vez, posea un peso dicha unión. Se ha considerado esta forma de unir los puntos ya que el principio buscado es plasmar la información de la imagen en el grafo. De esta forma, lo que se pretende conseguir es que al tener puntos característicos que se encuentran próximos entre sí, es muy probable que estos puntos pertenezcan a la misma característica del rostro, y por tanto es lógico que posean una relación entre ellos. De esta forma se les está dotando de significado a las relaciones entre los puntos, hecho que nos ayudará a su posterior interpretación y clasificación. Por otro lado, los puntos que se encuentren distantes, no estarán relacionados lo que a su vez también parece normal, ya que, como ejemplo, no interesa que la oreja izquierda está relacionada con el bigote. Este umbral de distancia para relacionarlos, es un parámetro importante ya que su manipulación deriva resultados distintos, pasando de grafos con una alta relación internodal a grafos en los que la mayoría de los nodos están monorelacionados. Para ello, se ha experimentado con 3 casos de distancias para los umbrales, 100, 400 y 1000 siendo las unidades de distancias, el píxel. Esto se ha realizado por distancia geométrica alrededor del punto característico analizado, eso es, estando centrando en dicho punto, a cualquier

otro nodo que se encuentra a una distancia inferior al umbral, se le conecta un enlace que denota una relación de proximidad y por tanto de “característica”.

En las figuras siguientes se puede apreciar el ejemplo de los puntos relacionados entre sí, que denotan una misma característica:



Figura 37: Imágenes representando las agrupaciones de nodos de una misma característica, bigote y orejas prominentes.

Por otro lado, se han impuesto unos requisitos para la construcción del grafo ya que nos hemos apoyado en el artículo de A. Barabasi [50], donde expone que los grafos deben ser grafos de orden libre para su posterior análisis y para que las propiedades que poseen este tipo de grafos sean las deseadas. A su vez, se ha considerado que el grafo sea totalmente conexo y que no haya ningún nodo o conjunto de nodos sueltos. Para ello, se ha procedido a una primera construcción uniendo cada grafo al nodo más cercano y de esta forma se consigue un grafo conexo. El segundo proceso, es relacionar los nodos que se encuentren a una distancia inferior al umbral. Una vez realizado esto, se comprueba si el grafo es de orden libre o es un grafo aleatorio. Según el artículo de Barabasi, existe una forma rápida de verificar el tipo de grafo construido a partir de la relación entre el número de nodos y los enlaces de esos nodos. Para el caso de un grafo aleatorio, la forma de esta relación es una campana de Gauss, ya que la media de los nodos tiene varios enlaces y sólo existen grupos reducidos de nodos con muchos o muy pocos enlaces, formándose así esa campana gaussiana. Por el contrario, un grafo de orden libre significa que la mayoría de los nodos están poco relacionados mientras que un conjunto reducido de nodos están muy relacionados. Este tipo de grafos presentan la forma de exponencial inversa para el caso de relacionar el número de nodos con sus relaciones.

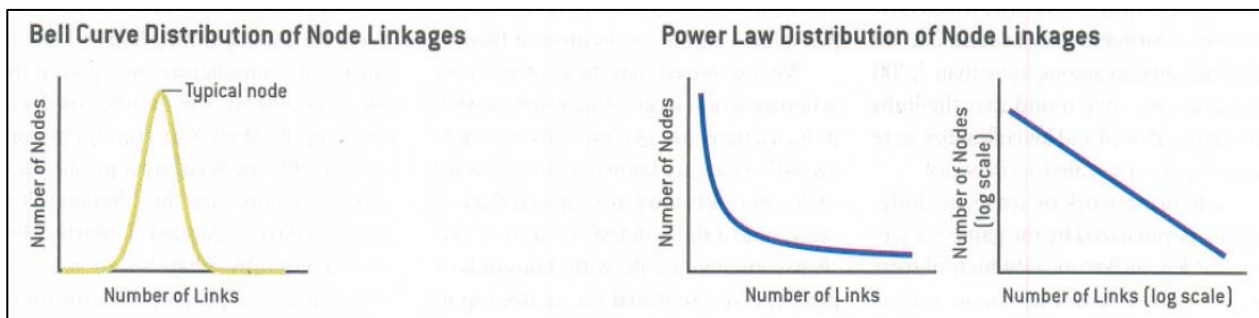


Figura 38: Distribuciones de grafos de orden libre y grafos aleatorios. (A.Barabasi [50])

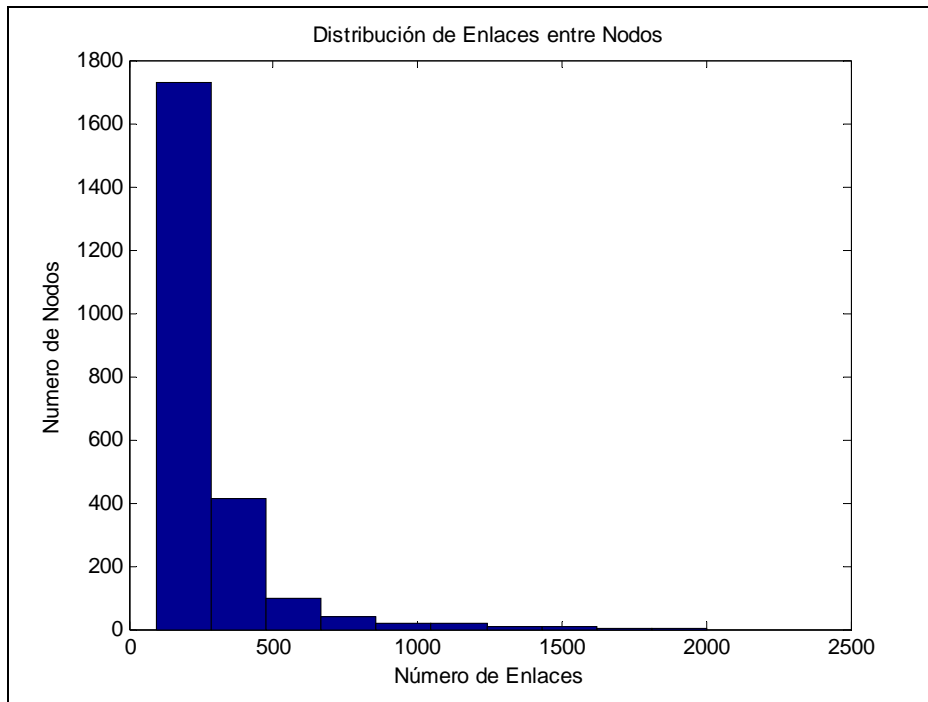


Figura 39: Distribución de enlaces entre nodos para el grafo usado que cumple la forma de exponencial inversa, por tanto es un grafo de orden libre.

Una vez vistas las distintas opciones, se realizó una plataforma de pruebas para ver la construcción definitiva del grafo. Para ello, se combinaron los distintos umbrales, inicialmente usándolos de forma independiente y posteriormente usando superposición por jerarquías o niveles. Es decir, el grafo se ha construyó a partir de su matriz de adyacencia donde los puntos característicos formaban las entradas de sus filas y columnas y si existe una relación entre los puntos se marcaba con el número de enlaces, es decir, un enlace obteniéndose la siguiente estructura general:

	Punto 1	Punto 2	...	Punto N
Punto 1	0	1	...	1
Punto 2	1	0	...	0
...
Punto N	1	0	...	0

Figura 40: Matriz de Adyacencia de un grafo genérico.

Como se puede observar de la matriz, la diagonal de dicha matriz debe ser ceros ya que se ha descartado el caso de que existan bucles en un mismo nodo, y a su vez, la matriz es simétrica al no poseer dirección los enlaces. Para nuestro caso, se ha utilizado la matriz de adyacencia como la forma de representar nuestro grafo, aunque en un reciente estudio de P.Zhu y R. Wilson [52], comparan los resultados de espectros

usando otras alternativas como la matriz Laplaciana, dejando esta comparativa abierta para futuras líneas de investigación.

4.3.2.2) CASO 2: *Palabras visuales* para toda la imagen.

La segunda aproximación ha consistido en construir un grafo con un conjunto de imágenes previamente clasificadas por tipo (clases: hombres, mujeres, por razas,..). De esta forma, se pretende dotar de semántica a las palabras que aparecen para un mismo tipo de imagen, y por tanto entre ellas deben estar relacionadas. Las *palabras* que describen una imagen del mismo tipo, deben ser parecidas entre sí, o aparecer con mayor frecuencia, y por tanto se relacionan.

Esta metodología, ha partido del uso de las *palabras visuales* como nodos, estando todas las palabras de una misma imagen relacionadas entre sí. Posteriormente, al introducir una nueva imagen, se relacionan las palabras que ya existan en el grafo con las nuevas que se introducen. De esta forma se pretende obtener un grafo diferente para cada una de las clases de imágenes que procesemos.

La forma de extraer el grafo de la clase, es de forma iterativa procesando cada imagen de forma individual y posteriormente se combina con el grafo resultante relacionando aquellas *palabras* que previamente ya existan en el grafo total.

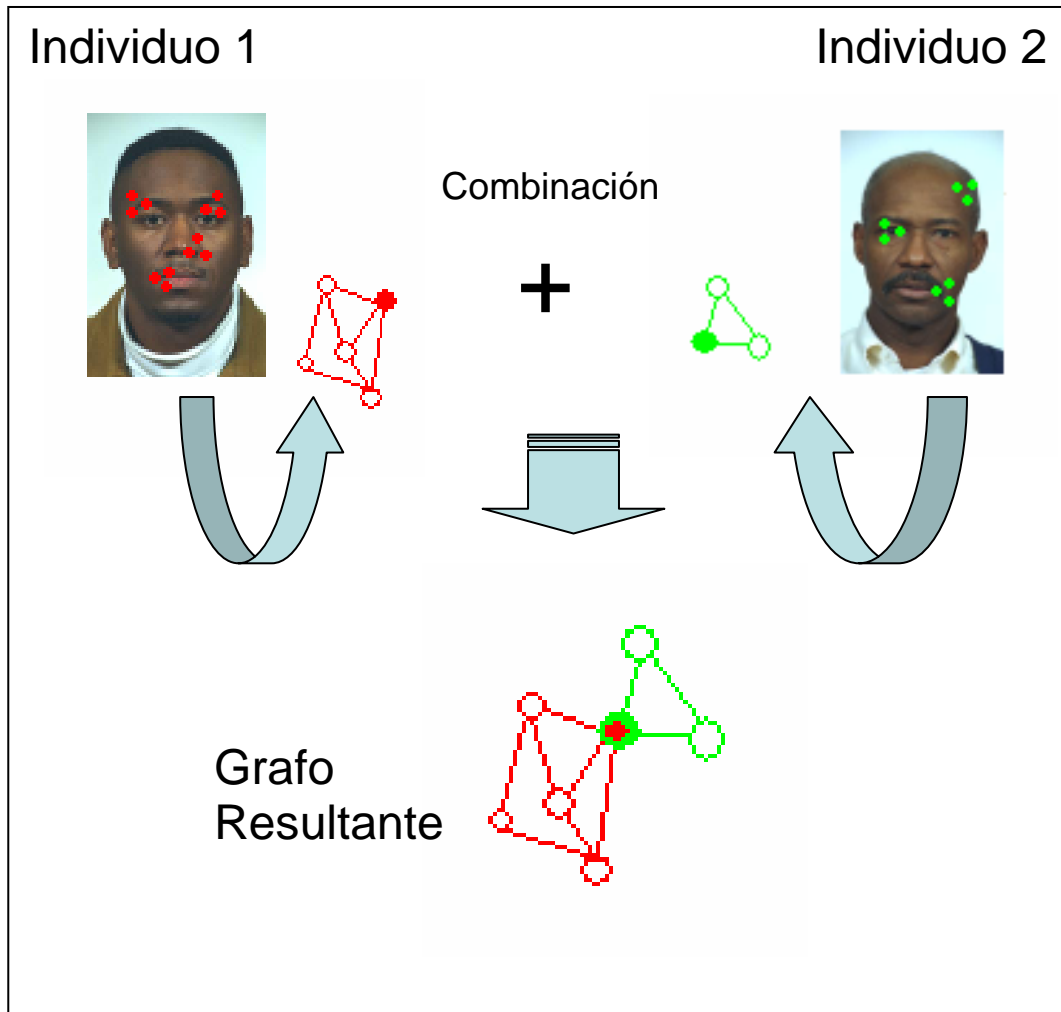


Figura 41: Ejemplo gráfico de la combinación de grafos de diferentes individuos para formar el grafo total por clase.

El grafo obtenido cumple las condiciones de A.Barabasi [50] obteniéndose un grafo de orden libre, donde las *palabras* nuevas introducidas decrecen en número al insertarse una nueva imagen, demostrando la relevancia de las palabras utilizadas.

4.3.2.3) CASO 3: Segmentación de la imagen en Celdas

Por último, se ha implementado una tercera metodología que varía significativamente la forma de construcción del grafo. Este método pretende equiparar una imagen a un texto corto, por ejemplo un párrafo, segmentando una imagen en celdas, las cuales equivaldrían a las frases, y posteriormente se extraen las *palabras visuales* cuyos equivalentes textuales son, evidentemente, las palabras. De esta forma, se consigue relacionar las palabras visuales de una misma celda entre sí, y luego aquellas que se repitan en otras celdas serán el nodo de unión para formar el grafo de la imagen completa.

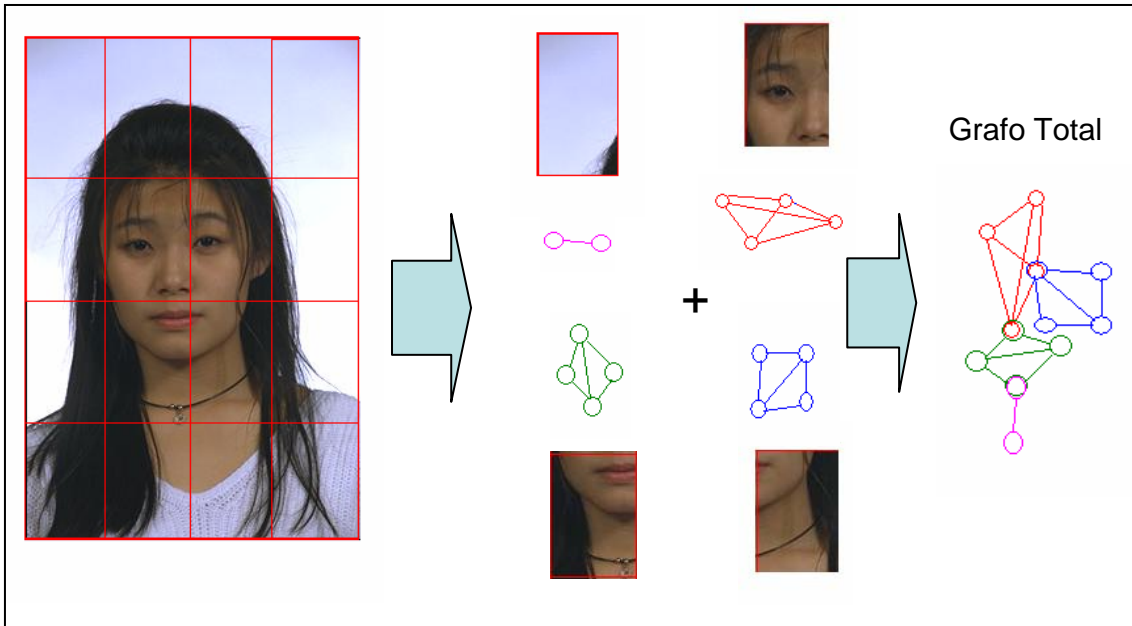


Figura 42: Ejemplo gráfico del proceso empleado en la construcción del grafo al segmentar una imagen en celdas.

De esta forma, se obtiene una estructura de grafo similar a la obtenida por en un texto si se relacionasen las palabras de una misma frase entre sí, y luego relacionar aquellas del mismo párrafo. [J.Santaella, 1].

4.3.3 Espectro de un Grafo

El espectro del grafo es la herramienta fundamental para la obtención de resultados en este proyecto. En él, se van a basar los resultados al usar esta forma de representación como medida de similitud o disimilitud entre los grafos analizados.

Para ello, y como se definió en el apartado 3.1, el espectro es el conjunto de los autovalores del grafo. Estos autovalores representan información de dicho grafo y por tanto se pueden analizar de varias formas para extraer su contenido:

- i. *Autovalores Máximos:* El primer autovalor es aquel que posee mayor información del grafo.
- ii. *Autovalores Extremos:* Estos autovalores son el máximo y mínimo de todo el conjunto de autovalores. En ellos también se puede extraer información relevante del grafo.
- iii. *Espectro completo:* El uso del conjunto completo de los autovalores, contiene toda la información sin truncar o limitar su contenido.

De estas formas, y basándonos en estudios previos, J. Santaella [1], se ha optado por usar el espectro completo para considerar toda la información del grafo y evitar así interpretaciones erróneas donde no se haya considerado toda la información relevante.

Por otro lado, una vez establecido como criterio el uso del espectro completo, se ha precisado de varias normalizaciones para poder equiparar los espectros para su posterior comparación. Esto se debe al hecho que los espectros no tienen el mismo número de puntos, ya que son directamente proporcionales al número de nodos, ni abarcan las mismas dimensiones en cuanto al valor de los autovalores. Para poder equipararlos y posteriormente compararlos, se ha procedido a varias normalizaciones evitándose en todo momento alterar la información del espectro.

- a) Se ha realizado una normalización del espectro para tener área unidad de forma análoga al área de una función de densidad de probabilidad. Así la comparativa se puede hacer de forma directa por diferenciación al estar diferenciando porcentajes del espectro en vez de valores concretos.
- b) Se ha efectuado una normalización del eje de las abscisas para obtener el mismo eje absoluto (dividiendo por el autovalor máximo obteniendo así un intervalo $[-\infty, 1]$). De esta forma tenemos el mismo eje relativo para todos los espectros.
- c) Se ha realizado una interpolación por splines cúbicos en el eje de las abscisas para obtener el mismo número de puntos para todos los espectros. Para minimizar costes computacionales, se han truncado los extremos para evitar la interferencia de “outliers” en el resultado final, asumiendo el coste, presumiblemente ínfimo, de este truncamiento de información.

A continuación se muestran unos ejemplos de los espectros generados en este proyecto, viéndose así su forma y valores del mismo.

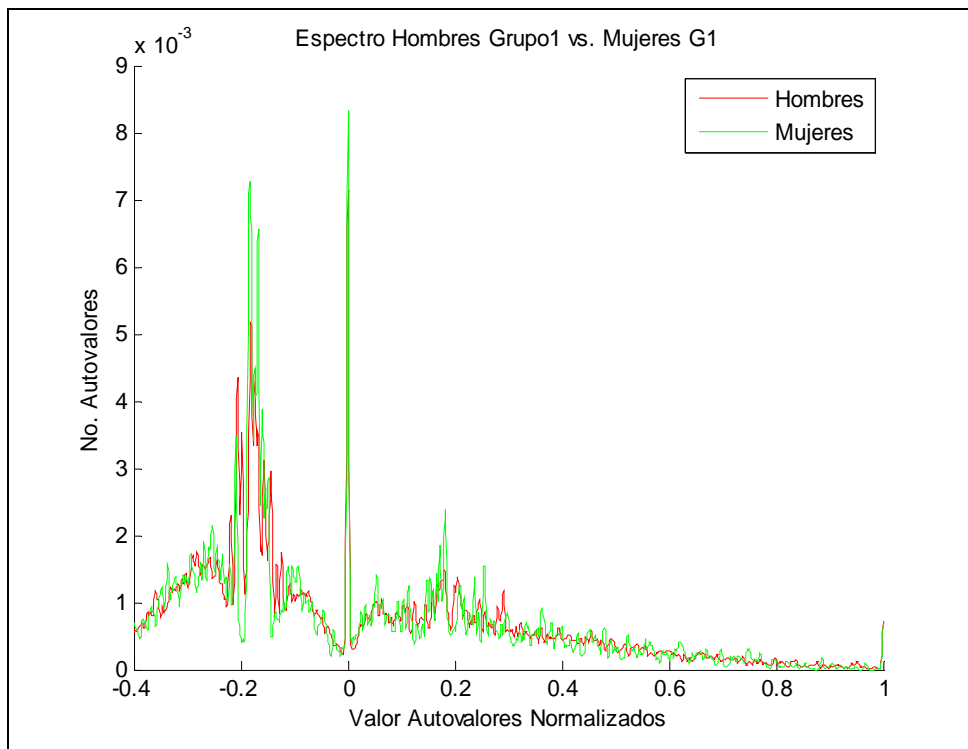
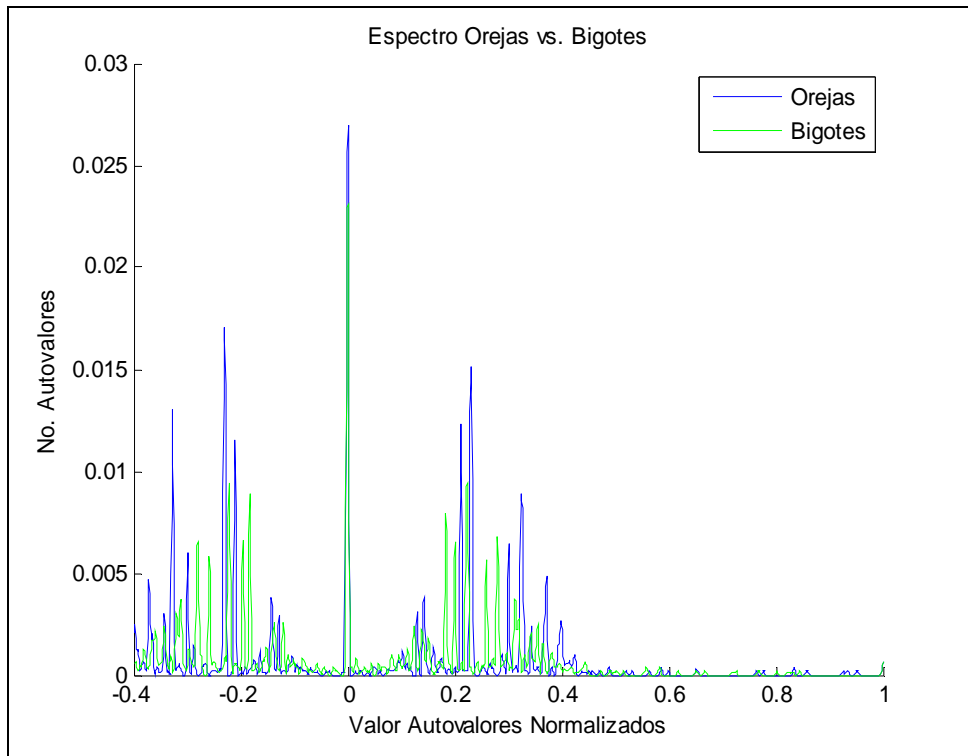


Figura 43: Ejemplos de distintos casos analizados de los espectros.

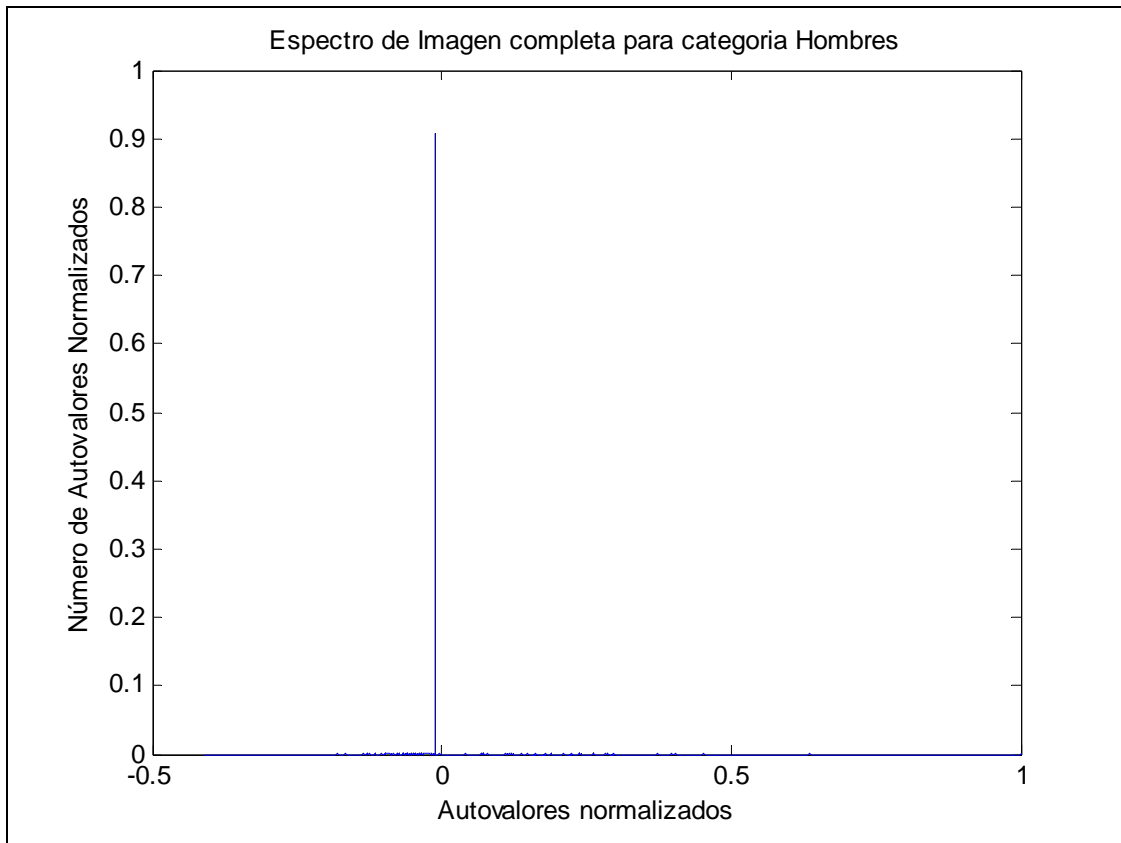


Figura 44: Ejemplo del espectro del tercer tipo de caso analizado.

4.3.4 Comparativa de espectros

A la hora de comparar dos espectros y medir su grado de similitud o disimilitud, se han usado dos medidas de distancia. Inicialmente se ha usado una diferenciación directa y posteriormente se ha utilizado la distancia de Hellinger que aproxima la divergencia de Kullback-Leibler (KL).

La primera medida de distancia entre espectros se ha realizado por diferenciación en cada intervalo dado que todos los espectros tienen información en los mismos puntos del eje de las abscisas al haber interpolado para conseguir el mismo número de puntos. A su vez, al tener área unidad la diferencia entre ambas nos da el porcentaje de disimilitud entre cada espectro. De esta forma, se puede distinguir cuando un espectro perteneciente a una categoría o clase se asimila más o menos a otro espectro.

La segunda distancia utilizada es la distancia de Hellinger, la cual aproxima la divergencia de Kullback-Leibler entre dos distribuciones obteniéndose la distancia con la siguiente expresión analítica:

$$d(p, q) = 1 - \int_{\Omega} \sqrt{p(x) \cdot q(x)} dx$$

Siendo: $p(x)$ y $q(x)$ cada uno de los espectros normalizados con área unitaria.

5. RESULTADOS

En este apartado se presentarán los resultados de varios escenarios de prueba para poder sacar conclusiones a partir de ellos.

CASO 1: Construcción directa del grafo a partir de los puntos característicos

Prueba I. Diferencias entre características

Objetivo: Observar la existencia de similitudes o diferencias entre 4 subcategorías de características en los rostros humanos (barba, bigote, pelo largo y orejas destacadas).

Con Preprocesado: Las diferencias entre características de ambos grupos son distintas. Existen casos iguales, en ambos grupos son más parecidas las imágenes “orejas” con aquellas con “bigotes”, pero el resto varía. Se podría esperar que las características “barba” y “bigote” fueran más similares ya que una subcategoría engloba a la otra, pero esto sólo ocurre para el Grupo2.

G1	Orejas(%)	Pelo Largo(%)	Bigote(%)	Barba(%)
Orejas(%)	0	30.95	24.63	26.27
Pelo Largo(%)	30.95	0	28.10	32.85
Bigote(%)	24.63	28.10	0	30.01
Barbas(%)	26.27	32.85	30.01	0

G2	Orejas(%)	Pelo Largo(%)	Bigote(%)	Barba(%)
Orejas(%)	0	32.86	30.51	33.80
Pelo Largo(%)	32.86	0	37.30	33.11
Bigote(%)	30.51	37.30	0	30.27
Barbas(%)	33.80	33.11	30.27	0

Tabla 3: Resultados de la comparativa entre características en un mismo grupo usando etapa de preprocesado. (En rojo los valores mínimos que presentan la mayor similitud entre ellas). *Nota:* Resultados expresados en % de disimilitud/diferencia entre espectros.

Imagen Completa: Sucede lo mismo que para el caso anterior, pero con diferencias mayores (lógico al haber eliminado información supuestamente irrelevante con el preprocesado). En el G2, nuevamente ocurre el hecho esperado de que los conjuntos “barba” y “bigotes” se asemejan.

G1	Orejas(%)	Pelo Largo(%)	Bigote (%)	Barba(%)
Orejas(%)	0	34.80	33.21	29.12
Pelo Largo	34.80	0	28.87	35.50
Bigote(%)	33.21	28.87	0	41.44
Barbas(%)	29.12	35.50	41.44	0

G2	Orejas(%)	Pelo Largo(%)	Bigote (%)	Barba(%)
Orejas(%)	0	40.09	34.66	32.97
Pelo Largo(%)	40.09	0	39	37.08
Bigote(%)	34.66	39	0	27.09
Barbas(%)	32.97	37.08	27.09	0

Tabla 4: Resultados de la comparativa entre características en un mismo grupo para imágenes completas. (En rojo los valores mínimos que presentan la mayor similitud entre ellas).

La comparación entre características sólo sirve para tener una idea de si realmente estamos representando información de las imágenes. Parece ser que sí, por el hecho de que la barba y bigote son más similares. Tras los resultados obtenidos, como el hecho de que la barba y el bigote sean más parecidos, parece ser que la información obtenida de las imágenes es ciertamente útil. El resto de los resultados parecen aleatorios lo que puede explicarse con el hecho de que a priori no debe haber ningún tipo de relación o vínculo entre las características de un rostro a simple vista. Esto hace que una imagen clasificada como "pelo largo" pueda parecerse más a otra con "barba" por el simple hecho de tener más puntos característicos distribuidos por todo el área de la imagen.

Prueba II. Comparación entre Grupos (G1 vs. G2)

Objetivo: Comprobar que la diferencia entre imágenes clasificadas dentro de una misma categoría es menor que la existente entre imágenes de categorías distintas.

Preprocesado: Para esta prueba, se cumple que en dos de los cuatro casos, las mismas categorías entre para ambos grupos son más similares, siendo uno de los casos erróneos, concretamente el de las categorías "bigote/barba". El motivo del fallo puede deberse a lo parecido de las dos categorías, sin embargo al comparar con las otras categorías sí se obtienen los resultados deseados. Por tanto, podría decirse que el único resultado erróneo se daría para el caso del "pelo largo" puesto que se da la mayor disimilitud para la comparación directa. Esto se puede deber al hecho de que con el pelo largo los puntos característicos se distribuyen esparcidos por toda la imagen y al usar un umbral bajo (distancias cercanas entre puntos) para la creación de enlaces, no se ve reflejada del todo bien en el espectro esta característica.

G1\G2	Orejas(%)	Pelo Largo(%)	Bigote (%)	Barba(%)
Orejas(%)	28.03	28.99	31.08	29.10
Pelo Largo(%)	28.64	33.86	27.08	30.31
Bigote(%)	32.55	36.21	30.75	29.39
Barbas(%)	33.86	28.88	31.07	27.52

Tabla 5: Resultados de la comparativa entre características entre grupos distintos usando preprocesado. (En rojo los valores mínimos que presentan la mayor similitud entre ellas, en verde los campos que debían ser mínimos).*Nota: Resultados expresados en % de disimilitud/diferencia entre espectros.*

Imagen Completa: Para la imagen total, se puede observar que sin preprocesado los resultados son muy pobres. Esto es debido a que por el criterio de decisión empleado para formar el grafo, se enlazan los puntos característicos que se encuentran a una distancia inferior al umbral y al no preprocesar la imagen, dichos puntos se ubican de forma más espaciada en vez de concentrarse en las características específicas de la imagen. Debido a esto los valores resultantes son mucho más elevados y en ninguno de los 4 casos se obtienen los resultados deseados, los cuales sí se obtienen cuando se realiza el preprocesado de la imagen.

G1\G2	Orejas(%)	Pelo Largo(%)	Bigote (%)	Barba(%)
Orejas(%)	35.16	38.66	34.48	32.11
Pelo Largo(%)	31.05	43.46	34.28	36.61
Bigote(%)	33.88	40.06	34.86	38.54
Barbas(%)	34.78	45.44	43.60	43.28

Tabla 6: Resultados de la comparativa entre características entre grupos distintos usando imagen completa. (En rojo los valores mínimos que presentan la mayor similitud entre ellas, en verde los campos que debían ser mínimos).

Prueba III. Diferencias entre géneros

Objetivo: Siguiendo el planteamiento anterior, se desea comprobar si se puede establecer una distinción entre imágenes de géneros distintos.

Preprocesado: Para este análisis, se ha utilizado un número mucho más elevado de individuos para extraer las medias y en vista de los resultados, se puede apreciar que las diferencias totales disminuyen. A su vez, se han obtenido los resultados esperados en la investigación puesto que se ha conseguido una diferencia menor entre las mismas categorías que entre las categorías cruzadas. Con esto se consigue demostrar, que las características extraídas en los hombres se parecen entre sí, al igual que las características de las mujeres también son más parecidas dentro de su mismo grupo en comparación con el grupo contrario. Con estos resultados se puede extraer como

conclusión que los espectros representan información de los grafos y siempre que estos grafos sean construidos de una forma específica para representar unas características, una idea o concepto, se puede extraer dicha idea de ellos.

G1\G2	Hombres(%)	Mujeres(%)
Hombres(%)	12.63	14.55
Mujeres(%)	16.30	15.96

Tabla 7: Resultados comparativa entre géneros de distintos grupos. (En verde valores mínimos que representan mayor similitud). *Nota: Resultados expresados en % de disimilitud/diferencia entre espectros.*

Imágenes Completas: Para este caso, ocurre lo mismo que en la comparativa de las características individuales, es decir, que no se puede extraer información sin el preprocesado de la imagen. Esto se debe a que en el preprocesado se están enfatizando zonas de los rostros donde se encuentran las características típicas y relevantes y este hecho es importante para la construcción del grafo y más concretamente para la construcción de las relaciones internodales del mismo.

G1\G2	Hombres(%)	Mujeres(%)
Hombres(%)	17.60	20.33
Mujeres(%)	15.26	22.30

Tabla 8: Resultados comparativa entre géneros de distintos grupos. (En rojo valores mínimos que representan mayor similitud, en verde los campos que debían ser mínimos).

Prueba IV. Mismas comparativas con grafos aleatorios

Objetivo: Realizar las mismas comprobaciones de características y de distinción de género usando grafos aleatorios, esto es, incrementando el umbral de relación por encima de 400 píxeles, lo que conlleva un incremento en el número de relaciones entre nodos.

Resultados: Los resultados obtenidos indican que el uso de grafos aleatorios, conlleve peores resultados que los obtenidos usando grafos de orden libre. Esto se debe principalmente a la pérdida de información al construir el grafo, ya que deja de tener efecto el hecho de relacionar puntos de interés cercanos entre sí, perdiendo el significado de una misma característica del rostro (por ejemplo, nariz, oreja, bigote...), la cual sí esta presente en la otra construcción del grafo.

Prueba V. Variaciones del umbral

La variación del umbral conlleva el cambio del tipo de grafo. Esta variación hace cambiar el tipo de grafo de “orden libre”, para umbrales pequeños, distancias entre puntos característicos inferior a 15 píxeles, a grafo “aleatorio”, donde se considera que puntos que distan bastante pueden estar relacionados entre sí.

Los resultados de las pruebas nos llevan a concluir que la distinción de características es mejor usando grafos de orden libre, y por tanto con umbrales bajos, que grafos aleatorios.

Prueba VI. Combinaciones de los algoritmos de extracción de características.

A la vista de la explicación teórica, comentada anteriormente, los distintos algoritmos de extracción de características proporcionan puntos de interés diferentes, y por tanto una mayor información puntual de la imagen. Por ello, y como es evidente, el uso combinado de los algoritmos proporciona mejores resultados. La limitación a esta combinación es la sobreinformación, cuando varios algoritmos producen la misma salida de puntos de interés, así como el mayor coste computacional al incrementar el número de algoritmos implicados en el proceso.

Prueba VII. Comparativas individuales en lugar de por grupos.

Objetivo: Comprobar que con el uso de espectros de imágenes de forma individual en vez de grupal se obtienen peores resultados. Comprobar, por otra parte, a partir de que número de individuos, se estabiliza el espectro.

Resultados: Los resultados obtenidos con el uso de espectros de forma individual fueron peores y en cierta medida aleatorios. Esto se debe al hecho de usar la información de un sólo individuo lo que implica información, que al ser excesivamente particular, impide generalizar. No obstante, al obtener el espectro medio de varios individuos previamente clasificados, se refuerza la información parecida entre ellos lo que hace que aumenten las diferencias entre individuos distintos. Esta información se ha supuesto estable a partir de 5 individuos, optimizándose a medida que se incrementa el número de uno a cinco. A su vez, se han usado medias de grupos de más de 50, pero sólo para casos más generales como ha sido la comparativa por género, no habiéndose podido usar para los casos de características concretas debido a la limitación de individuos con esas características. Además, para el uso de este último caso, la inclusión de muchos miembros para hacer los espectros medios, incluye mucha información irrelevante lo que supone ruido y una consecuente alteración del espectro.

CASO 2: Construcción del grafo usando palabras visuales para toda la imagen

Prueba VIII. Comprobación definición de palabras

Objetivo: Comprobar que la forma de definir las palabras visuales obtenga un conjunto representativo de palabras.

Resultados: La definición de palabras visuales usando esquinas de 3 dimensiones basadas en puntos característicos obtiene un conjunto representativo de palabras que se repite a lo largo de las distintas imágenes pero decreciendo en número. De esta forma se demuestra que se consigue un vocabulario finito de palabras que decrece a medida que se introducen nuevas imágenes.

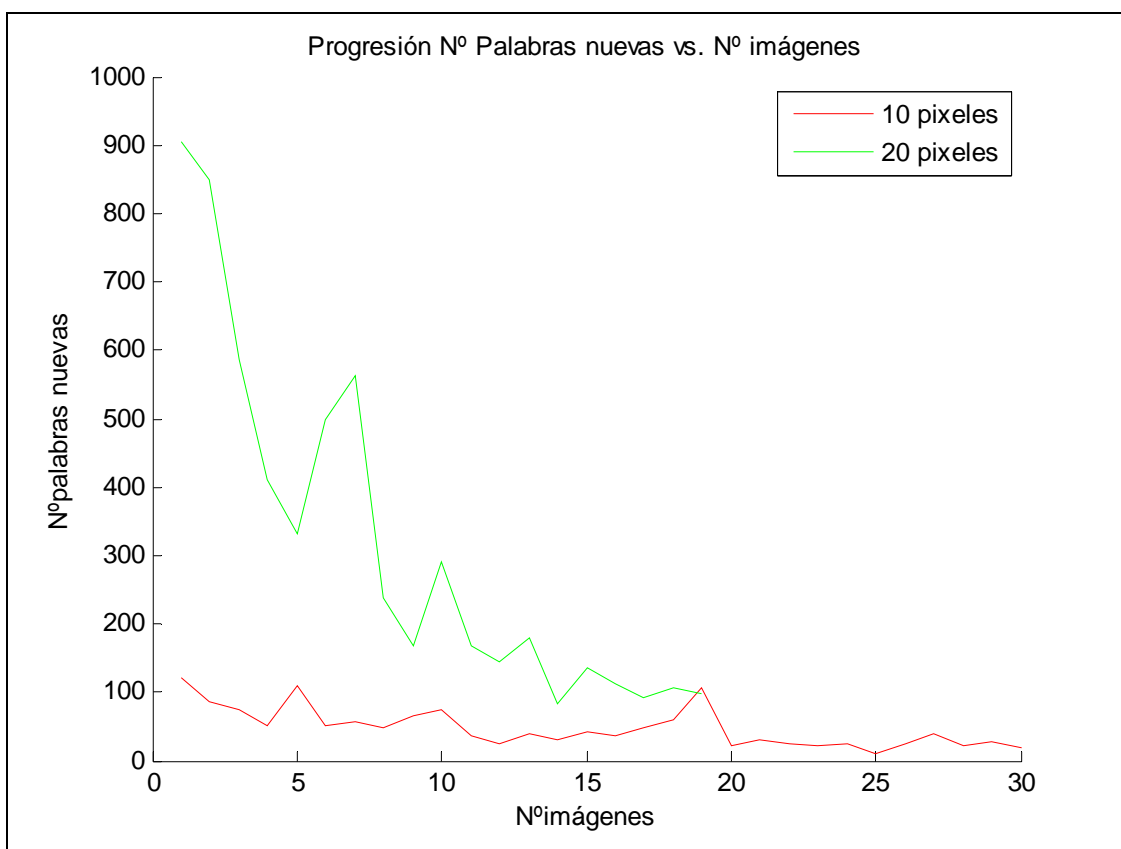


Figura 45: Representación de la evolución del número de palabras nuevas a medida que se procesan imágenes. Se ha probado para dos umbrales de distancia entre nodos que forman una esquina.

Estos resultados se han obtenido con la definición estricta de igualdad, la cual considera una palabra repetida cuando todas sus componentes son iguales, y para dos umbrales distintos de distancia entre los puntos que forman la esquina. Como se puede observar la progresión es decreciente para ambos umbrales, siendo más evidente con un umbral superior al considerarse inicialmente mayor número de palabras por imagen.

Prueba IX. Palabras visuales

Objetivo: Probar las distintas posibilidades de palabras definidas comprobando su validez.

Resultados: Se han probado 2 tipos distintos de *palabras visuales* aumentando la complejidad del descriptor local que las define.

- **Puntos Característicos:** Definiendo una palabra como un punto característico invariante obtenido con la combinación de los métodos de Harris y Logaritmo de Gaussiana, se ha comprobado que se obtiene un gran número de palabras pero que aportan poca información por sí solas. Con esta definición desconocemos las propiedades del punto extraído por lo que se busca una nueva definición que almacene información del punto característico.
- **Esquinas:** Definiendo una palabra visual como un conjunto de 3 puntos característicos, se logra mayor invarianza y se puede almacenar información de la región. De esta forma, se consigue una buena variedad de palabras, pero limitando su volumen a un conjunto finito y repetible de palabras.

Se ha optado por 3 dimensiones (ángulo, longitud del perímetro e iluminación media de la esquina) debido a que conlleva suficiente información y el se evita el coste computacional de añadir nuevas dimensiones. Para estas esquinas, y como se puede ver en los resultados de la prueba anterior, se ha fijado un umbral entre puntos distantes a unos 25 píxeles.

Prueba X. Espectro del Grafo con Palabras Visuales

Objetivo: Extraer el nuevo espectro del grafo usando las palabras visuales como nodos.

Resultados: Se esperaba obtener un espectro irregular como los obtenidos en el primer caso, para posteriormente extraer la distancia entre grafos de distintas categorías de las imágenes faciales.

El espectro obtenido para el conjunto de rostros de la categoría “*Mujeres*” es el que se puede ver en la figura siguiente:

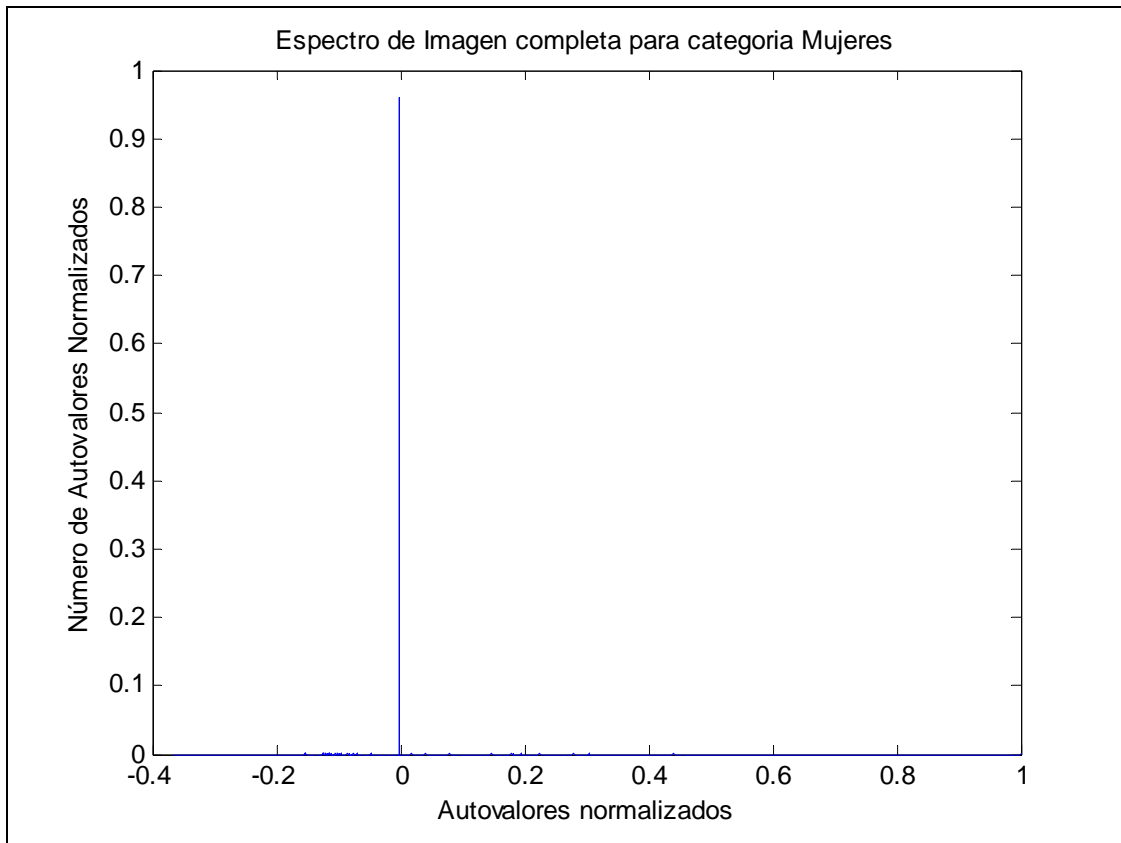


Figura 46: Espectro del grafo usando palabras visuales como nodos para la categoría *Mujeres*.

En la figura se puede observar el espectro en forma de delta centrada en el autovalor λ igual a -1 , el valor está normalizado por lo que está ligeramente desplazado a la izquierda del cero. Toda la información del espectro se ha centrado en ese valor almacenando más del 90% de la información.

Prueba XI. Variaciones en el Espectro

Objetivo: Obtener un espectro que aporte información y en la medida de lo posible sea propio para cada categoría. Para ello, se van a modificar los dos umbrales variables. El primero, modifica la definición de la palabra visual reduciendo la distancia entre los puntos que forman una esquina y obteniéndose menor número de palabras. El segundo umbral, está presente en la definición de distancia entre palabras. Al definirla de forma más flexible, se obtiene mayor número de repeticiones por lo tanto se reduce a su vez el número de palabras. El caso contrario, aumentar el número de palabras y por tanto relaciones, también se ha probado.

Resultados: Se muestra el espectro resultante con las variaciones introducidas:

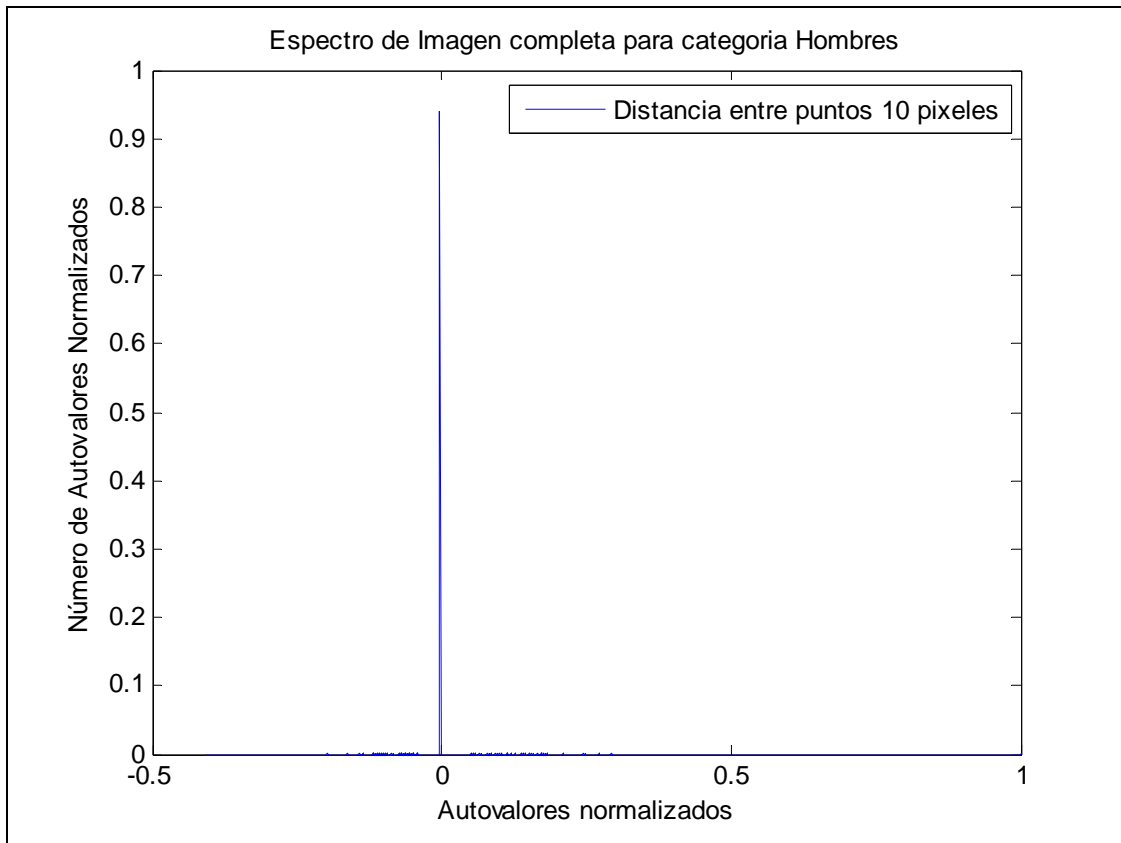


Figura 47: Espectro obtenido disminuyendo la distancia entre puntos a 10 pixeles y por tanto reduciendo el número de nodos en el grafo.

Por los resultados, observamos que no se ha conseguido el objetivo buscado manteniéndose la forma de delta centrada en $\lambda -1$. Evidentemente, se debe a la forma del grafo construido en base a las relaciones existentes entre sus nodos. Para evitar este comportamiento se define una nueva forma de construir la matriz de adyacencia, definiendo un grafo diferente (CASO 3) basado en una segmentación de la imagen en celdas.

CASO 3: Construcción del grafo usando palabras visuales con la imagen segmentada en celdas

Prueba XII. Segmentación de la imagen

Objetivo: Segmentar la imagen en celdas de un tamaño a determinar de forma empírica en base al número de palabras por celda.

Resultados: Debido a que se ha elegido trabajar con imágenes de un tamaño fijo (384x256 píxeles), se ha optado por dividir la imagen en celdas del mismo tamaño resultando las siguientes posibilidades por cada imagen:

- 4 celdas (192x128 px): Obtiene una media de 100 palabras por celda.
- 16 celdas (96x64 px): Obtiene una media de 20 palabras por celda.

- *32 celdas (48x32 px)*: Para este caso sale una media de unas 5 palabras por celda, pero teniendo en cuenta que existen muchas celdas que carecen de palabras, por lo que se ha descartado esta opción debido al bajo número de enlaces que se forman.

El propósito de esta segmentación es variar lo máximo posible la estructura del grafo formado, intentando variar las relaciones entre los nodos. Por ello, se ha optado elegir el término medio, 16 celdas por imagen, que posee un número suficiente de puntos por celda logrando interconexiones pequeñas entre los nodos de la misma celda y enlaces con aquellas palabras de distintas celdas.

A su vez se ha comprobado que la nueva forma de construir el grafo cumple la propiedad de ser un grafo de orden libre.

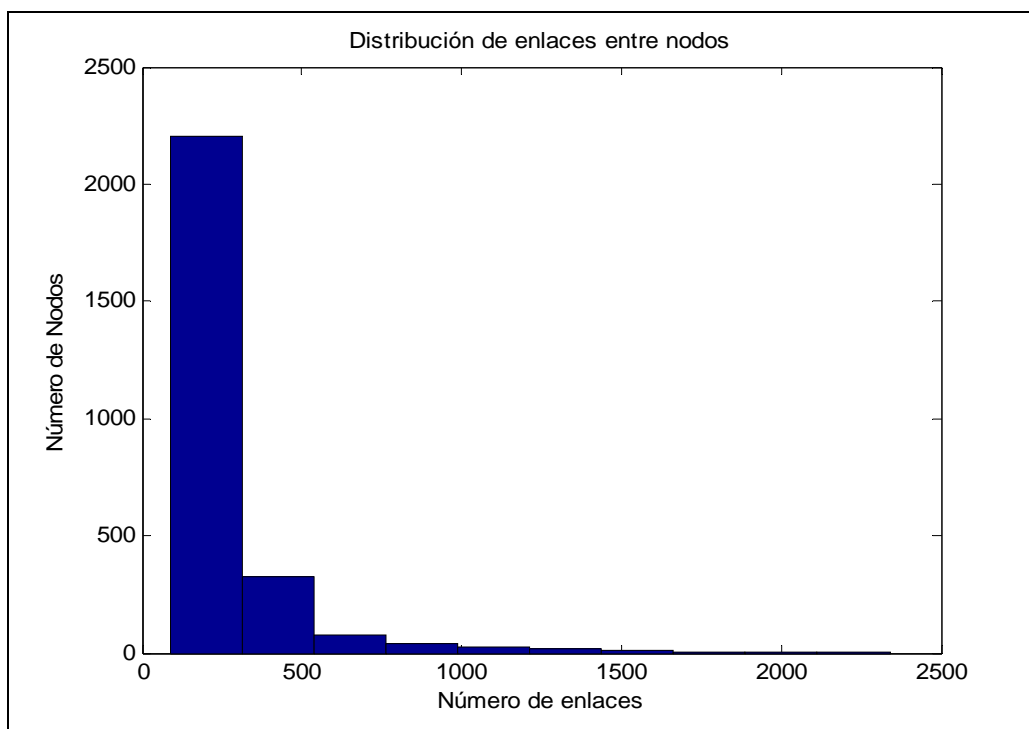


Figura 48: Distribución de los enlaces por nodos para la nueva definición del grafo, segmentando la imagen en celdas.

Prueba XIII. Espectro del Grafo con Segmentación de la Imagen en celdas.

Objetivo: Comprobar la forma del espectro para la nueva estructura del grafo.

Resultados: El resultado del espectro obtenido a partir de la nueva definición del grafo es, a la vista de la siguiente figura, prácticamente igual al caso anterior, donde se acumula la mayoría de la información en $\lambda - 1$ normalizado. Por tanto, imposibilita llevar a cabo la comparación entre espectros para las distintas categorías.

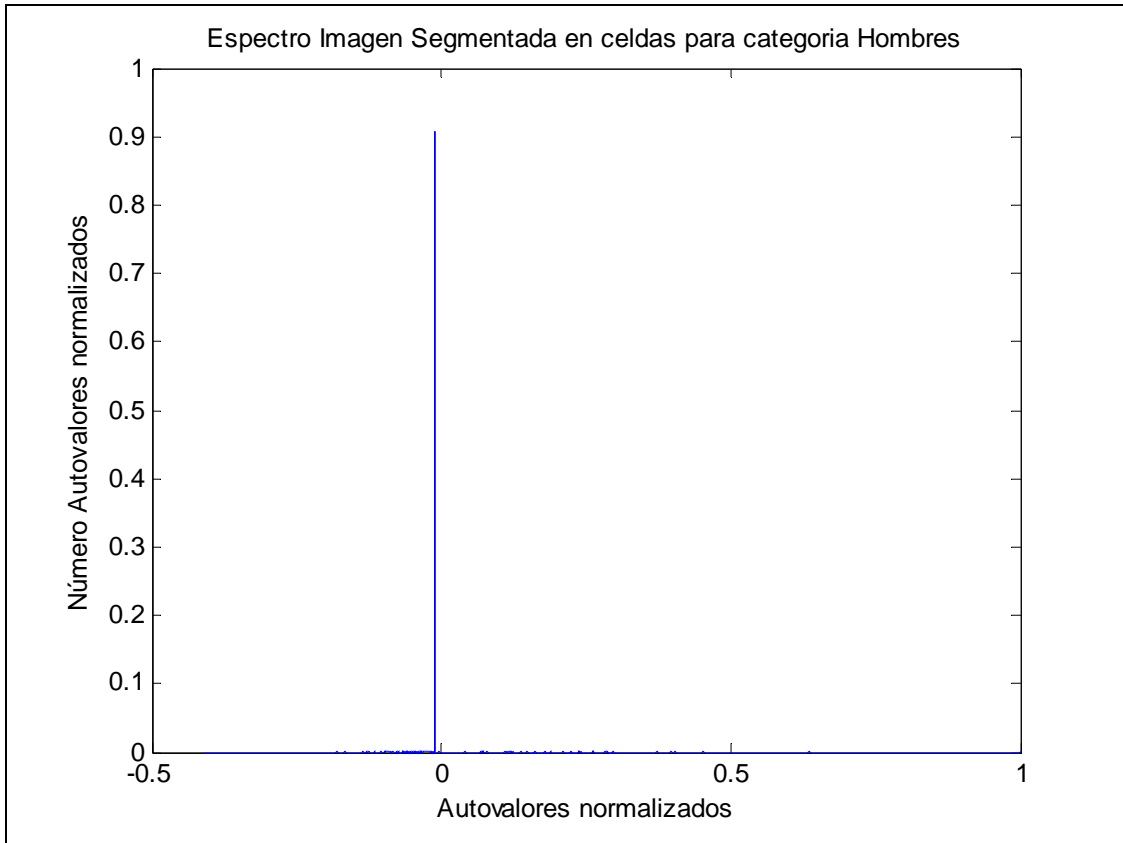


Figura 49: Espectro del grafo obtenido con la imagen segmentada en celdas para la categoría *Hombres*.

6. CONCLUSIONES

El proceso de investigación realizado en este proyecto fin de carrera ha revelado diversas conclusiones interesantes en el campo del análisis espectral de imágenes y más en concreto en el análisis de imágenes faciales.

El objetivo es valorar la relevancia del análisis espectral de grafos en el procesado de imágenes. Se ha recurrido para ello a varias pruebas experimentales usando diversas técnicas y planteamientos a fin de poder sacar conclusiones a favor o en contra de las mismas.

En primer lugar, se ha comprobado la necesidad de desarrollar un preprocesado de la imagen para obtener buenos resultados. Este preprocesado se ha basado en segmentaciones por colores que permiten eliminar la información irrelevante y quedarse exclusivamente con las características del rostro donde sí existe contenido significativo para la extracción de información. Esta segmentación podría eliminar información que a priori pareciera necesaria, pero a la vista de los resultados obtenidos para los distintos experimentos, se ha comprobado que la etapa de preprocesado es beneficiosa para aislar la información que es útil, lo cual proporciona mejores resultados que el uso de la imagen al completo.

Una vez comprobada la necesidad de realizar un preprocesado, se analizan las distintas técnicas de extracción de puntos característicos o puntos de interés. Estas técnicas se basan en operadores lógicos y, con cada una de ellas, se obtienen puntos característicos diferentes. La técnica final elegida fue una combinación entre el algoritmo de las esquinas de Harris y LoG, debido a que el conjunto de puntos que se obtenían, tanto en número como localización de los mismos, era el más adecuado. Se requerían puntos de interés que representaran la información de la imagen para posteriormente poder construir el grafo asociado, y los algoritmos seleccionados para el análisis cumplen esa propiedad así como otras, mencionadas anteriormente, como la robustez y la invarianza. Además se ha tenido en cuenta el coste computacional y el tiempo de ejecución de los mismos, llegando a un compromiso entre tiempo y resultados, esto es, intentando minimizar el tiempo de ejecución sin perder mucha calidad en los resultados.

Las siguientes conclusiones están relacionadas con la elaboración del grafo. Este punto ha desencadenado el desarrollo de varias posibilidades y ha sido la etapa más costosa de implementar desde el punto de vista del planteamiento. Para su desarrollo, se establecen unos objetivos a cumplir y unos criterios de decisión para implementar una plataforma que barajase las distintas opciones. El objetivo es poder representar la información de la imagen en un grafo y para ello es necesario describir unas relaciones entre los puntos que se habían obtenido. Estas relaciones deben contener información invariante de la imagen y por ello se optó por usar la información geométrica de la misma, relacionando los puntos en función de las distancias entre ellos. Con esto, se ha conseguido dotar del mismo significado de una característica a varios puntos de la imagen relacionándolos entre sí por proximidad. Esta posibilidad ha resultado aceptable de acuerdo a los resultados obtenidos, y se ha podido concluir que esta forma de construir el grafo representa información inicial de la imagen. A la vista de los

resultados, se observa que el sistema detecta que la similitud entre imágenes que comparten las mismas características es mayor que la que existe entre imágenes de individuos que no comparten esas características en cuestión. Del mismo modo se ha podido observar la similitud existente en imágenes de individuos del mismo género.

A pesar de estos resultados, los márgenes de disimilitud obtenidos son bajos no siendo, por tanto, este método el más adecuado para la diferenciación de características locales. Sin embargo, los resultados obtenidos para la diferenciación de género si son más esperanzadores y respaldan el uso de esta estrategia. A simple vista, una persona suele diferenciar si un individuo es hombre o mujer, y esto se debe a las características o rasgos que posee asociados a este hecho. Estas características al estar representadas en el grafo, agrupadas por género, han proporcionado unos resultados aceptables y que permiten demostrar que efectivamente contienen el significado de la imagen.

Estos resultados empíricos han sido producto de una comparativa entre espectros para lo cual se han tenido que normalizar a fin de conseguir que estos fueran independientes de la imagen y del número de puntos usados (esto proporcionará una mayor invarianza en todo el proceso). Gracias a este proceso, se ha podido realizar la diferenciación entre espectros asegurando que no se ha perdido o manipulado la información en la normalización de los mismos.

Uno de los problemas encontrados tras las primeras pruebas es que la hipótesis de usar la información geométrica de una imagen no presenta la invarianza necesaria, por lo que se extendió el enfoque del proyecto apoyándonos en una línea de investigación actual. Para llevar a cabo esta nueva investigación, hubo que definir unas *palabras visuales* usando descriptores locales. Como primera apuesta, se opta por un descriptor local de 3 dimensiones apoyado en los puntos característicos extraídos en la primera etapa del proyecto. Dicha implementación proporciona mayor invarianza además de permitirnos almacenar más información de la imagen. A su vez, se han estudiado variaciones de la forma de definir las relaciones entre las palabras. Los aspectos de abstracción semántica se implementan mediante las colecciones de imágenes previamente categorizadas para la generación del grafo y de su espectro. Este espectro resulta ser mayormente degenerado por lo que la posibilidad de comparar espectros es muy reducida. Un posible problema radica en la forma de implementar el grafo y se apunta como posibilidad el cambio en la definición del mismo.

En este sentido, se ha procedido a segmentar la imagen en celdas y tratar cada celda como una imagen individual, relacionando sólo *palabras visuales* de una misma imagen y hallando luego la media por categoría. Con este nuevo enfoque se han obtenido los mismos resultados negativos al no poder extraerse información del espectro. Esto nos hace concluir que la estrategia seguida para la construcción de un único grafo con diferentes imágenes, siguiendo la forma empleada para textos, y para la base de datos elegida, no ofrece resultados aceptables.

Para finalizar, se concluye que la capacidad de clasificación de características abstractas de imágenes con esta estrategia de definición de grafos es insuficiente, siendo necesaria una redefinición sustancial en la forma de obtener el grafo para la imagen.

7. LÍNEAS FUTURAS

Durante la realización de esta investigación se han abierto distintas líneas a seguir con el fin de seguir indagando en el uso del espectro en imágenes. Algunas de ellas pueden ser tomadas como referencia para el comienzo de una nueva investigación:

1. Cambio en la definición y construcción del grafo: Una construcción y definición alternativa del grafo puede conllevar a resultados distintos, sobre todo buscando otros significados en una imagen.
2. Cambio de la matriz del grafo: El uso de otras matrices para la representación del grafo, en vez de la matriz de adyacencia, es otro foco de investigación que puede proporcionar propiedades diferentes.
3. Diferentes criterios y métodos de extracción de características: El uso de otros métodos de extracción de características, puede suponer una mejora en los resultados, o la definición de otros criterios de diseño buscando diferentes significados.
4. Diferentes definiciones de Palabras Visuales: La elección de otros tipos de palabras visuales, ya sea usando descriptores locales distintos (p.ej: descriptores SIFT) o bien incrementando el número de dimensiones al descriptor utilizado.
5. Identificación de objetos en imágenes: La búsqueda de objetos en una imagen en vez de la búsqueda de características también puede ser un tema a investigar.

8. BIBLIOGRAFÍA

- [1] J. Santaella Vallejo, “*Análisis de Textos basado en Propiedades Espectrales de Grafo*, PFC Universidad Carlos III de Madrid”, Enero 2006.
- [2] A. J. Goldstein, L. D. Harmon, and A. B. Lesk, “*Identification of Human Faces*,” Proc. IEEE, May 1971, Vol. 59, No. 5, 748-760.
- [3] L. Sirovich and M. Kirby, “*A Low-Dimensional Procedure for the Characterization of Human Faces*,” J. Optical Soc. Am. A, 1987, Vol. 4, No.3, 519-524.
- [4] M. A. Turk and A. P. Pentland, “*Face Recognition Using Eigenfaces*,” Proc. IEEE, 1991, 586-591.
- [5] Color Feret, *Facial Image Database*, Image Group, Information Access Division, ITL, National Institute of Standards and Technology, October 2003.
- [6] Richard K. Belew, “*Finding Out About: A Cognitive Perspective on Search Engine Technology and the WWW*”, Cambridge University Press, 2000.
- [7] H. Plaza López, “*Recuperación, Acceso y Organización de la Información*”, asignatura de Recuperación y acceso a la información
- [8] S.Anayan and M.Kiselev, “*Automated Analysis of Unstructured Text*” para Technology and Implementations, Megaguters R&D projects.
- [9] Ananiadou, Sophia and McNaught, John. “*Text Mining for Biology and Biomedicine*”. Artech House Publishers, 2006 ISBN 1-58053-984-X, 302pp.
- [10] Stegmann and Grohmann, “*Hypothesis generation guided by co-word clustering*”, Akadémiai Kiadó, co-published with Springer Science+Business Media B.V., Formerly Kluwer Academic Publishers B.V., pp 111-135, 2003
- [11] Tanja Bekhuis, “*Conceptual biology, hypothesis discovery, and text mining: Swanson's legacy*”, Department of Library & Information Science, School of Information Sciences, University of Pittsburgh, Abril 2006.
- [12] Feng Tang, “*Feature Detection in Images*”, IJCV (*International Journal of Computer Vision*), Octubre 2004.
- [13] M.^a del Pilar Cantero Blanco, “*POSIBLE APLICACIÓN DE LA MINERÍA DE TEXTOS A LOS TRABAJOS DE LA COMISIÓN MINISTERIAL DE INFORMÁTICA*”, General de Planificación y Coordinación Informática Ministerio de Trabajo y Asuntos Sociales, Septiembre 2004
- [14] SPSS, “*Text Mining para Clementine*”, Presentación Comercial SPSS (Statistical Product and Service Solutions Iberica, S.L)

- [15] Nicky Hager, “*Exposing the Global Surveillance System*”, in *Covert Action Quarterly*, 26 December 1997
- [16] T.Gruber, “*A TRANSLATION APPROACH TO PORTABLE ONTOLOGY SPECIFICATIONS*”, *KNOWLEDGE ACQUISITION*. Vol.5,1993
- [17] Alex Biryukov “*Using Semantic Web Techonologies to Analyze Learning Content*”, *IEEE Computer Society* September 2007
- [18] Gottfried Vossen, Miltiadis Lytras, and Nick Koudas, “*Revisiting the (Machine) Semantic Web: The Missing Layers for the Human Semantic Web*”, *IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING*, VOL. 19, NO. 2, FEBRUARY 2007
- [19] Dragan Gamberger, Rudjer Boskovic, “*Ontologies in Medical Knowledge Representation*”, Alan Jovic, Marin Prcela, *Institute, Laboratory of Informational Systems*, *Proceedings of the ITI 2007 29th Int. Conf. on Information Technology Interfaces*, June 25-28, 2007, Cavtat, Croatia
- [20] Hao Zhang Oliver van Kaick Ramsay Dyer, “*Spectral Methods for Mesh Processing and Analysis*”, GrUVi Lab, School of Computing Science, Simon Fraser University, BC, Canada, 2007
- [21] Song Jun-feng, Zhang Wei-ming, Xiao Wei-dong, Xu Zhen-ning, “*Study on Construction and Integration of Military Domain Ontology, Situation Ontology and Military Rule Ontology for Network Centric Warfare*”, School of Information System and Management, The National University of Defense Technology, Changsha 410073,China
- [22] “*Echelon Surveillance System Threatens U.S. Privacy Rights – Government*” *Activity Online Newsletter*, Sept, 2000
- [23] T. Gruber, “*A Translation Approach to Portable Ontology Specifications*”, *Knowledge Acquisition*, Vol 5., pp 199-220, 1993.
- [24] Blagosklonny MV, Pardee AB. “*Unearthing the gems*”. *Nature Magazine*. 2002; pp. 416:373
- [25] Sujeewa Alwis, “*Challenges in Face Recognition biometrics*”, *Cybula-High performance pattern recognition systems* 2004.
- [26] M. Turk, A. Pentland, “*Eigenfaces for Recognition*”, *Journal of Cognitive Neuroscence*, Vol. 3, No. 1, 1991, pp. 71-86
- [27] A. Pentland, B. Moghaddam, T. Starner, “*View-Based and Modular Eigenspaces for Face Recognition*”, *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 21-23 June 1994, Seattle, Washington, USA, pp. 84-91

- [28] H. Moon, P.J. Phillips, “*Computational and Performance aspects of PCA-based Face Recognition Algorithms*”, Perception, Vol. 30, 2001, pp. 303-321
- [29] W. Zhao, R. Chellappa, A. Krishnaswamy, “*Discriminant Analysis of Principal Components for Face Recognition*”, Proc. of the 3rd IEEE International Conference on Face and Gesture Recognition, FG'98, 14-16 April 1998, Nara, Japan, pp. 336-341.
- [30] K. Etemad, R. Chellappa, “*Discriminant Analysis for Recognition of Human Face Images*”, Journal of the Optical Society of America A, Vol. 14, No. 8, August 1997, pp. 1724-1733
- [31] M.S. Bartlett, J.R. Movellan, T.J. Sejnowski, “*Face Recognition by Independent Component Analysis*”, IEEE Trans. on Neural Networks, Vol. 13, No. 6, November 2002, pp. 1450-1464
- [32] C. Liu, H. Wechsler, “*Comparative Assessment of Independent Component Analysis (ICA) for Face Recognition*”, Proc. of the Second International Conference on Audio- and Video-based Biometric Person Authentication, AVBPA'99, 22-24 March 1999, Washington D.C., USA, pp. 211-216
- [33] W. Zhao, R. Chellappa, A. Rosenfeld, “*Face Recognition: A Literature Survey*”, University of Maryland. National Institute of Standards and Technology.
- [34] L. Wiskott, J.-M. Fellous, N. Krueger, C. von der Malsburg, “*Face Recognition by Elastic Bunch Graph Matching*”, IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 19, No. 7, 1997, pp. 775-779
- [35] C. Liu, H. Wechsler, “*Evolutionary Pursuit and Its Application to Face Recognition, IEEE Trans. on Pattern Analysis and Machine Intelligence*”, Vol. 22, No. 6, June 2000, pp. 570-582
- [36] J. Huang, B. Heisele, V. Blanz, “*Component-based Face Recognition with 3D Morphable Models*”, Proc. of the 4th International Conference on Audio- and Video-Based Biometric Person Authentication, AVBPA 2003, 09-11 June 2003, Guildford, UK, pp. 27-34
- [37] A. Bronstein, M. Bronstein, R. Kimmel, and A. Spira. “*3D face recognition without facial surface reconstruction*”, in Proceedings of ECCV 2004, Prague, Czech Republic, May 11-14, 2004
- [38] Fei Wang, Jingdong Wang, Changshui Zhang and James Kwok, “*Face recognition using spectral features*”, Department of Automation, State Key Laboratory of Intelligent Technology and Systems, Tsinghua University, Beijing, China Department of Computer Science, The Hong Kong University of Science and Technology, Clear Water Bay, Hong Kong January 2007.

- [39] Berk Gökberk, M. Okan İrfanoğlu, Lale Akarun and Ethem Alpaydın, “*Learning the best subset of local features for face recognition*” Department of Computer Engineering, Boğaziçi University, Bebek, Istanbul, Turkey, November 2006.
- [40] Hongchuan Yu, Mohammed Bennamouna, “*Complete invariants for robust face recognition*,” School of Computer Science and Software Engineering, University of Western Australia, Perth, WA6009, Australia, August 2006.
- [41] Análisis Sukno, Federico M.; Ordas, Sebastian; Butakoff, Constantine; Cruz, Santiago; Frangi, Alejandro F.,”*Active Shape Models with Invariant Optimal Features: Application to Facial*,” Pattern Analysis and Machine Intelligence, IEEE Transactions on Volume 29, Issue 7, July 2007 Page(s):1105 – 1117
- [42] D.G. Lowe. “*Distinctive image features from scale-invariant keypoints*”. *Int. J. Comput. Vision*, 60(2):91–110, 2004.
- [43] M. Villamizar, A. Sanfeliu, and J. Andrade- Cetto. “*Computation of rotation local invariant features using the integral image for real time object detection*”. IEEE Comp. Soc. in *Proc. 18th IAPR Int. Conf. Pattern Recog.*, Hong Kong, Aug. 2006.
- [44] J.J. Yokono and T. Poggio. “*Oriented filters for object recognition: An empirical study*”. In *Proc. 6th IEEE Int. Conf. Automatic Face Gesture Recog.*, pages 755–760, Seoul, 2004.
- [45] J.J. Yokono and T. Poggio. “*Rotation invariant object recognition from one training example*”. Technical Report 2004-010, MIT AI Lab., Apr. 2004.
- [46] David Liu, Tsuhan Chen, “*Semantic-Shift for Unsupervised Object Detection*”, Department of Electrical and Computer Engineering, Carnegie Mellon University Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop
- [47] Witkin, A.P. 1983. “*Scale-space filtering*”. In *International Joint Conference on Artificial Intelligence*, Karlsruhe, Germany, pp. 1019-1022.
- [48] M. Keyvanpour,S. Asbaghi, M. Fathy, “*A New Scheme of Automatic Semantic Propagation in the Image Data base Using a Hierarchical Structure of Semantics*”, 18th International Workshop on Database and Expert Systems Applications IEEE Computer Society, 1529-4188/07 © 2007 IEEE DOI 10.1109/DEXA.2007.72
- [49] Parikh, Devi; Chen, Tsuhan “*Unsupervised Learning of Hierarchical Semantics of Objects (hSOs)*”, *Computer Vision and Pattern Recognition*, 2007. CVPR apos;07. IEEE Conference on Volume , Issue , 17-22 June 2007 Page(s):1 – 8
- [50] J.Farkas, I. Derényi, A.Barabasi, T.Bishkek, “*Spectra of “real-world” graphs: Beyond the semicircle law*” *PHYSICAL REVIEW E*, VOLUME 64, 026704, 2001

[51] Timothee Cour, Florence Benezit, Jianbo Shi. “*Spectral Segmentation with Multiscale Graph Decomposition*”. IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), 2005.

[52] P. Zhu and R. C. Wilson, “*A Study of Graph Spectra for Comparing Graphs*”, Computer Science Department, University of York, UK

[53] P. F. Felzenszwalb and D. P. Huttenlocher, “*Efficient Graph-Based Image Segmentation*”, Artificial Intelligence Lab, Massachusetts Institute of Technology and Computer Science Department, Cornell University

[54] K. Mikolajczyk and C. Schmid, “*A Performance Evaluation of Local Descriptors*”, IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 27, NO. 10, OCTOBER 2005

[55] Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, Cédric Bray, “*Visual Categorization with Bags of Keypoint*”, Xerox Research Centre Europe 2006.

[56] Xuerong Yong, “*On the distribution of eigenvalues of a simple undirected graph*”, Department of Computer Science, The Hong Kong University of Science and Technology, Clear Water Bay, Kowloon, Hong Kong March 1999



ANEXO

CÓDIGOS MATLAB

ÍNDICE

CASO 1: Construcción directa del grafo a partir de los puntos característicos.....	95
recog.m.....	95
Crea_matriz.m.....	98
Draw2.m.....	99
findLocalMaxima.m.....	99
kp_harris.m.....	100
kp_log.m.....	101
kp_harrislaplace.m.....	103
matriz_basica.m.....	105
pt_inv.m.....	106
union.m.....	106
CASO 2: Construcción del grafo usando <i>palabras visuales</i> para toda la imagen.....	107
angulo.m.....	107
difPalabras.m.....	108
eliminarRep.m.....	108
esquinas.m.....	109
IluminacionMedia.m.....	110
longitudTotal.m.....	110
main2.m.....	111
uniraMatriz.m.....	111
ventana.m.....	112
CASO 3: Construcción del grafo usando <i>palabras visuales</i> con la imagen segmentada en celdas.....	113
Main3.m.....	113
reducirImagen.m.....	114

CASO 1: Construcción directa del grafo a partir de los puntos característicos

recog.m

```
function [n,ejeX,nt,ejeXt,ancho,anchot,areaT]=recog(filename);

a = imread(filename);
r = im2col(a(:,:,1),[size(a,1) size(a,2)]);%Pasamos la componente roja
de la imagen a columna.
b = im2col(a(:,:,2),[size(a,1) size(a,2)]);%Pasamos la componente
verde de la imagen a columna.
c = im2col(a(:,:,3),[size(a,1) size(a,2)]);%Pasamos la componente azul
de la imagen a columna.
%CIELUV
%b = colorspace(a,'RGB_>CIELUV')
original = [r b c];
grande = [r b c];
nuevo = ones(size(grande,1),3)*255;
nuevod = ones(size(grande,1),3)*255;
%Aplicamos kmedias y sacamos 2 centroides
[indices ce]=kmeans(double(grande),2,'EmptyAction',
'singleton','Replicates',2);

ce;
%Cogemos los pixels que se ajustan al primero de los centroides.
valores = find(indices == 1);
size(valores);
for i =1:size(valores,1)
grande(valores(i),:) = ce(1,:);
nuevo(valores(i),:) = original(valores(i),:,:);
end
%Cogemos los pixels que se ajustan al segundo de los centroides.
valoresd = find(indices == 2);
size(valoresd);
for i =1:size(valoresd,1)
grande(valoresd(i),:) = ce(2,:);
nuevod(valoresd(i),:) = original(valoresd(i),:,:);
end
%Hay que reformatear la imagen en filas y en RGB.
imagen_nueva(:,:,1) = reshape(grande(:,1),[size(a,1) size(a,2)]);
imagen_nueva(:,:,2) = reshape(grande(:,2),[size(a,1) size(a,2)]);
imagen_nueva(:,:,3) = reshape(grande(:,3),[size(a,1) size(a,2)]);
%imshow((imagen_nueva))
%title('Imagen segmentada en 2 regiones sin color')
% figure
imagend(:,:,1) = reshape(nuevod(:,1),[size(a,1) size(a,2)]);
imagend(:,:,2) = reshape(nuevod(:,2),[size(a,1) size(a,2)]);
imagend(:,:,3) = reshape(nuevod(:,3),[size(a,1) size(a,2)]);
% subplot(1,2,1);imshow(uint8(imagend))

% if (mean(ce(1,:)) < mean(ce(2,:)))%Si la media de los centroides es
mayor es el fondo
```

```

%     title('Fondo de imagen')
% else
%     title('Imagen con información relevante, sección rostro: corpus
de trabajo')
% end
imagen(:,:,1) = reshape(nuevo(:,1),[size(a,1) size(a,2)]);
imagen(:,:,2) = reshape(nuevo(:,2),[size(a,1) size(a,2)]);
imagen(:,:,3) = reshape(nuevo(:,3),[size(a,1) size(a,2)]);
% subplot(1,2,2);imshow(uint8((imagen)))

%Repetimos la segmentación en 2 zonas sólo con la parte de la imagen
con
%información relevante.

%Trabajamos con el corpus
if (mean(ce(1,:)) > mean(ce(2,:)))%Si hay más de la mitad de los
píxeles totales
    imagen=imagend;
end

    r = im2col(imagen(:,:,1),[size(a,1) size(a,2)]);
    g = im2col(imagen(:,:,2),[size(a,1) size(a,2)]);
    b = im2col(imagen(:,:,3),[size(a,1) size(a,2)]);
    info = [r g b];

    nuevo = ones(size(info,1),3)*255;
    nuevod = ones(size(info,1),3)*255;
    nuevot = ones(size(info,1),3)*255;
% %Aplicamos kmedias y sacamos 2 centroides
[indices ce]=kmeans(double(info),3,'EmptyAction',
'singleton','Replicates',2);
ce;
%Cogemos los pixels que se ajustan al primero de los centroides.
valores = find(indices == 2);

for i =1:size(valores,1)
nuevo(valores(i),:) = info(valores(i),:);
end
valoresd = find(indices == 1);
for i =1:size(valoresd,1)
nuevod(valoresd(i),:) = info(valoresd(i),:);
end
valoresc = find(indices == 3);

for i =1:size(valoresc,1)
nuevot(valoresc(i),:) = info(valoresc(i),:);
end

imag(:,:,1) = reshape(nuevod(:,1),[size(a,1) size(a,2)]);
imag(:,:,2) = reshape(nuevod(:,2),[size(a,1) size(a,2)]);
imag(:,:,3) = reshape(nuevod(:,3),[size(a,1) size(a,2)]);
% figure
% subplot(1,3,1);imshow(uint8(imag))%Centroide 1
% title(['Segunda tandal
no.pixels=',num2str(size(valoresd,1)),',ce=',num2str(mean(ce(1,:))])])
imag2(:,:,1) = reshape(nuevo(:,1),[size(a,1) size(a,2)]);
imag2(:,:,2) = reshape(nuevo(:,2),[size(a,1) size(a,2)]);
imag2(:,:,3) = reshape(nuevo(:,3),[size(a,1) size(a,2)]);

```



```

% subplot(1,3,2);imshow(uint8(imag2))%Centroide 2
% title(['Segunda tanda2
no.pixels=',num2str(size(valores,1)), 'ce=',num2str(mean(ce(2,:)))]])
imag3(:,:,1) = reshape(nuevot(:,1),[size(a,1) size(a,2)]);
imag3(:,:,2) = reshape(nuevot(:,2),[size(a,1) size(a,2)]);
imag3(:,:,3) = reshape(nuevot(:,3),[size(a,1) size(a,2)]);
% subplot(1,3,3);imshow(uint8(imag3))%Centroide 3
% title(['Segunda tanda3
no.pixels=',num2str(size(valoresc,1)), 'ce=',num2str(mean(ce(3,:)))]])

%Seleccionamos parte de piel y parte con caracterísitcas.
if (min(ce)==ce(1,:))
    caract=imag;
    if (max(ce)==ce(2,:))
        piel=imag3;
    else
        piel=imag2;
    end
else if (min(ce)==ce(2,:))
    caract=imag2;
    if (max(ce)==ce(1,:))
        piel=imag3;
    else
        piel=imag;
    end
else
    caract=imag3;
    if (max(ce)==ce(1,:))
        piel=imag2;
    else
        piel=imag;
    end
end
end
figure;
subplot(1,2,1);imshow(uint8(caract));title('Caracterisitcas');
subplot(1,2,2);imshow(uint8(piel));title('Piel');

%Sacamos puntos invariantes de la cara que nos haran de base del
grafo.
pts=pt_inv(caract,a);
ptsTotal=pt_inv(a,a);%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

m=matriz_basica(pts);%Une cada nodo con al menos otro nodo para
conseguir una lista encadenada.
matriz=crea_matriz(m,pts,200);%Une los nodos que se encuentren a una
distancia inferior del umbral=200.
%Sacamos los autovalores de la matriz de adyacencia

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
mT=matriz_basica(ptsTotal);%Une cada nodo con al menos otro nodo para
conseguir una lista encadenada.
matrizT=crea_matriz(mT,ptsTotal,200);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

AVm=eig(matriz);
AVm=AVm';

```

```

AV=zeros(1,size(AVm));%Inicializamos con el mínimo.
% Eliminamos los valores en -1
cont=0;
for i=1:length(AVm)
    if round(AVm(i)*10)~=-10
        AV(i)=AVm(i);
        cont=cont+1;
    end
end

[n,ejeX]=hist(AV,size(pts,1));
%Normalizamos
n=n.*100./sum(n);%cont;
ancho=(max(AV)-min(AV));
% figure;plot(ejeX,n)
% title(['Histograma de los Autovalores de: ',filename]);

AVT=eig(matrizT);
AVT=AVT';
AVt=zeros(1,size(AVT));%Inicializamos con el mínimo.
%Eliminamos los valores en -1
cont=0;
for i=1:length(AVT)
    if round(AVT(i)*10)~=-10
        AVt(i)=AVT(i);
        cont=cont+1;
    end
end

[nt,ejeXt]=hist(AVt,size(ptsTotal,1));
% figure;plot(ejeXt,nt)
% title(['Histograma de los Autovalores Totales de: ',filename]);
nt=nt.*100./cont;
anchot=(max(AVt)-min(AVt));

%disp('Area del espectro:');
areaT=sum(abs(nt.*ejeXt));

```

Crea_matriz.m

```

function matriz=crea_matriz(mat,pts,umbral)
%Pasando la matriz total hasta el momento de la iteración, la lista de
puntos extraídos y el umbral fijado.

matriz=mat;
%umbral=200;%Una distancia de unos 10 pixels en cada direccion.
%Bucle que establece un enlace entre dos nodos que se encuentren a
menor
%distancia del umbral.
for i=1:size(pts,1)
    for j=1:size(pts,1)
        if(i~=j)

            if (dist(pts(i,:),pts(j,:))<umbral)
                matriz(i,j)=1;%dist(pts(i,:),pts(j,:));
            end
        end
    end
end

```

```

        matriz(j,i)=1;%dist(pts(i,:),pts(j,:));
    end
end
end%for j
end

```

Draw2.m

```

function draw2(img,pt)

figure;
imshow(uint8(img));
title('Puntos invariantes unidos')
hold on;
axis off;
switch size(pt,2)
    case 2%Escala de grises
        s = 2;
        for i=1:size(pt,1)
            rectangle('Position',[pt(i,2),pt(i,1),2,2], 'Curvature',[1
                1], 'EdgeColor','y', 'LineWidth',2);
        end
    case 3%RGB
        for i=1:size(pt,1)
            rectangle('Position',[pt(i,2)-pt(i,3),pt(i,1)-
                pt(i,3),2*pt(i,3),2*pt(i,3)], 'Curvature',[0
                0], 'EdgeColor','w', 'LineWidth',2);
        end
end
    line(pt(:,1)',pt(:,2)', 'Color','y', 'LineWidth',2);
end

```

findLocalMaxima.m

```

function [row,col,max_local] = findLocalMaximum(val,radius)
% Determine the local maximum of a given value
%
% Author :: Vincent Garcia
% Date   :: 09/02/2007
%
% INPUT
% =====
% val    : the NxM matrix containing values
% radius : the radius of the neighborhood
%
% OUTPUT
% =====
% row    : the row position of the local maxima
% col    : the column position of the local maxima

```

```

% max_local : the NxM matrix containing values of val on unique
local maximum
%
% EXAMPLE
% =====
% [l,c,m] = findLocalMaximum(img,radius);

% FIND LOCAL MAXIMA BY DILATION (FAST) /\ NON UNIQUE /\
% mask = fspecial('disk',radius)>0;
% val2 = imdilate(val,mask);
% index = val==val2;
% [row,col] = find(index==1);
% max_local = zeros(size(val));
% max_local(index) = val(index);

% FIND UNIQUE LOCAL MAXIMA USING FILTERING (FAST)
mask = fspecial('disk',radius)>0;
nb = sum(mask(:));
highest = ordfilt2(val, nb, mask);
second_highest = ordfilt2(val, nb-1, mask);
index = highest==val & highest~=second_highest;
max_local = zeros(size(val));
max_local(index) = val(index);
[row,col] = find(index==1);

```

End

kp_harris.m

```

function points = kp_harris(im)
% Extract keypoints using Harris algorithm (with an improvement
% version)
%
% Author :: Vincent Garcia
% Date   :: 05/12/2007
%
% INPUT
% =====
% im      : the graylevel image
%
% OUTPUT
% =====
% points : the interest points extracted
%
% REFERENCES
% =====
% C.G. Harris and M.J. Stephens. "A combined corner and edge
detector",
% Proceedings Fourth Alvey Vision Conference, Manchester.
% pp 147-151, 1988.
%

```

```

    % Alison Noble, "Descriptions of Image Surfaces", PhD thesis,
Department
    % of Engineering Science, Oxford University 1989, p45.
    %
    % C. Schmid, R. Mohrand and C. Bauckhage, "Evaluation of Interest
Point Detectors",
    % Int. Journal of Computer Vision, 37(2), 151-172, 2000.
    %
    % EXAMPLE
    % =====
    % points = kp_harris(im)

    % only luminance value
im = double(im(:,:,1));
sigma = 1.5;

    % derivative masks
s_D = 0.7*sigma;
x = -round(3*s_D):round(3*s_D);
dx = x .* exp(-x.*x/(2*s_D*s_D)) ./ (s_D*s_D*s_D*sqrt(2*pi));
dy = dx';

    % image derivatives
Ix = conv2(im, dx, 'same');
Iy = conv2(im, dy, 'same');

    % sum of the Auto-correlation matrix
s_I = sigma;
g = fspecial('gaussian',max(1,fix(6*s_I+1)), s_I);
Ix2 = conv2(Ix.^2, g, 'same'); % Smoothed squared image
derivatives
Iy2 = conv2(Iy.^2, g, 'same');
Ixy = conv2(Ix.*Iy, g, 'same');

    % interest point response
cim = (Ix2.*Iy2 - Ixy.^2)./(Ix2 + Iy2 + eps); %
Alison Noble measure.
    % k = 0.06; cim = (Ix2.*Iy2 - Ixy.^2) - k*(Ix2 + Iy2).^2; %
Original Harris measure.

    % find local maxima on 3x3 neighborhood
[r,c,max_local] = findLocalMaximum(cim,3*s_I);

    % set threshold 1% of the maximum value
t = 0.01*max(max_local(:));

    % find local maxima greater than threshold
[r,c] = find(max_local>=t);

    % build interest points
points = [r,c];
end

```

kp_log.m

```

function [points] = kp_log(img,o_nb_blobs)
% Extract keypoints using Laplacian of Gaussian (LoG) algorithm
%
% Author :: Vincent Garcia
% Date   :: 05/12/2007
%
% INPUT
% =====
% img      : the graylevel image
% o_nb_blobs : (optional) number of blobs detected
%
% OUTPUT
% =====
% points : the interest points extracted
%
% REFERENCES
% =====
% Lindeberg, T. Feature Detection with Automatic Scale Selection
% IEEE Transactions Pattern Analysis Machine Intelligence, 1998,
30, 77-116kp_harris(im)
%
% EXAMPLE
% =====
% points = kp_log(img)

% input image
img = double(img(:,:,1));

% number of blobs detected
if nargin==1
    nb_blobs = 150;
else
    nb_blobs = o_nb_blobs;
end

% Laplacian of Gaussian parameters
sigma_begin = 2;
sigma_end   = 15;
sigma_step  = 1;
sigma_array = sigma_begin:sigma_step:sigma_end;
sigma_nb    = numel(sigma_array);

% variable
img_height = size(img,1);
img_width  = size(img,2);

% calcul scale-normalized laplacian operator
snlo = zeros(img_height,img_width,sigma_nb);
for i=1:sigma_nb
    sigma      = sigma_array(i);
    snlo(:,:,i) = sigma*sigma*imfilter(img,fspecial('log',
floor(6*sigma+1), sigma),'replicate');
end

% search of local maxima
snlo_dil      = imdilate(snlo,ones(3,3,3));
blob_candidate_index = find(snlo==snlo_dil);

```

```

        blob_candidate_value = snlo(blob_candidate_index);
        [tmp,index]         = sort(blob_candidate_value, 'descend');
        blob_index         = blob_candidate_index(
index(1:min(nb_blobs,numel(index)))) );
        [lig,col,sca]      =
ind2sub([img_height,img_width,sigma_nb],blob_index);
        points            =
[lig,col,3*reshape(sigma_array(sca),[size(lig,1),1])];
end

```

kp_harrislaplace.m

```

function points = kp_harrislaplace(img)
% Extract keypoints using Harris-Laplace algorithm
%
% Author :: Vincent Garcia
% Date   :: 05/12/2007
%
% INPUT
% =====
% img    : the graylevel image
%
% OUTPUT
% =====
% points : the interest points extracted
%
% REFERENCES
% =====
% K. Mikolajczyk and C. Schmid. Scale & affine invariant interest
point detectors.
% International Journal of Computer Vision, 2004
%
% EXAMPLE
% =====
% points = kp_harrislaplace(img)

% IMAGE PARAMETERS
img          = double(img(:,:,1));
img_height  = size(img,1);
img_width   = size(img,2);

% SCALE PARAMETERS
sigma_begin = 1.5;
sigma_step  = 1.2;
sigma_nb    = 13;
sigma_array = (sigma_step.^(0:sigma_nb-1))*sigma_begin;

% PART 1 : HARRIS
harris_pts = zeros(0,3);
for i=1:sigma_nb

```

```

% scale (standard deviation)
s_I = sigma_array(i); % integración scale
s_D = 0.7*s_I; % derivative scale %0.7

% derivative mask
x = -round(3*s_D):round(3*s_D);
dx = x .* exp(-x.*x/(2*s_D*s_D)) ./ (s_D*s_D*s_D*sqrt(2*pi));
dy = dx';

% image derivatives
Ix = conv2(img, dx, 'same');
Iy = conv2(img, dy, 'same');

% auto-correlation matrix
g = fspecial('gaussian',max(1,fix(6*s_I+1)), s_I);
Ix2 = conv2(Ix.^2, g, 'same');
Iy2 = conv2(Iy.^2, g, 'same');
Ixy = conv2(Ix.*Iy, g, 'same');

% interest point response
%cim = (Ix2.*Iy2 - Ixy.^2)./(Ix2 + Iy2 + eps); %
Alison Noble measure.
k = 0.06; cim = (Ix2.*Iy2 - Ixy.^2) - k*(Ix2 + Iy2).^2; %
Original Harris measure.

% find local maxima on neighborhood
[l,c,max_local] = findLocalMaximum(cim,3*s_I);%3*s_I

% set threshold 1% of the maximum value
t = 0.2*max(max_local(:));

% find local maxima greater than threshold
[l,c] = find(max_local>=t);

% build interest points
n = size(l,1);
harris_pts(end+1:end+n,:) = [l,c, repmat(i,[n,1])];
end

% PART 2 : LAPLACE
% compute scale-normalized laplacian operator
laplace_snlo = zeros(img_height,img_width,sigma_nb);
for i=1:sigma_nb
    s_L = sigma_array(i); % scale
    laplace_snlo(:,:,i) = s_L*s_L*imfilter(img,fspecial('log',
floor(6*s_L+1), s_L),'replicate');
end
% verify for each of the initial points whether the LoG attains a
maximum at the scale of the point
n = size(harris_pts,1);
cpt = 0;
points = zeros(n,3);
for i=1:n
    l = harris_pts(i,1);
    c = harris_pts(i,2);
    s = harris_pts(i,3);

```



```

    val = laplace_snlo(l,c,s);
    if s>1 && s<sigma_nb
        if val>laplace_snlo(l,c,s-1) && val>laplace_snlo(l,c,s+1)
            cpt = cpt+1;
            points(cpt,:) = harris_pts(i,:);
        end
    elseif s==1
        if val>laplace_snlo(l,c,2)
            cpt = cpt+1;
            points(cpt,:) = harris_pts(i,:);
        end
    elseif s==sigma_nb
        if val>laplace_snlo(l,c,s-1)
            cpt = cpt+1;
            points(cpt,:) = harris_pts(i,:);
        end
    end
end
points(cpt+1:end,:) = [];

% SET SCALE TO 3*SIGMA FOR DISPLAY
points(:,3) = 3*sigma_array(points(:,3));
end

```

matriz_basica.m

```

function matriz=matriz_basica(pts)
%Forma la union de cada nodo con el nodo más cercano devolviendo la
matriz de adyacencia.

matriz=zeros(size(pts,1));
aux=10000000;%inicializamos a número muy elevado.
d=0;
ind_menor=1;
cont=0;
%Bucle que establece un enlace entre dos nodos que se encuentren a
menor
%distancia del umbral.
for i=1:size(pts,1)
    for j=1:size(pts,1)
        cont=cont+1;
        if(i~=j)
            d=dist(pts(i,:),pts(j,:));
            if (d<aux)
                aux=d;
                ind_menor=j;
            end
        end
    end
    if(cont==size(pts,1))
        matriz(i,ind_menor)=1;
        matriz(ind_menor,i)=1;
    end
end%for j
cont=0;

```

```

    ind_menor=1;
    aux=10000000;
end

```

pt_inv.m

```

function pts=pt_inv(im,original)
%Se van a extraer puntos caracterísiticos usando Harris y LoGaussian
img=rgb2gray(uint8(im));

%    imwrite(img,'tiras.pgm','PGM');

% Harris
%
    pts1 = kp_harris(img);
    pts2 = kp_log(img);

    pts2=pts2(:,1:2);%Nos quedamos con las coordenadas
    pts=[pts1;pts2];
    draw(original,pts,'Harris + LoG')

```

union.m

```

function matriz=union(pts)

matriz=zeros(size(pts,1));
umbral=200;%Una distancia de unos 10 pixels en cada direccion.
aux=10000000;
ind_menor=0;
menor=0;
cont=1;
%Bucle que establece un enlace entre dos nodos que se encuentren a
menor
%distancia del umbral.
for i=1:size(pts,1)
    for j=1:size(pts,1)
        if(i~=j)

            if (dist(pts(i,:),pts(j,:))<umbral)
                matriz(i,j)=dist(pts(i,:),pts(j,:));
                matriz(j,i)=dist(pts(i,:),pts(j,:));
            else
                cont=cont+1;
                d=dist(pts(i,:),pts(j,:));
                if(d<aux)
                    aux=d;
                    ind_menor=i;
                    menor=j;
                end
            end
        end
    end
end

```

```

        end

        end
        %En el caso de que no se haya enlace directo, se une al nodo
más
        %cercano.
        if(cont==size(pts,1))

matriz(ind_menor,menor)=dist(pts(ind_menor,:),pts(menor,:));
matriz(menor,ind_menor)=dist(pts(ind_menor,:),pts(menor,:));
        end

        end%for j
        cont=1;
        ind_menor=0;
        menor=0;
        aux=10000000;
end

```

CASO 2: Construcción del grafo usando palabras visuales para toda la imagen

angulo.m

```

function angl=angulo(a,b)
%Calculamos el ángulo entre dos vectores pasados por parámetros.

%Calculamos vertice del triángulo
if(a(1,')==b(1,:))
    v=a(1,:);
    u=a(2,:);
    w=b(2,:);
else if(a(1,')==b(2,:))
    v=a(1,:);
    u=a(2,:);
    w=b(1,:);
else
    v=a(2,:);
    u=a(1,:);
    w=b(2,:);
end
end

m1=(u(2)-v(2))/(u(1)-v(1));
a1=atan(m1)*180/pi;
if(a1==0)&&(u(2)==v(2))&&(u(1)<u(2))
    a1=180;
end
if(a1<0)
    a1=360+a1;
end
end

```

```

m2=(w(2)-v(2))/(w(1)-v(1));
a2=atan(m2)*180/pi;
if(a2==0)&&(w(2)==v(2))&&(w(1)<v(2))
    a2=180;
end
if(a2<0)
    a2=360+a2;
end
a1;
a2;
angl=abs(a2-a1);
if(angl>180)
    angl=360-angl;
end
if(isnan(angl)==1)
    angl=0;
end

```

difPalabras.m

```

function [dW,dM]=difPalabras(word1,word2)
%Se halla la distancia entre palabras visuals usando la distancia
%euclídea y la distancia de Mahalanobis.

d1=(word1(1)-word2(1))^2;
d2=(word1(2)-word2(2))^2;
d3=(word1(3)-word2(3))^2;
dW=sqrt(d1+d2+d3);

%Varianzas de las variables del DL
sigma1=180;%Angulo maximo
sigma2=30;%Longitud maxima
sigma3=256;%Intensidad maxima

dM1=((word1(1)-word2(1))/sigma1)^2;
dM2=((word1(2)-word2(2))/sigma2)^2;
dM3=((word1(3)-word2(3))/sigma3)^2;
dM=sqrt(dM1+dM2+dM3);

```

eliminarRep.m

```

function [words indices]=eliminarRep(DL);
% Devuelve vector con los DL no repetidos de un listado.
aux=zeros(1,size(DL,1));
words=zeros(1,3);
replicas=ones(1,size(DL,1))*-1;
size(DL,1);
for i=1:size(DL,1)
    if(aux(i)==0)
        aux(i)=i;
    end
end

```

```

end

for j=1:size(DL,1)
    if((i~=j) && (difPalabras(DL(i,:),DL(j,:))<0.03))%==0)
%Si la diferencia es mayor entonces es una palabra nueva y se
incorpora al vector.
        if(aux(j)==0)
            aux(j)=i;
            replicas(i)=j;
        end
    end
end%for j
%Si no
end%for i
%Palabras iguales tienen el mismo indice.

for i=1:size(DL,1)
    if(find(aux==i,1)~=NaN)%Coge la primera palabra para cada indice.
        words=[words;DL(i,:)];
    end
end
%
aux;
replicas=replicas(2:end);
indices=find(replicas>0);
words=words(2:end,:);
size(words);
cp=cputime;

```

esquinas.m

```

function DL=esquinas(pts,bw);
%Devuelve los descriptores local extraídos de los puntos
característicos de una imagen bw.

%esq=[fila; x's o y's si pones 1 o 2; coordenada del pto entera (1,2,3
uno de los tres)]
esq=zeros(1,2,3);
umbral=15;%15; %Distancia alrededor de 20pixels

%Se cogen todas los puntos que formen una esquina cuya distancia entre
%puntos sea inferior al umbral.
for i=1:size(pts,1)
    for j=1:size(pts,1)
        if(i~=j)
            %Primero buscamos los dos primeros puntos
            if (dist(pts(i,:),pts(j,:))<umbral)
                for k=1:size(pts,1)
                    if(i~=j)&&(i~=k)&&(k~=j)
                        if
                            (dist(pts(i,:),pts(k,:))<umbral)&&(dist(pts(j,:),pts(k,:))<umbral)
                                aux=zeros(1,2,3);
                                aux(1,:,1)=pts(i,:);

```

```

                                aux(1,:,2)=pts(j,:);
                                aux(1,:,3)=pts(k,:);
                                %Concatenamos a las esquinas
que ya
                                %hayamos obtenido previamente.
                                esq=[esq;aux];
                                end
                                end
                                end
                                end%for j
end

%Calculamos la información del descriptor local
%(ángulo,distancia,iluminación media,color medio)
%DL es el descriptor local
DL=zeros(length(esq)-1,3);
%quitamos la primera esquina que es cero
for i=2:length(esq)
    aux=zeros(1,3);
    %Sacamos rectas para calcular angulos
    a=[esq(i,:,1);esq(i,:,2)];
    b=[esq(i,:,2);esq(i,:,3)];
    DL(i-1,1)=angulo(a,b);
    DL(i-1,2)=longitudTotal(esq(i,:,1),esq(i,:,2),esq(i,:,3));
    DL(i-1,3)=iluminacionMedia(esq(i,:,1),esq(i,:,2),esq(i,:,3),bw);

end

%Posible normalización a cantidades enteras.
%DL=round(DL);

```

IluminacionMedia.m

```

function ilu=iluminacionMedia(u,v,w,image)
%Extrae la iluminación media del eventanado que almacena el triángulo.

centro=round(mean([u;v;w]));
l=0;
if(centro(1)>5 && centro(1)<251)
    if(centro(2)>5 && centro(2)<381)
        l=ventana(centro,image);
    end
end
ilu=l;

```

longitudTotal.m

```

function l=longitudTotal(u,v,w)
%Perimetro del triangulo
l=dist(u,v)+dist(u,w)+dist(v,w);

```

main2.m

```

close all;
clear all;
warning OFF;
images = dir;
num=length(images);
%Inicializamos listado de palabras
DL_TOT=[0 0 0];
matriz=0;
%Se analizan todas las imagenes dentro de la carpeta
for i=1:num
    nombre=getfield(images,{i,1},'name');
    t=length(nombre);
    if t>4
        term=nombre(t-3:end);%Comprobamos que sea una imagen
        if (term == '.jpg')
            a = imread(nombre);
            bw=rgb2gray(a);
            %Sacamos puntos invariantes de la cara que nos haran de
base del grafo.
            pts=pt_inv(a,a);
            DL=esquinas(pts,bw);
            [DL indices]=eliminarRep(DL)

            %Agregamos nueva imagen a matriz
            [DL_TOT matriz]=uniraMatriz(DL_TOT,DL,matriz);
        end
        end
        nombre
        i
        size(DL_TOT,1)
        pause;
    end
AVm=eig(matriz);
AVm=AVm';
AV=zeros(1,size(AVm));%Inicializamos con el mínimo.

[n,ejeX]=hist(AV,size(DL_TOT,1));
%Normalizamos
n=n./sum(n);%cont;
ejex=ejeX/max(ejeX);
figure;plot(ejex,n)
title(['Histograma de los Autovalores de: ',nombre]);
cputime

```

uniraMatriz.m

```

function [DTnueva matriz]=uniraMatriz(DL_TOT,DL_IMG,MAT);

%Cada imagen es como un parrafo. Las palabras en ese parrafo están

```

```

%%relacionadas entre si. Luego una nueva imagen se une al grafo. No
tiene
%%pq quedar conexo.
%Fijamos el punto de inicio de las palabras de la nueva imagen
inicio=size(MAT,1)+1;
%Se extraen las nuevas palabras no repetidas anteriormente.
[words indices]=eliminarRep([DL_TOT;DL_IMG]);

num_nuevas=abs(size(DL_TOT,1)-size(words,1));
%Añadimos a la matriz las nuevas filas de las nuevas palabras y
copiamos la
%matriz anterior.
new_mat=zeros(size(words,1));
new_mat(1:length(MAT),1:length(MAT))=MAT;

for i=0:size(new_mat,1)
    if (find(indices==i)~=NaN)
        new_mat(i,inicio:end)=1;
        new_mat(inicio:end,i)=1;
    end
end
%Rellenamos los enlaces nuevos relacionados entre sí por pertenecer a
la
%nueva imagen.
new_mat(inicio:end,inicio:end)=not(eye(num_nuevas));
matriz=new_mat;
DTnueva=words;

```

ventana.m

```

function l=ventana(c,image)

v=zeros(5,5);
for i=1:5
    for j=1:5
        if((c(1)+i-3<384)&&(c(2)+j-3<256))
            v(i,j)=image(c(1)+i-3,c(2)+j-3);
        else
            v(i,j)=0;
        end
    end
end

%Media de la ventana en iluminación.
l=mean(mean(v));

```


CASO 3: Construcción del grafo usando palabras visuales con la imagen segmentada en celdas

Main3.m

```

close all;
clear all;
warning OFF;
images = dir;
num=length(images);
%Inicializamos listado de palabras
DL_TOT=[0 0 0];
matriz=0;
frasesFila=4; %Frases por Fila
frasesCol=4; %Frases por Columna
celdaF=96;%Tamaño reduccion de imagen en pixels
celdaC=64;
Fmin=0;
Fmax=0;
Cmin=0;
Cmax=0;
%Total 96 frases
%Se analizan todas las imagenes dentro de la carpeta
for i=1:num
    nombre=getfield(images,{i,1},'name');
    t=length(nombre);
    if t>4
        term=nombre(t-3:end);%Comprobamos que sea una imagen
        if (term == '.jpg')
            a = imread(nombre);
            bwG=rgb2gray(a);
            %Sacamos puntos invariantes de la cara que nos haran de
            base del grafo.
            pts=pt_inv(a,a);
            bw=rgb2gray(a);%imagen que se irá segmentando en
            celdas
            imgT=bw;%imagen al completo
            for k=1:frasesCol
                for j=1:frasesFila
                    ptsImg=[0 0];
%Segmentando por celdas del tamaño definido en el inicio del script.
                    for m=1:size(pts)
                        if ( (pts(m,1)>(k-
1)*celdaF+1)&&(pts(m,1)<(k*celdaF))&&(pts(m,2)>(j-
1)*celdaC+1)&&(pts(m,2)<(j*celdaC)) )
                            ptsImg=[ptsImg;pts(m,:)];
                        end
                    end%For coge los puntos de la celda
                    Fmin=(k-1)*celdaF+1;
                    Fmax=k*celdaF;
                    Cmin=(j-1)*celdaC+1;
                end
            end
        end
    end
end

```

```

Cmax=j*celdaC;

bw=rgb2gray(a((k-
1)*celdaF+1):(k*celdaF),((j-1)*celdaC+1):(j*celdaC),:));

%           size(ptsImg)
%           ptsImg

if(size(ptsImg,1)>3)

DL=esquinas(ptsImg,bw,Fmin,Fmax,Cmin,Cmax,imgT);

[DL indices]=eliminarRep(DL);
[DL_TOT
matriz]=uniraMatriz(DL_TOT,DL,matriz);

end%if de número pts minimo para
esquinas

end%columnas
end%filas
end% if .jpg

end%if t>4

% nombre
% i
% size(DL_TOT,1)
end%For de las imagenes
AVm=eig(matriz);
AVm=AVm';
AV=zeros(1,size(AVm));%Inicializamos con el mínimo.

[n,ejeX]=hist(AV,size(DL_TOT,1));
%Normalizamos
n=n./sum(n);%cont;
ejex=ejeX/max(ejeX);
figure;plot(ejex,n)
title(['Histograma de los Autovalores de: ',nombre]);
cputime

```

reducirImagen.m

```

function reducirImagen
    celda=32;
    bwG=rgb2gray(a);
    %Sacamos puntos invariantes de la cara que nos haran de base
del grafo.

```

```
for k=1:(size(bwG,1)/celda)
    for j=1:(size(bwG,2)/celda)
        im=a(((k-1)*celda+1):(k*celda),((j-1)*celda+1):(j*celda),:);
        pts=pt_inv(im,im);
        bw=rgb2gray(im);
        DL=esquinas(pts,bw)
    end
end
end
```