



Reconocimiento de rostros con *elastic bunch graph matching* y *constrained local neural fields* para vídeo vigilancia

José Rodrigo Galdos Chávez

Orientador: Dr. Juan Carlos Gutiérrez Cáceres

Jurado:

Dr. Guillermo Cámara Chávez – Universidade Federal de Ouro Preto – Brasil

Dr. Jorge Poco – University of Washington – USA

Dr. Erick Gomez Nieto – Universidade de Sao Paulo – Brasil

Dr. Alex Cuadros Vargas – Universidad Católica San Pablo – Perú

*Tesis presentada al
Centro de Investigación e Innovación en Ciencia de la Computación (RICCS)
como parte de los requisitos para obtener el grado de
Maestro en Ciencia de la Computación.*

Universidad Católica San Pablo – UCSP
Septiembre de 2017 – Arequipa – Perú

*A Dios y a mi Madre por todo lo que
me han dado.*

Abreviaturas

LDA *Linear Discriminant Analysis*

PCA *Principal Component Analysis*

EBGM *Elastic Bunch Graph Matching*

CCTV *Closed-Circuit Television*

KFA *Kernel Fisher Analisys*

CLNF *Constrained Local Neural Field*

CLM *Constrained Local Model*

HOG *Histogram of Oriented Gradients*

SVM *Support Vector Machine*

PDBNN *Probabilistic Decision-Based Neural Network*

RBF *Radial Basis Function*

Agradecimientos

Deseo agradecer de manera especial al Consejo Nacional de Ciencia, Tecnología e Innovación Tecnológica (CONCYTEC) y al Fondo Nacional de Desarrollo Científico, Tecnológico e Innovación Tecnológica (FONDECYT-CIENCIACTIVA), que mediante Convenio de Gestión UCSP-FONDECYT N° 011-2013, han permitido la subvención y financiamiento de mis estudios de Maestría en Ciencia de la Computación en la Universidad Católica San Pablo (UCSP).

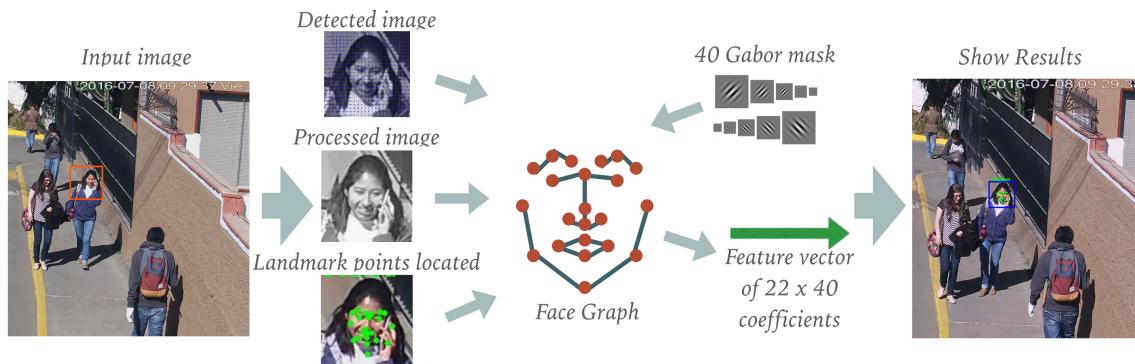
Agradezco de forma muy especial a mi orientador Dr. Juan Carlos Gutiérrez Cáceres por haberme guiado en esta tesis.

Deseo agradecer de forma especial al Dr. Alex Cuadros por sus consejos y por alentarme en el transcurso de mis estudios, también deseo agradecer al Dr. Jorge Poco por sus observaciones y ayuda a lo largo de la investigación de esta tesis.

Deseo agradecer al Sr. Alejandro Camargo por su consejos en la redacción de este documento de tesis.

Finalmente deseo agradecer al personal administrativo de la universidad: Guillermo Laures, Carlos Begazo y Katherine Ayala. Muchas gracias por la atención brindada y por su disposición para aydarme en los distintos momentos a lo largo de estos años.

Abstract



Face recognition is an area with a large number of applications and techniques. Many of these techniques provide good results when they are applied to situations where the environment in which recognition is desired is controlled, this is understood as the control of the factors that influence the recognition process, such as illumination, face pose , facial expression, etc.

But for the case of uncontrolled environments, such as video surveillance, face recognition still presents difficulties: variation in illumination, lack of collaboration of people to recognize, among several others. Because of the importance it has in security and the amount of existing infrastructure, it is necessary to apply face recognition to video surveillance.

To address the above-mentioned problems, a face-recognition pipeline is proposed using *Elastic Bunch Graph Matching* (**EBGM**) with *Constrained Local Neural Field* (**CLNF**) as a replacement for the point detection function of the original algorithm to finally be applied in video. In this thesis, a parametric analysis of **EBGM** was performed to find the most influential factor in its performance along with its comparison with other methods of face recognition. It was also determined elements that are part of the pipeline presented as final result.

Finally, the proposal was tested in a database created from shots of a security camera, consisting of 24 subjects with 8 images each. The final

results show an improvement of 10 % and 6 % in images taken in the morning and at noon respectively.

Keywords: Face recognition, Video surveillance, Uncontrolled environment, *Elastic Bunch Graph Matching* ([EBGM](#)).

Resumen

El reconocimiento de rostros es un área con una gran cantidad de aplicaciones y técnicas. Muchas de esas técnicas ofrecen buenos resultados cuando se aplican a situaciones donde el ambiente en el cual se desea realizar el reconocimiento es controlado, esto se entiende como el control de los factores que influyen en el proceso de reconocimiento, tales como iluminación, pose del rostro, expresión facial, etc.

Pero para el caso de ambientes no controlados, como lo es la video vigilancia, el reconocimiento de rostros aún presenta dificultades: variación en la iluminación, falta de colaboración de las personas a reconocer, entre varios otros. Debido a la importancia que tiene en seguridad y a la cantidad de infraestructura existente, es necesario aplicar el reconocimiento de rostros a video vigilancia.

Para afrontar los problemas mencionados, se propone un *pipeline* de reconocimiento de rostros usando **EBGM** con **CLNF** como reemplazo a la función de detección de puntos del algoritmo original, para finalmente ser aplicado en vídeo. En este trabajo de tesis se realizó un análisis paramétrico de **EBGM** para encontrar el factor mas influyente en su rendimiento, junto con su comparación con otros métodos de reconocimiento de rostros. También se determinó que elementos forman parte del *pipeline* presentado como resultado final.

Finalmente la propuesta fue probada en una base de datos creada a partir de tomas de una cámara de seguridad, que consta de 24 sujetos con 8 imágenes cada uno. Los resultados finales muestra una mejora de 10 % y 6 % en imágenes tomadas en la mañana y en el medio día respectivamente.

Palabras clave: Reconocimiento de rostros, Vídeo vigilancia, Ambiente no controlado, *Elastic Bunch Graph Matching* (**EBGM**).

Índice general

1. Introducción y contexto	1
1.1. Planteamiento del problema	3
1.2. Objetivos	7
1.2.1. Objetivo general	7
1.2.2. Objetivos específicos	7
1.3. Alcance y límites de la tesis	7
1.4. Organización de la tesis	8
2. Revisión Bibliográfica	11
2.1. Métodos holísticos	12
2.1.1. <i>Principal Component Analysis (PCA)</i>	12
2.1.2. <i>Kernel Fisher Analisys (KFA)</i>	14
2.1.3. <i>Linear Discriminant Analysis (LDA)</i>	14
2.2. Métodos basados en modelos y características	15
2.2.1. 3-D Morphable Model	15
2.3. Métodos basados en redes neuronales	15
2.4. Consideraciones finales	16
3. Marco teórico de la propuesta	17
3.1. Métodos de detección	17
3.1.1. Detector de Viola-Jones	17

ÍNDICE GENERAL

3.1.2. Detector de <i>Histogram of Oriented Gradients</i> (HOG)	19
3.1.3. <i>Support Vector Machine</i> (SVM)	20
3.1.4. <i>Constrained Local Neural Field</i> (CLNF)	20
3.2. Técnicas de pre-procesamiento	21
3.2.1. Ecualización de histograma	21
3.2.2. Transformada de logaritmo	22
3.2.3. Transformada discreta de coseno	22
3.3. <i>Elastic Bunch Graph Matching</i> (EBGM)	23
3.3.1. Gabor Wavelet	24
3.3.2. Localización de puntos fiduciales	25
3.3.3. Face Graph	28
3.3.4. Función de similitud	28
3.4. Consideraciones finales	29
4. Propuesta	31
4.1. Esquema general de la propuesta	31
4.2. Detección y validación de rostros	34
4.2.1. Detector de Viola-Jones	34
4.2.2. Validación con detector de HOG	35
4.3. Detector de puntos fiduciales	36
4.4. Pre-procesamiento y normalización de imágenes	37
4.5. Filtro y reajuste de puntos detectados	38
4.6. Proceso de EBGM	39
4.7. Consideraciones finales	40
5. Resultados	41
5.1. Base de datos	41

ÍNDICE GENERAL

5.1.1. AT&T	41
5.1.2. Yale A	42
5.1.3. Georgia Tech	43
5.1.4. Base de datos generadas a partir de imágenes de cámara de vigilancia	43
5.2. Método de experimentación	43
5.3. Comparación de EBGM con métodos holísticos	44
5.4. Evaluación paramétrica de EBGM	45
5.4.1. Incrementar la cantidad de modelos	46
5.4.2. Modificar la función de similitud	47
5.4.3. Incrementar la cantidad de imágenes de entrenamiento a través de transformaciones de perspectiva	48
5.4.4. Ajuste de tamaño a máscaras de Gabor	50
5.5. Evaluaciones en el <i>pipeline</i> de reconocimiento	51
5.5.1. Evaluación del detector de Viola-Jones para detección de ojos	51
5.5.2. Evaluar validación de detecciones mediante color de piel	52
5.5.3. Mejora de la iluminación en el pre-procesamiento	53
5.5.4. Tracking de rostros	54
5.5.5. Evaluar la detección de puntos fiduciales	55
5.6. Prueba con base de datos de vídeo vigilancia	57
5.7. Uso de la propuesta para reconocer imágenes ofuscadas	59
5.7.1. Blurring	59
5.7.2. Comparación de redes neuronales	59
5.8. Consideraciones Finales	60
6. Conclusiones, publicaciones y trabajos futuros	63
6.1. Conclusiones	63
6.2. Trabajos futuros	64

ÍNDICE GENERAL

6.3. Publicaciones	64
Bibliografía	68

Índice de cuadros

1.1. Aplicaciones típicas del reconocimiento de rostros según Zhao Zhao et al. (2003)	2
3.1. Lista de puntos fiduciales presentada en Bolme (2003)	26
5.1. Comparación con bases de datos ATT, Yale A y Georgia	45
5.2. Resultado aciertos con incremento de modelos en EBGM	46
5.3. Tabla de distribución de pesos propuesta	48
5.4. Valores del vector W de pesos derivado del Cuadro 5.3	49
5.5. Resultados de adición de pesos EBGM	50
5.6. Resultados de pruebas en tamaños de mascaras de Gabor	50
5.7. Resultados de la propuesta de iluminación	54
5.8. Comparación de puntos fiduciales	56
5.9. Resultado de uso de CLNF como detector de puntos fiduciales para CLNF	57
5.10. Resultados de la propuesta en comparación a EBGM sin cambios . .	58
5.11. Comparación entre propuesta y método propuesto en McPherson et al. (2016)	60

ÍNDICE DE CUADROS

Índice de figuras

1.1. Ejemplos de la variación de la iluminación en un rostro. Extraídos de la base de datos "Yale A".	4
1.2. Ejemplos de como el punto de visión modifica un rostro. Extraídos de la base de datos "FEI".	4
1.3. Ejemplos de la variación de la expresión en un rostro. Extraídos de la base de datos "Yale A".	5
1.4. Ejemplos de la variación del tiempo en un mismo sujeto. Extraídos de la base de datos "Georgia".	5
1.5. Ejemplos extraídos del proyecto de "CV Dazzle" Harvey (2014).	5
1.6. Un ejemplo clásico de la oclusión del rostro es el uso de lentes de sol. Extraídos de la base de datos "Yale A".	6
2.1. Clasificación de métodos de reconocimiento de rostro por enfoque según Zhao et al. (2003).	12
3.1. Ejemplo de característica Haar, extraído de Viola and Jones (2001). Estos rectángulos pueden ser entendidos como rectángulos en los cuales son determinados como áreas de baja iluminación y otros como áreas de alta iluminación de esta manera podemos representar el contraste entre dos secciones de la imagen,	18
3.2. Muestra del clasificador en cascada, donde una imagen es analizada por un clasificador tras otro desde el primero hasta el n-esimo si resulta la imagen positiva, extraído de Viola and Jones (2001).	19
3.3. Ecualización de histograma - Izquierda: imagen en escala de Grises. Derecha: imagen ecualizada.	22
3.4. Transformada de logaritmo - Izquierda: imagen en escala de Grises. Derecha: imagen con Transformada de logaritmo.	22

ÍNDICE DE FIGURAS

3.5. Muestra de los puntos fiduciales elegidos en EBGM ^{Bolme (2003)} , en cada punto se realiza 40 convoluciones con máscaras de Gabor para crear un Gabor Jet por punto	26
3.6. Representación de Face Graph presentada en Wiskott et al. (1997)	28
4.1. <i>Pipeline</i> de propuesta, muestra el proceso de reconocimiento de rostros desde la detección en una escena hasta la muestra del resultado.	32
4.2. Muestra de una imagen optima detector de Viola-Jones, todos los sujetos observan a la cámara y es una imagen en buena calidad, extraída de Internet	34
4.3. Muestra de una escena de vídeo vigilancia usando el detector de Viola-Jones	35
4.4. Ejemplo de detecciones usando el detector de Viola-Jones, donde la ultima imagen es un falso positivo	35
4.5. Muestra de representación del descriptor de HOG(derecha) en una imagen de rostro(izquierda), donde el total de la imagen es transformada a un vector de características.	36
4.6. Muestra del pre-procesamiento aplicado a imágenes, comparando antes y después	37
4.7. Puntos fiduciales entregados por CLNF, extraído del proyecto “Open Face”	39
4.8. Puntos fiduciales después del proceso de filtrado	39
5.1. Muestra de imágenes que conforman la base de datos AT&T	42
5.2. Muestra de imágenes que conforman la base de datos Yale A	42
5.3. Muestra de imágenes que conforman la base de datos Georgia Tech . .	43
5.4. Muestra de imágenes que conforman la base de datos obtenida a partir de una cámara de seguridad	44
5.5. Muestra del proceso de experimentación, donde las imágenes se dividen por grupos y todos los grupos rotan hasta que todas las combinaciones de prueba y entrenamiento sean probadas	45
5.6. Comparación de EBGM con otros algoritmos, la escala empieza en 68,00 para que pueda apreciarse las diferencias entre los métodos	46
5.7. Curso de procesos que forman parte del reconocimiento de rostros en EBGM	47

ÍNDICE DE FIGURAS

5.8. Comparación entre las configuraciones propuestas para los pesos en la función de similitud de EBGM	48
5.9. Imagen de entrenamiento original	50
5.10. Resultados de las transformaciones de perspectiva	51
5.11. Ejemplo de proceso de detección de ojos Nro 1	51
5.12. Ejemplo de proceso de detección de ojos Nro 2	52
5.13. Ejemplo de falsas validaciones en colores cercanos al rojo y naranja. . .	53
5.14. Comparación entre propuesta, ecualización de histograma y EBGM . .	54
5.15. Ejemplo de tracking usando Camshift	55
5.16. Ejemplo de como el área de tracking se expande debido a áreas circundantes del mismo color	56
5.17. Resultado de pruebas de puntos fiduciales, donde CLNF con 22 es la propuesta final y CLNF con 68 es el resultado sin un filtrado de puntos	58
5.18. Escena de captura de imágenes para pruebas	59
5.19. Imagen antes y después de un proceso de <i>blurring</i>	60

Capítulo 1

Introducción y contexto

Visión computacional es el conjunto de métodos y técnicas por las cuales se intenta comprender y analizar las imágenes del mundo real mediante el uso de una computadora. Representa un problema bastante difícil, en parte porque es un problema inverso en el cual intentamos recuperar algunas incógnitas de cómo llegar a una solución específica, esto quiere decir que sabemos cuál es la solución, por ejemplo reconocer un auto o segmentar una figura, pero no conocemos la totalidad del proceso por el cual hemos llegado a dicha solución [Szeliski \(2010\)](#).

Según Szeliski [Szeliski \(2010\)](#) las áreas más comunes de visión computacional son:

- Procesamiento de imágenes.
- Detección de características.
- Segmentación.
- Alineamiento en base a características.
- Estimación del movimiento.
- Fusión de imágenes.
- Fotografía computacional.
- Estéreo correspondencia.
- Modelamiento de formas 3D.
- *Rendering* basado en imágenes.
- Reconocimiento.

Donde el reconocimiento es la tarea de encontrar e identificar objetos en particular en una imagen o secuencia de vídeo. Mientras los seres humanos podemos reconocer varios objetos en una imagen sin esfuerzo. Replicar este proceso en una maquina con toda su versatilidad sigue siendo un gran desafío. Entre las varias formas de reconocimiento existe el reconocimiento de rostros que se presenta de dos maneras, una es verificación y la otra es identificación [Alice \(2003\)](#).

Mientras la verificación intenta contestar sí una imagen de un rostro pertenece una persona en particular, la identificación intenta contestar la pregunta ¿ a quién le pertenece la imagen del rostro? siendo un proceso más complejo que la verificación debido a que mientras uno es una comparación uno a uno entre la imagen a examinar y el rostro del sujeto a verificar, el otro es una comparación de uno a varios entre la imagen del rostro y el conjunto de rostros que se tengan para identificar teniendo en cuenta la posibilidad de que el rostro a identificar no pertenezca al conjunto de rostros con los que se realiza la comparación.

Cuadro 1.1: Aplicaciones típicas del reconocimiento de rostros según Zhao [Zhao et al. \(2003\)](#)

Áreas	Aplicaciones específicas
Entretenimiento	Vídeo juegos, realidad virtual, programas de entrenamiento
	Interacciones robot-humano, Interacciones humano-computadora
Tarjetas inteligentes	Licencias de conducir, Programas de autorización
	Inmigración, Documento de identificación, Pasaportes, registro de votantes
	Seguro contra fraude
Seguridad de la información	TV de control parental, Ingreso a dispositivos personales, Ingreso a desktop
	Aplicaciones de seguridad, Seguridad de base de datos, encriptación de archivos
	Seguridad de Intranet, Acceso a Internet, Registros médicos
	Seguridad de terminales de comercio
Aplicación de la ley y vigilancia	Vídeo vigilancia avanzada, control de CCTV
	Control de Portales, Análisis Post evento
	Robo de tiendas, Investigación y seguimiento de sospechosos

Según el *survey* de Zhao [Zhao et al. \(2003\)](#), entre las varias aplicaciones del reconocimiento de rostros, Cuadro 1.1, se menciona su aplicación en video vigilancia y *Closed-Circuit Television (CCTV)*. En estas aplicaciones usualmente el proceso de reconocimiento es llevado a cabo por seres humanos, esto implica que la calidad del trabajo no será la misma en todo momento, y aumenta la posibilidad de un error por factor humano. Se explica en mayor detalle este problema y la necesidad de automatizar este proceso en la Sección 1.1.

El uso de cámaras para monitorear lugares de interés no es nuevo, de hecho se ha realizado durante décadas, este arreglo entre cámaras de seguridad y la infraestructura necesaria para su funcionamiento y visualización de las imágenes se conoce como **CCTV**.

Inicialmente su uso estaba reservado para complejos industriales debido al alto coste de los equipos que se necesitaba, en especial las cámaras. Pero hoy la situación es contraria, ahora se usa desde espacios públicos hasta edificios privados y hogares. Su difusión ha ido aumentando con el paso del tiempo por el menor costo del *hardware* y una mayor preocupación por un control de seguridad. Esta mayor preocupación motiva

la investigación de técnicas que permita identificar a las personas sin su cooperación.

Existen muchas iniciativas para el uso del reconocimiento de rostros en ambientes no controlados, esto quiere decir que no se tiene control sobre las condiciones de iluminación, angulo de visión, occlusiones e incluso la colaboración de los sujetos a reconocer entre otros factores.

En trabajos como el de Gorodnichy [Gorodnichy et al. \(2014\)](#) para el gobierno de Canadá, se puede ver la necesidad de contar con un sistema fiable de vídeo vigilancia en este caso para puestos fronterizos. En [Tian et al. \(2005\)](#) se presenta un método para el análisis de fondo de una escena de vídeo proporcionado por una cámara de vigilancia estática, y en [Nazare et al. \(2014\)](#) se presenta una propuesta de *framework* para la unión de varios procesos de visión computacional agrupándolos en módulos: reconocimiento de áreas de interés, seguimiento e identificación y extracción de conocimiento, todos ellos aplicados a vídeo vigilancia.

El uso visión computacional en vídeo vigilancia no solo se limita al ámbito académico, existen varias situaciones donde la necesidad de seguridad ha creado la infraestructura necesaria para que investigaciones como las citadas anteriormente tengan un uso práctico. Un ejemplo es [Odobeza et al. \(2012\)](#) que es la demostración del prototipo de un sistema de vídeo vigilancia para un metro. Todo este incremento también ha despertado un gran interés sobre el alcance de la vídeo vigilancia como en el caso de la ciudad de Londres, donde [Wood et al. \(2006\)](#) muestra la gran cantidad de cámaras instaladas en la ciudad y habla sobre los posibles problemas concernientes a la privacidad en una sociedad donde la vigilancia en pro de la seguridad es una prioridad, un caso similar es el de la ciudad de Chicago expuesto en [Schwartz \(2012\)](#).

Se ha mencionado el contexto de esta tesis que es el reconocimiento de rostros en vídeo vigilancia en ambientes no controlados, que nace de la necesidad de un control de seguridad automatizado y también debido a la oportunidad generada por la existencia de una creciente infraestructura en cámaras de vigilancia junto al hecho que los trabajos de visión computacional son lo suficientemente maduros como para ponerse en práctica.

1.1. Planteamiento del problema

El problema de reconocimiento de rostro en vídeo vigilancia aun no ha sido resuelto en visión computacional e inclusive resulta un problema para el ser humano como se explica en [Burton et al. \(1999\)](#) donde se prueba que las personas tienen problemas para reconocer sujeto con los que no son familiares en videos de baja resolución y también en vídeos de alta calidad [Burton et al. \(1999\)](#). Este problema es el resultado de varios factores que pueden ser agrupados en problemas relacionados con el ambiente no controlado.

En el contexto actual del reconocimiento de rostros se puede mencionar algunos problemas que han sido objeto de investigación, muchos de ellos han sido resueltos

cuando se aplican a situaciones y condiciones determinadas.

Listamos a continuación los problemas más relevantes según Gross [Gross et al. \(2001\)](#):

1. *Efectos de la iluminación* [Kalocsai et al. \(1998\)](#) [Johnston et al. \(1992\)](#) [Bruce et al. \(1998\)](#) [Hill and Bruce \(1996\)](#).- Debido a que un rostro es un objeto tridimensional, las variaciones en las fuentes de iluminación pueden provocar alteraciones en la captura de la imagen, esto también es valido para fuentes de luz indirecta, ello puede causar sombras que acentúan o disminuyen ciertas características faciales. Un ejemplo es la cantidad inherente de luz reflejada en la piel, los ajustes de captura dentro de la cámara, etc. Todo ello ha de afectar la calidad de la imagen y/o vídeo. Un ejemplo se puede observar en la Figura 1.1



Figura 1.1: Ejemplos de la variación de la iluminación en un rostro. Extraídos de la base de datos "Yale A".

2. *Variación según el punto de visión* [Hill et al. \(1997\)](#).- El rostro es un objeto tridimensional y debido al ángulo de visión de la cámara la apariencia del rostro puede variar debido a la deformación proyectiva, la cual lleva a que la parte inferior de rostro se vea más angosta de lo que realmente es. Esto también se aplica para las rotaciones del rostro como se puede ver en la Figura 1.2



Figura 1.2: Ejemplos de como el punto de visión modifica un rostro. Extraídos de la base de datos "FEI".

3. *Expresión*.- El rostro es un objeto no rígido, la expresión de la emociones es parte de la comunicación paralingüística junto con el habla, y la variación de esas expresiones faciales influye en el reconocimiento del rostro. Ejemplos de ello se puede ver en la Figura 1.3



Figura 1.3: Ejemplos de la variación de la expresión en un rostro. Extraídos de la base de datos "Yale A".

4. *Time delay*.- Es el intervalo de tiempo entre la adquisición de la imagen que se desea reconocer y la imagen o los datos con los cuales realizamos la identificación. Por ejemplo, intentar identificar a un sujeto el cual su ultima fotografía fue tomada hace un año.



Figura 1.4: Ejemplos de la variación del tiempo en un mismo sujeto. Extraídos de la base de datos "Georgia".

5. *Factores individuales*.- Se considera factores individuales a aquellos como el uso de maquillaje, estilos de peinado, disfraces, y otras formas en las que las personas modifican su apariencia a gusto personal. Un ejemplo de ello es el proyecto de "CV Dazzle" [Harvey \(2014\)](#) que usa el maquillaje para burlar las técnicas de detección de rostros como se puede ver en la Figura 1.5.

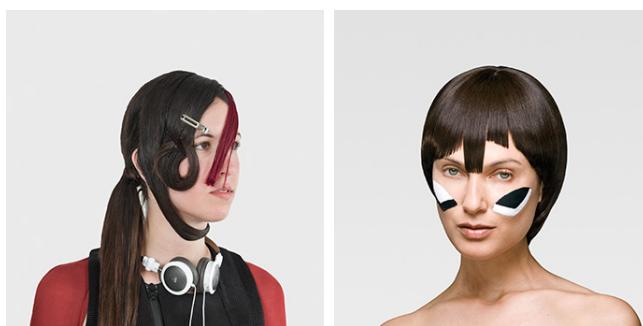


Figura 1.5: Ejemplos extraídos del proyecto de "CV Dazzle" [Harvey \(2014\)](#).

6. *Oclusión*.- La oclusión puede darse debido a objetos en la escena o al uso de lentes de sol y otros objetos, también debido a la falta de cooperación de los individuos y a la variación según el punto de visión, Figura 1.6.



Figura 1.6: Un ejemplo clásico de la oclusión del rostro es el uso de lentes de sol. Extraídos de la base de datos "Yale A".

A parte de los problemas listados, cuando el reconocimiento de rostros se aplica a vídeo vigilancia surgen nuevas dificultades siendo las mas importantes:

1. *La calidad del video.*- Debido a que la grabación de vídeo ocurre en ambientes no controlados todos los problemas anteriormente citados se presentan varias veces afectando la calidad final del vídeo. Cabe mencionar que a pesar de las mejora en resolución en las cámaras de vídeo vigilancia, frecuentemente se opta por equipos de gama baja en hogares y oficinas más como medida de disuasión que con el propósito de hacer vídeo vigilancia.
2. *Falta de cooperación de los individuos.*- Es un problema que solo se presenta en vídeo vigilancia, debido a que en otros escenarios de reconocimiento de rostros como los presentados en la Tabla 1.1 siempre se puede esperar que las personas cooperen para ser identificadas. Esto resulta ser uno de los motivos por el cual el reconocimiento de rostros es un problema de difícil resolución.
3. *Las imágenes de los rostros son pequeños.*- Debido a las condiciones de adquisición, las imágenes de los rostro son más pequeñas y de menor calidad de lo que asumen la mayoría de los sistemas de reconocimiento de rostros existentes. Por ejemplo, una región de rostros valida puede ser tan pequeña como 20×20 pixeles, mientras los tamaños de las imágenes de los rostros usados en los sistemas actuales son tan grandes como 128×128 . El pequeño tamaño no solo hace la tarea de reconocimiento difícil, también afecta la precisión de la detección de puntos de interés que a menudo son necesarios en los métodos de reconocimiento.
4. *Las características de los rostros como objetos Zhao et al. (2003).*- El rostro humano como objeto es más fácil de reconocer si se compara con otro objeto que no sea un rostro (diferencia Inter-clase) pero resulta mas difícil de reconocer si se le compara con otro rostro (diferencia Intra-clase) por ello detectar y localizar rostros es más fácil que reconocer un rostro en específico.

Como se ha podido observar en esta sección el reconocimiento de rostros en vídeo vigilancia es una tarea muy difícil debido a la conjunción de varios factores, como el ambiente no controlado donde aparecen todas las dificultades relacionadas al típico reconocimiento de rostros. A ello se suma los problemas del escenario de video vigilancia en el cual se puede esperar la falta de cooperación de los sujetos a identificar y finalmente la calidad de los videos y las imágenes de los rostros.

Como parte de la propuesta expuesta en detalle en el Capítulo 4 intentamos resolver el problema de reconocimiento de rostros en un *pipeline* de video vigilancia bajo las condiciones de una cámara de seguridad en alta definición, proponemos enfrentar las variaciones de iluminación a través de técnicas de pre-procesamiento, afrontar en parte el problema de variación de poses con el uso de los últimos avances en el *Constrained Local Neural Field* (**CLNF**) y *Elastic Bunch Graph Matching* (**EBGM**) como método de reconocimiento biométrico.

1.2. Objetivos

1.2.1. Objetivo general

Proponer un *pipeline* para el reconocimiento de rostros en una escena de vídeo con la adaptación de *Constrained Local Neural Field* (**CLNF**) y *Elastic Bunch Graph Matching* (**EBGM**) para su uso final en vídeo vigilancia.

1.2.2. Objetivos específicos

Como objetivos específicos se tiene:

- Comparar los diferentes métodos holísticos existentes contra **EBGM**.
- Analizar el impacto de la modificación de parámetros de **EBGM**.
- Analizar la influencia de incrementar el conjunto de entrenamiento mediante transformaciones de perspectiva en **EBGM**.
- Desarrollar los métodos de mejora del *pipeline* propuesto con **EBGM**.
- Probar el *pipeline* propuesto en imágenes de vídeo vigilancia.

1.3. Alcance y límites de la tesis

A partir de los problemas mencionados en la Sección 1.1 la propuesta, explicada en el Capítulo 4 enfrenta los siguientes problemas:

- Iluminación.- Con el uso del pre-procesamiento explicado en las Sección 4.4 que permite iluminar u oscurecer imágenes según la situación lo requiera, de esta manera lograr una normalización el lo referente a iluminación cuando existe luz diurna.

- Variación según el punto de visión, Expresión facial.- Estos dos problemas son enfrentados con el uso de **CLNF** como detector de puntos ya que como se muestra en el Capítulo 5 mejora el rendimiento del reconocimiento. debido a su mayor exactitud en encontrar puntos fiduciales independientemente de las expresiones faciales.
- Falta de cooperación de los individuos.- Es enfrentado mediante la unión del detector de Viola-Jones, detector de **HOG**. También entra en consideración la posición de la cámara que fue puesta de formal tal que pueda captar los rostros de los transeúntes, Figura 4.3.

Problemas como baja calidad del vídeo, imágenes de rostros muy pequeñas no son enfrentados con la propuesta, mas allá del pre-procesamiento y transformación de tamaño propio de **EBGM** ya que no se emplea ninguna técnica de súper-resolutorio o mejora de imagen. Para el caso de oclusión y factores individuales no se propone ninguna mejora al funcionamiento de los detectores usados mas que su combinación para validar rostros y tampoco se prueba la propuesta cuando las imágenes de vídeo vigilancia son de tomas de luz infrarroja.

1.4. Organización de la tesis

En los próximos capítulos se desarrollará el trabajo que comprende toda la tesis, en esta sección se realiza una breve descripción de cada uno de ellos.

1. En el Capítulo 2 se desarrolla una explicación del reconocimiento de rostros, una categorización de los métodos de reconocimiento de rostros, establecemos las diferencias entre métodos basados en características y métodos holísticos, donde se presenta el estado del arte en el reconocimiento de rostros.
2. En el Capítulo 3 se expone el marco teórico de nuestra propuesta. Se realiza una introducción a las técnicas de reconocimiento probadas en esta tesis, se profundiza el funcionamiento de **EBGM** y se explica las razones de su uso para este trabajo. Definimos el concepto de biométria y cuales son los requerimientos para que un método de reconocimiento sea considerado biométrico.
3. En el Capítulo 4 se describe en detalle la propuesta de esta tesis, concentrándose en el *pipeline* de vídeo desde la detección de rostros hasta el resultados del reconocimiento.

Se menciona el aporte de la modificación de **EBGM** usando **CLNF** para su adaptación al uso de vídeo vigilancia, junto al uso de mejoras en la iluminación.

4. En el Capítulo 5 se muestran los resultados experimentales de la propuesta junto con las opciones exploradas para lograr el objetivo del trabajo de tesis.

5. Finalmente en el Capítulo 6 se mencionan las conclusiones obtenidas en esta tesis y se listan los posibles trabajos futuros en continuación a este trabajo.

Se ha expuesto el contexto y la necesidad de contar con propuestas para afrontar el trabajo de video vigilancia en este trabajo, es necesario seguir resaltando su dificultad no solo para los algoritmos de visión computacional, sino para las personas en diversas situaciones.

Capítulo 2

Revisión Bibliográfica

El reconocimiento de individuos usando el rostros se realiza a partir de imágenes, las cuales pueden ser estáticas (fotografías) o dinámicas (imágenes de vídeo). En el caso de la video vigilancia se toma en cuenta el hecho que no se tiene control sobre varios factores que influye en proceso de reconocimiento como se explica en la sección 1.1.

Según *surveys* de la literatura como Zhao [Zhao et al. \(2003\)](#), Parmar [Parmar and Mehta \(2014\)](#) y Pandya [Pandya et al. \(2013\)](#) los métodos de reconocimiento de rostros se pueden categorizarse mediante el enfoque en por el cual abordan el problema, es decir como tratan a la imagen del rostro. En la figura 2.1 se puede ver algunos algoritmos clasificados por enfoque. Siendo algunos de los más conocidos *Linear Discriminant Analysis (LDA)* [Zhao et al. \(1999\)](#), *Principal Component Analysis (PCA)* [Turk and Pentland \(1991\)](#) , *Elastic Bunch Graph Matching (EBGM)* [Wiskott et al. \(1997\)](#).

Como se observa en la Figura 2.1 los métodos de reconocimiento de rostros pueden ser agrupados en métodos holísticos y métodos basados en características como en [Tseng \(2003\)](#) , y según [Zhao et al. \(2003\)](#) podemos agregar los métodos basados en inteligencia artificial y clasificadores o según [Parmar and Mehta \(2014\)](#) y [Pandya et al. \(2013\)](#) pueden agregarse los métodos híbridos, pero las dos primeras clasificaciones son una constate en la literatura.

Los métodos holísticos tratan a la imagen como un todo, donde la extracción de características depende de la variación total de la información que conforma la imagen, por esta razón no sabemos que características se extraen, ni podemos darles una jerarquía de valores, esta es la razón por la cual son sensibles a cambios de iluminación y fondo.

Este es el motivo por el cual varias de las contribuciones en estado del arte se concentran en presentar métodos holísticos robustos a esta clase de problemas como [Zhao and Yang \(1999\)](#) o nuevas forma de pre-procesamiento como [Gross and Brajovic \(2003\)](#).

Los métodos basados en modelos, también conocidos como basados en caracte-

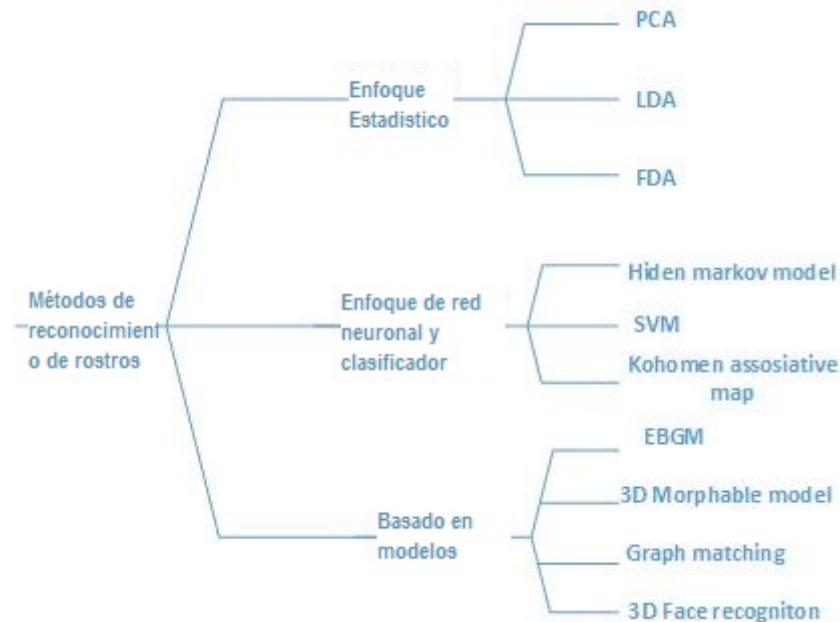


Figura 2.1: Clasificación de métodos de reconocimiento de rostro por enfoque según Zhao et al. (2003).

rísticas, como lo dice su nombre se basan en formas localizadas de extracción de características o áreas localizadas de la imagen como pueden ser los ojos. Generalmente el reconocimiento se realiza mediante una fórmula de distancia entre las características de dos diferentes imágenes, un buen ejemplo de ello es **EBGM** en Wiskott et al. (1997) y en Bolme (2003).

Además Zhao et al. (2003) considera que se puede agregar a estas dos categorías un tercer enfoque basado en redes neuronales y clasificadores donde en vez de tener una fórmula de distancia, el reconocimiento se realiza a través de un clasificador o otro método basado en inteligencia artificial. Pero la extracción de características sigue dependiendo de una forma que puede encajar como método holístico o basado en características.

2.1. Métodos holísticos

En esta sección se hace una breve introducción a los métodos holísticos que se usan como referencia para la propuesta. Se menciona su funcionamiento general y sus principales características.

2.1.1. *Principal Component Analysis (PCA)*

PCA es un método para la reducción de la dimensionalidad sin supervisión que puede ser aplicada para abordar el problema de reconocimiento de rostros. En Sirovich

and Kirby (1987) se argumenta que cualquier imagen de rostro puede ser aproximadamente reconstruida como la suma con pesos de una pequeña colección de imágenes base que llaman *eigenimages* y la media de las imágenes del rostro. Con ello Turk and Pentland (1991) presenta su conocido método de *eigenfaces*.

A continuación una breve explica como funciona este método:

Se supone que Γ es un vector $N^2 \times 1$ que corresponde a una imagen de rostro $N \times N$ llamada I donde el objetivo es representar $\Gamma(\Phi - \text{media})$ en un espacio de baja dimensión; $\Phi - \text{media} = w_1 u_1 + w_2 u_2 + \dots + w_k u_k$ donde $k \ll N^2$, w es el peso asociado al *eigenface* u .

Entonces si se tiene un conjunto de imágenes de rostro del mismo tamaño I_1, I_2, \dots, I_M donde cada I_i se puede representar como un vector Γ_i , se tiene que calcular el vector promedio de rostro Ψ :

$$\Psi = \frac{1}{M} \sum_{i=1}^M \Gamma_i$$

La nueva representación del rostro es:

$$\Phi_i = \Gamma_i - \Psi$$

Después se calcula una matriz de covarianza C :

$$C = \frac{1}{M} \sum_{n=1}^M \Phi_n \Phi_n^T = A A^T$$

Siendo una matriz $N^2 \times N^2$ donde $A = [\Phi_1 \Phi_2 \dots \Phi_M]$ es una matriz $N^2 \times M$.

Sobre la matriz $A A^T$ se debe hallar auto vectores u_i , pero resulta ser una matriz muy grande lo que resulta poco práctico, por lo que se considera la matriz $A^T A$ que es una matriz $M \times M$ donde si podemos calcular auto vectores v_i siendo:

$$A^T A v_i = \mu_i v_i$$

Dónde existe una relación de correspondencia donde los M autovalores y autovectores de $A^T A$ son los M autovalores y autovectores más grandes de $A A^T$. De esta manera mantenemos los K vectores más significativos. Ahora cada rostro está representado por:

$$\Phi_i - \text{media} = \sum_{j=1}^K w_j u_j, (W_j = u_j^T \Phi_i)$$

Dónde u_j es un *eigenface*, finalmente cada rostro puede ser representado como una colección de pesos:

$$\Omega_i = \begin{bmatrix} w_1^i \\ w_2^i \\ \dots \\ w_k^i \end{bmatrix}, i = 1, 2, \dots, M$$

De esta manera **PCA** reduce la dimensión de una imagen de rostro a un conjunto de pesos. Un problema que tiene esta relacionado al proceso de reducción en si donde no se sabe que características se están extrayendo y presenta una susceptibilidad a la variación de la data ingresada.

2.1.2. **KFA**

KFA Liu (2006) es un método que capitaliza la técnicas de incremento de la dimensionalidad, donde argumenta que métodos de reducción de dimencionalidad como **PCA** tienen problemas reconociendo patrones complejos como lo es el rostro, por lo tanto es necesario un método que pueda reconocer patrones de alta dimencionalidad.

KFA primero realiza un mapeo del espacio de la información de entrada a un espacio de características de alta dimensionalidad, y después implementa el análisis multiclase de Fisher en dicho espacio de características. **KFA** usa Gabor Wavelets (en la sección 3.3 se explica con más detalle) como método para incrementar la dimensio-nalidad.

KFA posee varias ventajas sobre **PCA** ya que su representación de la información permite una mejor distinción entre las clases que componen su espacio dimensional.

2.1.3. **Linear Discriminant Analysis (LDA)**

En aplicaciones aplicaciones típicas de reconocimiento de rostros tenemos vectores de imágenes de rostros de enormes dimensiones, y a pesar de ser proyectados en un sub-espacio, la dimensión de las características en varios casos son mayores a centenar, tampoco es conveniente que la dimensión del sub-espacio sea pequeña.

LDA usado para el reconocimiento de rostros **Zhao et al. (1999)** es un método holístico que consiste en dos pasos: primero se proyecta la imagen del rostro de su representación original como vector a un sub-espacio de rostro a través de **PCA**, luego usamos LDA para obtener un clasificador lineal en el sub-espacio de rostros creado. El criterio que se usa para determinar la dimensión del sub-espacio permite generar características que separan las clases a través de **LDA** del total de la representación del sub-espacio.

LDA presenta problemas relacionados a **PCA** como poca tolerancia al ruido y su se obtiene datos de entrada con grandes variaciones de iluminación.

En esta sección se ha descrito ejemplos de métodos holísticos de diferentes clases. Cada uno enfrenta el reconocimiento de rostros de manera diferente mediante reducción de dimencionalidad lineal, con un clasificador y con un aumento de la dimencionalidad.

En todos ellos no sabemos que características estamos midiendo, en contraste con la próxima sección donde presentamos un método basado en características.

2.2. Métodos basados en modelos y características

Un enfoque para abordar el reconocimiento de rostros es mediante el uso de modelos para poder hallar características locales las cuales son ponderadas para hallar una respuesta al reconocimiento de rostros. A continuación mencionamos algunos métodos que usan este enfoque.

2.2.1. 3-D Morphable Model

3-D *Morphable Model* se utilizan para el análisis de rostros debido a las propiedades intrínsecas de los rostros en 3D que proporcionan una representación que es inmune a las variaciones intra-personales como la pose y la iluminación.

En [Huang et al. \(2003\)](#) dada una imagen de entrada facial individual, un 3-D *Morphable Model* puede recuperar el rostro en 3D (forma y textura) y propiedades de la escena (pose y la iluminación) a través de un proceso de adaptación. El proceso de reconocimiento se realiza mediante una comparación de características con el modelo 3D puesto en la misma posición que la imagen a identificar.

2.3. Métodos basados en redes neuronales

La idea básica del uso de una red neuronal es adaptar la red con una entrada para cada pixel, pero como esto es una solución poco práctica debido a la gran cantidad de píxeles, una solución es modificar los datos de entrada de la red a través de alguna técnica de reducción de dimensionalidad o algún método de extracción de características.

Una propuesta del uso de redes neuronales es [Cottrell Cottrell and Fleming \(1990\)](#) donde se utiliza dos redes perceptrón multicapa con *back propagation* donde la primera capa trabaja en modo auto asociado extrayendo características para la segunda capa donde se realiza la clasificación. Es propuesto como un método para reconocer una gran cantidad de imágenes pero el autor menciona que inclusive en buenas condiciones no mejora el desempeño en comparación a [PCA](#).

Otro método de reconocimiento de rostros automatizado es presentado por [Lin et al. \(1997\)](#), donde utiliza una red neuronal que se basa en la toma de decisiones probabilísticas *Probabilistic Decision-Based Neural Network* ([PDBNN](#)), consta de tres módulos, detector de rostros, detector de ojos, y reconocedor de rostros. Este método se diferencia de otros porque utiliza la zona de la cara que contiene las cejas, los ojos y la nariz, pero no la boca. La boca no se considera porque presenta muchas variaciones por el cambio de expresión facial, por lo que al fin de lograr un método robusto a cambios de expresión se descarta esa zona.

Las PDBNN tienen una característica única, que es su estructura modular. Es decir, para que se reconozca cada clase, el PDBNN dedica una de sus sub-redes para la representación de esta clase. En comparación con la mayoría de los sistemas de reconocimiento multi-clase utilizando una función de discriminar entre dos clases, tiene PDBNN una menor tasa de falsas alarmas / rechazo debido a que sus funciones discriminantes obedecen a una restricción probabilística.

En Er et al. (2002) se presentó un método holístico para el reconocimiento facial, donde las características se extraen con PCA y se utilizan como entrada para una red neuronal *Radial Basis Function* (RBF). Las redes RBF tienen un buen rendimiento para los problemas de reconocimiento, tienen topología compacta y el aprendizaje es rápido. Además de las tareas de reconocimiento de rostro tradicionales (identificación y verificación de la identidad) las redes neuronales se han utilizado para diversas tareas, tales como la identificación de género y el reconocimiento de expresiones faciales.

2.4. Consideraciones finales

El reconocimiento de rostros es un área con una gran cantidad de trabajos y muy amplia en técnicas como se vio en este capítulo, cabe resaltar las varias formas en que la totalidad de técnicas presentadas en el estado del arte pueden ser agrupadas, ya que no existe un consenso unificado en su clasificación.

En el estado del arte aun son pocas las propuestas de llevar el reconocimiento de rostros a la video vigilancia como se observa en el Capítulo 1 por ello su necesidad de proponer y probar propuestas para lograr un reconocimiento de rostros en video vigilancia abordando todas las dificultades que se presenta en este tipo de escenarios donde el ambiente no es controlado.

Capítulo 3

Marco teórico de la propuesta

En este capítulo se presenta todos los conceptos y técnicas usadas para el desarrollo de la propuesta desde las técnicas de detección usadas hasta el proceso de reconocimiento.

3.1. Métodos de detección

Es necesario resaltar la importancia de una detección adecuada para el reconocimiento ya que ambos son procesos muy relacionados entre sí. En esta sección se explican tres métodos de detección uno para encontrar rostros, otro para validar los rostros detectados y finalmente uno para encontrar puntos de dentro de los rostros.

3.1.1. Detector de Viola-Jones

El detector de Viola-Jones también conocido como *Haar Detector* en implementaciones libres, se presenta en [Viola and Jones \(2001\)](#) donde se hace uso de las características Haar (Figura 3.1) en una representación conocida como Imagen Integral, también se utiliza el Algoritmo Ada Boost y un clasificador en cascada para completar el proceso de detección.

En [Viola and Jones \(2001\)](#) se explica que el trabajo de detección se basa en tres conceptos:

3.1.1.1. Imagen integral

Usa un concepto parecido a una tabla de texturas donde tenemos en cada punto el valor de la suma de los pixeles de arriba y a la derecha de dicho punto.

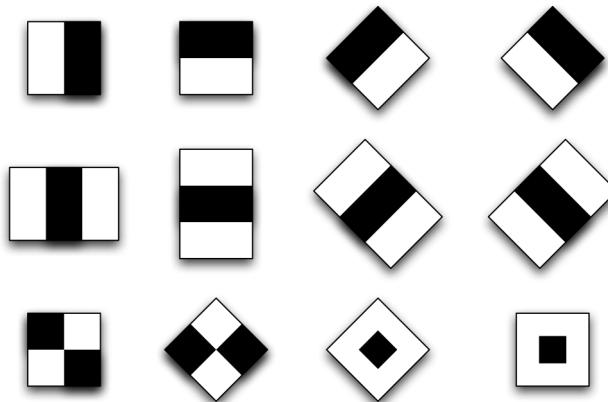


Figura 3.1: Ejemplo de característica Haar, extraído de [Viola and Jones \(2001\)](#). Estos rectángulos pueden ser entendidos como rectángulos en los cuales son determinados como áreas de baja iluminación y otros como áreas de alta iluminación de esta manera podemos representar el contraste entre dos secciones de la imagen,

Esto permite el calculo rápido de valor de los pixeles para cualquier sección de la imagen, dicho valor nos ayuda a hacer un calculo rápido del contraste en cualquier sección, lo cual permite el calculo de cualquier característica Haar en tiempo constante.

3.1.1.2. Algoritmo Ada Boost

Debido a la existencia de un gran numero de características Haar, aproximadamente más de 45,000 no es posible usar todas en un proceso de clasificación, por lo que es necesario encontrar un conjunto pequeño de características que juntas formen un clasificador efectivo.

En [Viola and Jones \(2001\)](#) se presenta una modificación al algoritmo Ada Boost para seleccionar las mejores características Haar y con ellas entrenar varios clasificadores débiles en términos de su tasa de aciertos, que en combinación permiten obtener un clasificador fuerte. De esta manera tener un clasificado robusto para el proceso de detección.

3.1.1.3. Clasificador en cascada

La idea tras un clasificador en cascada es organizar todos los clasificadores débiles en un estilo en el cual estén uno detrás de otro como una cascada. Cuando se analiza una región esta pasa por el primer clasificador y si el resultado es negativo se descarta, pero si resulta positivo pasa al siguiente clasificador hasta el final de la cascada y así sucesivamente de esta manera obtiene un solo clasificador, más robusto que los clasificadores que lo componen. Este proceso se puede apreciar en la Figura 3.2.

El detector de Viola-Jones es ampliamente usado y reconocido como un detector

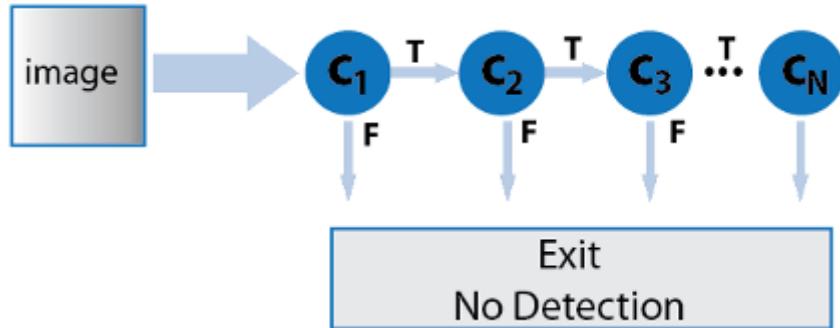


Figura 3.2: Muestra del clasificador en cascada, donde una imagen es analizada por un clasificador tras otro desde el primero hasta el n-esimo si resulta la imagen positiva, extraído de [Viola and Jones \(2001\)](#).

que entrega una gran cantidad de verdaderos positivos (detecciones acertadas). Pero es necesario mencionar sus limitaciones y falencias, de la misma manera que entrega un gran porcentaje de verdaderos positivos también entrega falso positivos (detecciones presentadas como verdaderas pero que no lo son) esto es debido a que el cálculo de la imagen integral se hace en la escala de grises a veces alguna configuración de sombras producidas en la imagen genera la misma distribución de contrastes que un rostro humano.

Otro defecto es que solo detecta rostro de frente. No puede detectar otras posiciones del rostro humano simultáneamente. A pesar de estas deficiencias sigue siendo usado no solo para rostros sino para otro tipo de objetos ya que su mayor ventaja es que es entrenarle.

3.1.2. Detector de *Histogram of Oriented Gradients* (**HOG**)

Un detector de *Histogram of Oriented Gradients* (**HOG**) usa una combinación del descriptor de **HOG** y un clasificador basado en *Support Vector Machine* (**SVM**), originalmente usado para detectar personas [Dalal and Triggs \(2005\)](#), su uso se ha incrementado a varias situaciones, incluyendo la detección de rostros, a continuación exponemos las ideas detrás de este detector.

3.1.2.1. Descriptor de **HOG**

Un descriptor de **HOG** muestra el cálculo de la gradiente de una imagen, la gradiente es cambio de la intensidad de la imagen en una cierta dirección, pudiendo ser representado como un vector que posee magnitud g y dirección θ , el cálculo del gradiente en una imagen se hace por pixel y retorna la dirección del cambio y su magnitud, este proceso se observa en las siguientes ecuaciones:

$$dx = I(x+1, y) - I(x-1, y) \quad (3.1)$$

$$dy = I(x, y + 1) - I(x, y - 1) \quad (3.2)$$

$$\theta(x, y) = \arctan\left(\frac{dy}{dx}\right) \quad (3.3)$$

$$g(x, y) = \sqrt{dx^2 + dy^2} \quad (3.4)$$

Donde x y y son las coordenadas de la imagen y la función $I(x, y)$ es el valor del pixel o intensidad en dichas coordenadas. Luego se divide la imagen en secciones iguales y en cada una de estas secciones se calcula un histograma de las orientaciones para conocer la orientación dominante de cada sección, de esta manera creamos un vector de características.

Con esta información se puede describir el contorno y forma de una figura, lo que es útil para realizar una clasificación.

3.1.3. *Support Vector Machine (SVM)*

Un [SVM](#) [Cortes and Vapnik \(1995\)](#) es un clasificador lineal en el cual se intenta crear una linea divisoria entre las representación (vectores de características) de ejemplos positivos y negativos de un objeto determinado.

Esta basado en el aprendizaje estadístico para resolver problemas de clasificación de patrones. Los clasificadores lineales se caracterizan porque aprenden una función lineal para separar las clases. No se trata de una agrupación por similitudes, sin que existe una separación definida entre clases.

Dado un conjunto de ejemplos de entrenamiento (muestras) podemos etiquetar las clases y entrenar una SVM para construir un modelo que prediga la clase de una nueva muestra. Intuitivamente, una SVM es un modelo que representa a los puntos de muestra en el espacio, separando las clases por una linea representada por la función del clasificador. Cuando las nuevas muestras se ponen en correspondencia en función de su proximidad, pueden ser clasificadas a una u otra clase, dependiendo de la proximidad a cada una. Mas formalmente, la idea principal de SVM es construir un hiperplano o conjuntos de hiperplanos en un espacio de dimensionalidad muy alta como superficie de decisión, de tal forma que, el margen de separación entre ejemplos positivos y negativos sea el máximo. Una buena separación entre las clases permitirá una clasificación correcta.

[SVM](#) resulta ser un clasificador muy robusto y usado, el cual presenta varias modificaciones alrededor del estado del arte.

3.1.4. *Constrained Local Neural Field (CLNF)*

[CLNF](#) es un método para encontrar puntos de interés en imágenes de rostros presentado en [Baltrusaitis et al. \(2013\)](#) e implementado para aplicaciones de video

en [Baltrušaitis et al. \(2016\)](#) en la presentación de su framework “Open Face” para el tracking y detección de rostros, donde publica el código de su propuesta, [CLNF](#) se basa en *Constrained Local Model* ([CLM](#)), por lo tanto empezamos con una explicación de [CLM](#).

[CLM](#) es una forma para localizar un conjunto de puntos restringido por un modelo en una imagen, el proceso consiste en tomar una muestra de una área de la imagen donde se realizó una estimación inicial proyectarla a un marco de referencia. Por cada punto se genera una imagen de respuesta donde se le da una puntuación por tener el punto a localizar en cada pixel. Este puntaje es determinado a través de un detector local que evalúa el punto según un modelo de distribución de puntos en donde penaliza distribuciones complicadas y premia distribuciones simples y mas probables, de esta manera se evalúan todos los puntos en conjunto.

En [Baltrusaitis et al. \(2013\)](#) se propone el uso de redes neuronales para los detectores locales de los puntos, con lo cual logra una mejor determinación de puntos en relación con el resto de puntos detectados. Compara sus resultados con otras técnicas del estado del arte así mismo realiza pruebas con imágenes obtenidas de Internet que presentan condiciones no controladas.

3.2. Técnicas de pre-procesamiento

Como se puede observar en las secciones anteriores un problema en especial difícil es la iluminación, por lo que es necesario hablar sobre las técnicas de pre-procesamiento, se entiende como pre-procesamiento a las técnicas que modifican una imagen para acentuar alguna característica, estas serán usadas para afrontar dicho problema.

3.2.1. Ecualización de histograma

Las ecualización del histograma de una imagen es una transformación que pretende obtener para una imagen un histograma con una distribución uniforme. Es decir, que exista el mismo número de pixeles para cada nivel de gris del histograma de una imagen monocroma [Orlova \(2002\)](#). La función de la ecualización es:

$$h(v) = Redondeo \left(\frac{cdf(v) - cdf_{min}}{(M \times N) - cdf_{min}} \times (L - 1) \right) \quad (3.5)$$

Donde cdf_{min} es el mínimo valor no nulo de la función de distribución de acumulación. $M \times N$ es el numero de pixeles en la imagen y L es el numero de niveles de la escala de gris.



Figura 3.3: Ecualización de histograma - Izquierda: imagen en escala de Grises. Derecha: imagen ecualizada.

3.2.2. Transformada de logaritmo

La Transformada de logaritmo asigna un rango estrecho de valores (píxeles en escala de grises o un canal de un espacio de color) de un rango amplio de valores de entrada. La transformada de logaritmos es útil si se necesita expandir los valores de píxeles oscuros de una imagen mientras se comprime los valores altos [Thamizharasi and Jayasudha \(2015\)](#)

$$s = c * \log(1 + 256 * r) \quad (3.6)$$

Donde r es la imagen de entrada, c es una constante, s es la imagen mejorada. Para la transformada de logaritmo se utilizo el valor de 20 para la constante c .



Figura 3.4: Transformada de logaritmo - Izquierda: imagen en escala de Grises. Derecha: imagen con Transformada de logaritmo.

3.2.3. Transformada discreta de coseno

La transformada Discreta de coseno (DCT por sus siglas en inglés) es utilizada especialmente en el procesamiento de señales e imágenes. DCT expresa en una secuencia finita de puntos, datos en términos de sumas de funciones de coseno en diferentes frecuencias. En el procesamiento de imágenes, la utilización de DCT ayuda a descomponer una imagen en frecuencias, donde usualmente los pequeños componentes de frecuencia altas pueden ser descartados. La ecuación en 2D está dada por la Ecuación. [3.7](#) [Thamizharasi and Jayasudha \(2015\)](#)[Vishwakarma \(2015\)](#)

$$C(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} f(x, y) \cos\left[\frac{\pi(2x+1)u}{2N}\right] \cos\left[\frac{\pi(2y+1)v}{2N}\right] \quad (3.7)$$

Donde: u , v , N , $\alpha(u)$ y $\alpha(v)$ se definen de igual forma que la ecuación 1D. La ecuación inversa está definida por la Ecuación. 3.8.

$$f(x, y) = \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \alpha(u)\alpha(v)C(u, v) \cos\left[\frac{\pi(2x+1)u}{2N}\right] \cos\left[\frac{\pi(2y+1)v}{2N}\right] \quad (3.8)$$

El resultado de la transformada de coseno, es una matriz de coeficientes positivos y negativos, los cuales representan la adición o resta de una determinada frecuencia para generar la imagen procesada por la transformada discreta de Coseno.

Se ha expuesto tres técnicas de pre-procesamiento que serán usadas en usadas en la propuesta del Capítulo 4, todas abordan el problema de la iluminación de manera diferente, es necesario tener en cuenta que este problema ha sido tratado mucho en el estado del arte pero aun no se encuentra una técnica de mejora de iluminación sea robusta en cualquier escenario.

3.3. *Elastic Bunch Graph Matching (EBGM)*

EBGM presentado en [Wiskott et al. \(1997\)](#) se basa en el concepto que las imágenes de los rostros reales tienen muchas características no lineales que no son abordadas por los métodos de reconocimiento holísticos, tales como variaciones en la iluminación, pose y expresión. También se apoya en el argumento que las características visuales que se basan en Gabor Wavelet, han probado ser un buen modelo del procesamiento visual temprano en el cerebro, más precisamente células simples en la corteza visual primaria, por ello también se considera un algoritmo inspiración biológica.

Para ello, una transformación de Gabor wavelet crea una arquitectura de enlaces dinámicos que proyecta el rostro en una malla elástica que es conocida como Face Graph. El Gabor Jet es un nodo en la malla elástica, el cual describe el comportamiento alrededor de un pixel. Esto es el resultado de la convolución de una imagen con varias máscaras de Gabor, el cual es usado para detectar formas y extraer características.

El reconocimiento esta basado en la similaridad de la respuesta mascara de Gabor a cada nodo. La dificultad con este método es el requerimiento de marcar puntos precisos en los rostros.

A continuación se explica en detalle su funcionamiento

3.3.1. Gabor Wavelet

Las Gabor Wavelet son funciones que modifican las imágenes en el espacio de las frecuencias. El espacio de las frecuencia esta estrechamente relacionado al análisis de Fourier, por lo que es necesario hacer una breve descripción.

Una transformada de Fourier descompone una señal de tal manera que pueda ser representada como una combinación de sinusoidales. Cuando se procesa señales como lo pueden ser las imágenes, son representadas como una longitud de onda, ello es muy útil ya que revela información que no puede ser vista cuando observamos la imagen original.

Mientras una señal es un función en base al tiempo, la transformada de Fourier es una función en base a la frecuencia, se puede observar la transformada de Fourier en una dimensión en la siguiente ecuación:

$$F(x(t))(\omega) = \int_{-\infty}^{\infty} x(t)e^{-i\omega t} dt \quad (3.9)$$

En la transformada de Fourier y en los Gabor Wavelet la función comprende una parte imaginaria y una real:

$$F(x(t))(\omega) = \int_{-\infty}^{\infty} x(t)\cos(\omega t)dt - i \int_{-\infty}^{\infty} x(t)\sin(\omega t)dt \quad (3.10)$$

Cuando una transformada de Fourier se aplica una frecuencia en particular el resultado es un numero complejo que corresponde a la amplitud del coseno y del seno de la función original, en cada frecuencia hay un componte real y otro imaginario.

Las Gabor Wavelet son como la transformada de Fourier solo que tiene un alcance limitado, básicamente son una sinusoidal multiplicada por una Gaussiana.

Cuando una función es convolucionada con un Gabor Wavelet la información más cerca al centro de la campana de la Gaussiana es la que es tomada en cuenta mientras la más lejana es ignorada.

La ecuación de una de una Gabor Wavelet es la siguiente:

$$W(t, t_0, \omega) = e^{-\sigma(t-t_0)^2} e^{-i\omega(t-t_0)} \quad (3.11)$$

y su convolucion es:

$$C(x(t))(t_0, \omega) = \int_{-\infty}^{\infty} x(t)W(t, t_0, \omega)dt \quad (3.12)$$

De la misma manera que Fourier produce una resultado complejo un Gabor Wavelet produce:

$$C(x(t))(t_0, \omega) = a_{real} + ia_{imag} \quad (3.13)$$

Mientras las ecuaciones presentadas hasta ahora trabajan en una dimensión es necesario presentar una forma en la que se pueda trabajar en dos dimensiones, esta es la representación del Gabor Wavelet como mascara para una convoluciónn en imágenes.

La Ecuación 3.14 define como crear una máscara de Gabor donde x, y son las posiciones en la mascara, para cualquier tamaño de mascara.

$$W(x, y, \theta, \lambda, \phi, \sigma, \gamma) = e^{-\frac{x'^2 + \gamma y'^2}{2\sigma^2}} \cos(2\pi \frac{x'}{\lambda} + \phi) \quad (3.14)$$

$$x' = \cos\theta + y\sin\theta$$

$$y' = -x\sin\theta + y\cos\theta$$

Los parámetros usados para la construcción de los Gabor wavelet son los mismo que se utilizan en la implementación de [Bolme \(2003\)](#), a continuación los explicamos brevemente:

- θ especifica la orientación del Gabor Wavelet.
Siendo $\theta \in \{0, \pi/8, 2\pi/8, 3\pi/8, 4\pi/8, 5\pi/8, 6\pi/8, 7\pi/8\}$
- λ especifica el ancho de onda de la función seno, empieza con 4 pixeles y aumenta en medias octavas siendo $\lambda \in \{4, 4\sqrt{2}, 8, 8\sqrt{2}, 16\}$
- ϕ especifica la fase de la función seno, pudiendo ser par e impar que representa la parte imaginaria y la parte real del wavelet respectivamente. Siendo $\{0, \pi/2\}$
- δ especifica el radio de la Gaussiana. En este caso $\delta = \lambda$.
- γ especifica el ratio de aspecto de la Gaussiana. Este parámetro es incluido para que el Wavelet se aproxime a ciertos modelos biológicos. Siendo $\gamma = 1$.

De esta manera podemos crear varios tamaños de máscaras que en la configuración original son $N \in \{25, 37, 51, 71, 101\}$. Dando 80 configuraciones de Gabor wavelet y siendo efectivas 40 máscaras por punto, debido a que existe una mascara que extraer la parte imaginaria y otra la parte real del Wavelet.

Al conjunto coeficientes (reales e imaginarios) producidos por estas máscaras de Gabor son llamados Gabor Jet que son el corazón de todo el proceso de reconocimiento de [EBGM](#).

A continuación se explica como es el proceso por el cual [EBGM](#) determina en que puntos de una imagen va extraer características. A estos punto se les conocen como puntos fiduciales.

3.3.2. Localización de puntos fiduciales

Como se menciona en la sección anterior la convolución del conjunto de máscaras de Gabor sobre un punto produce un Gabor Jet que es un conjunto de coeficientes.

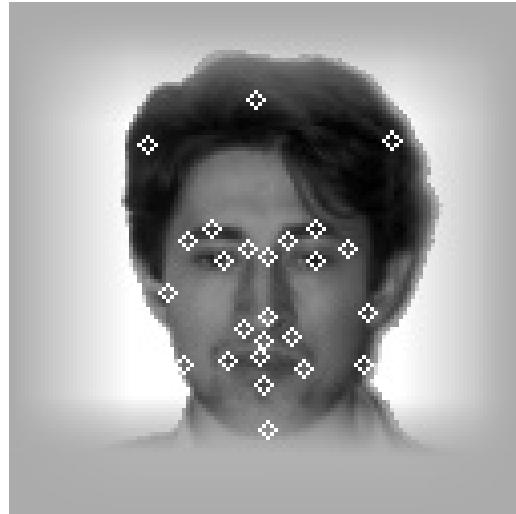


Figura 3.5: Muestra de los puntos fiduciales elegidos en [EBGM](#)[Bolme \(2003\)](#), en cada punto se realiza 40 convoluciones con máscaras de Gabor para crear un Gabor Jet por punto

Siguiendo el trabajo de [Wiskott et al. \(1997\)](#), Bolme realiza una implementación de código libre de todo el algoritmo y da un mayor detalle de su funcionamiento, listando que puntos de interés debemos encontrar en una imagen. Estos puntos son conocidos como puntos fiduciales, en el Cuadro 3.1 se puede observar una lista de ellos, y en la Figura 3.5 se puede ver su localización.

Cuadro 3.1: Lista de puntos fiduciales presentada en [Bolme \(2003\)](#)

1. Ojo izquierdo	2. Ojo derecho
3. Centro del puente de la nariz	4. Pico de la ceja izquierda
5. Pico de la ceja derecha	6. Interior de la ceja izquierda
7. Interior de la ceja derecha	8. Exterior de la ceja izquierda
9. Exterior de la ceja derecha	10. Centro de la punta de la nariz
11. Centro de la base de la nariz	12. Izquierda de la base de la nariz
13. Derecha de la base de la nariz	14. Centro parte superior boca
15. Centro parte inferior boca	16. Esquina izquierda de la boca
17. Esquina derecha de la boca	18. Parte central del tope de la cabeza
19. Parte izquierda del tope de la cabeza	20. Parte derecha del tope de la cabeza
21. Costado izquierdo del rostro	22. Costado derecho del rostro
23. Centro de la barbilla	24. Quijada izquierda
25. Quijada derecha	

El método por el cual [EBGM](#) encuentra los puntos fiduciales es a través del uso de moldes, imágenes en las cuales los puntos han sido encontrados manualmente. En los trabajos de [Wiskott et al. \(1997\)](#) y [Bolme \(2003\)](#) utilizan 70 imágenes normalizada a un mismo tamaño y donde las coordenadas de los ojos son las mismas.

Sobre cada imagen de molde y en cada punto marcado se realiza las convoluciones de las máscaras de Gabor para producir un Gabor Jet, de esta manera todos los puntos

en todas las imágenes tienen un Gabor Jet. Finalmente todos los Gabor de un punto, por ejemplo el ojo izquierdo, son reunidos en un solo grafo conocido como Bunch Graph que contiene todos los Gabor Jet de los modelos.

De esta manera se obtiene un modelo general con información de todos los modelos, también contiene la información de coordenada de cada punto en cada modelo y un punto de coordenada promedio.

Para reconocer una nueva imagen se realiza una normalización de tamaño y una traslación coordenadas de los ojos a coordenadas preestablecidas a través de una transformación matricial. Mediante un alineamiento entre las coordenadas normalizadas de los ojos y las coordenadas de los ojos del Bunch Graph se encuentran el resto de los puntos fiduciales en la imagen usando la información de coordenada promedio almacenada en el Bunch Graph. Finalmente se encuentra la posición final del punto usando como referencia el Gabor Jet con mayor similaridad en el Bunch Graph

Según [Bolme \(2003\)](#) y [Wiskott et al. \(1997\)](#), y como se menciona en la sección anterior el resultado de una transformada de Fourier en una función y en una frecuencia determinada es un numero complejo, que tiene una correspondencia en amplitud en términos de senos y cosenos que pueden ser representada como una coordenada polar.

$$x_\omega(t) = a_{\omega,r} \cos(t) + a_{\omega,i} \sin(t) = a \cos(t + \phi). \quad (3.15)$$

Es posible representar las suma de las sinusoidales como una función coseno dándole una amplitud y fase específica. Esto sucede cuando las sinusoidales son multiplicadas por una Gaussiana como se observa en la Ecuación 3.13 y según lo visto en la Ecuación 3.15 se puede representar los coeficientes complejos del resultado de las convoluciones como coordenadas polares de magnitud a y un angulo ϕ

$$a_{real} = a \cos \phi \quad (3.16)$$

$$a_{imag} = a \sin \phi \quad (3.17)$$

y podemos halla a y ϕ mediante:

$$a = \sqrt{a_{real}^2 + a_{imag}^2} \quad (3.18)$$

$$\phi = \begin{cases} \arctan(a_{imag}/a_{real}) & \text{si } a_{real} > 0 \\ \pi + \arctan(a_{imag}/a_{real}) & \text{si } a_{real} < 0 \\ \pi/2 & \text{si } a_{real} = 0 \text{ y } a_{imag} >= 0 \\ -\pi/2 & \text{si } a_{real} = 0 \text{ y } a_{imag} < 0 \end{cases} \quad (3.19)$$

Gracias a esta transformación a coordenadas polares podemos representar la información compleja del resultado de la convolucion de una mascara de Gabor.

Cabe mencionar que la mayor debilidad del algoritmo yace en este proceso ya que si el alineamiento del grafo falla por culpa de un error en las coordenadas de los ojos el resto de las estimaciones iniciales falla. Otra dificultad que presenta es la dependencia a sus modelos, esto quiere decir que por ejemplo si entre sus modelos no

se encuentra alguien con barba cuando se intente encontrar puntos en nueva imagen de una persona con barba no va existir ningún Gabor Jet en el Bunch Graph que sea una buena referencia para el ajuste de la posición final del punto.

Finalmente es necesario contar con las coordenadas de ojos de antemano, este hecho por simple que parezca es una de las limitaciones para que **EBGM** se pueda usar en un sistema de reconocimiento de rostro automatizado sobre todo en vídeo vigilancia donde encontrar ojos es difícil.

3.3.3. Face Graph

Una vez hallada la posición final de los puntos se extrae los coeficientes finales (la representación en coordenadas polares de la convolución) para los Gabor Jet donde se almacenan una estructura que se conoce como Face Graph. A pesar que en la literatura se representa como un grafo , Figura 3.6 , Un Face Graph en realidad es un vector de características.

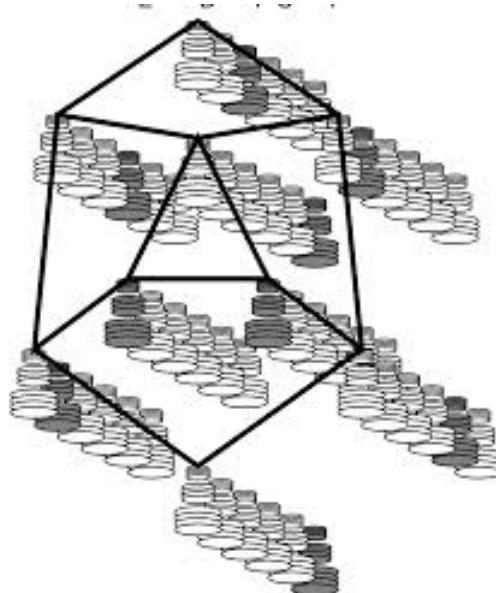


Figura 3.6: Representación de Face Graph presentada en [Wiskott et al. \(1997\)](#)

Al final solo se hace una comparación punto a punto entre dos Face Graph para el proceso de reconocimiento.

3.3.4. Función de similitud

El proceso de reconocimiento se lleva a cabo mediante una función de similitud que compara cuan iguales son dos Face Graph, dicha función se expresa de la siguiente

forma:

$$L_{jet}(G, G') = \frac{1}{N} \sum_{i=0}^N S(J_i, J'_i) \quad (3.20)$$

Donde N es el numero de puntos fiduciales y J_i, J'_i son los i-ésimos Gabor Jet que pertenecen respectivamente a los Face Graph G, G' que van a ser comparados.

La función de similitud entre dos Gabor Jet es la siguiente:

$$S(J, J') = \frac{\sum_{j=1}^N a_j a'_j \cos(\phi_j - \phi'_j)}{\sqrt{\sum_{j=1}^N a_j^2 \sum_{j=1}^N a'^2_j}} \quad (3.21)$$

Después de explicar el funcionamiento de **EBGM** es necesario mencionar las situaciones donde tiene problemas para funcionar, según [Givens et al. \(2004\)](#) en su estudio comparativo con **PCA** donde explica varias pruebas. Cuando hay cambios bruscos de expresión, cambio de los ojos y boca, y ojos cerrados son las situaciones en las cuales **EBGM** tiene problemas para realizar el reconocimiento, finalmente concluye que dichos problemas tienen en parte relación con el hecho que se necesita las coordenadas de los ojos para empezar el reconocimiento y las dificultades relacionadas para encontrar dichas posiciones afectan el desempeño del algoritmo.

En esta sección se explicó el funcionamiento de **EBGM** que extrae característica de partes bien definidas del rostros, que también son conocidas como características biométricas.

3.4. Consideraciones finales

En este capítulo se ha expuesto todos los métodos que interviene en alguna u otra forma en la propuesta, se ha presentado una introducción a cada una de ellos.

Es necesario resaltar que **EBGM** es un método de reconocimiento biométrico, que se refiere a una forma de reconocimiento basado en un vector de características que derivan mayormente de características biológicas, se considera mucho mas confiable que otros métodos de reconocimiento que usan claves, contraseñas, tarjetas, etc [Alice \(2003\)](#). Esto se debe a cumple con los requerimientos necesarios para ser considerado como tal siendo estos los siguientes:

- Universalidad.- Toda persona debe tener dicha característica.
- Distintividad.- Cualquier par de personas debe ser diferente en términos de dicha característica
- Permanencia.- La característica debe ser lo suficientemente invariable a lo largo del tiempo para que pueda ser usada como criterio de comparación.

- Medible.-La característica debe ser cuantitativamente mensurable.

Por ello es adecuado su uso para un escenario de video vigilancia en el cual se aplica la propuesta presentada en el siguiente capítulo.

Capítulo 4

Propuesta

La propuesta de este trabajo presenta un *pipeline* de reconocimiento de rostros en vídeo donde se usa **EBGM** que ha sido elegido por sus resultado en comparativas realizadas con otros métodos holísticos, y por ser un método que es considerado biométrico por lo expuesto en los capítulos anteriores.

Cabe resaltar que la principal dificultad para su uso en vídeo vigilancia es su necesidad de tener las coordenadas de los ojos ya encontradas para poder determinar el resto de puntos fiduciales. Este hecho es especialmente importante ya que el proceso de estimación de puntos esta ligado a que tipos de modelos se usa y puede ser influenciado por algún patrón dentro de los modelos.

A pesar de la dificultad mencionada, **EBGM** sigue siendo un algoritmo adecuado para el reconocimiento de rostros en vídeo vigilancia. Por ello se propone adoptar el uso de **CLNF** para reemplazar el proceso de estimación de puntos fiduciales que **EBGM** usa.

A continuación se presenta el esquema general de la propuesta de este trabajo de tesis.

4.1. Esquema general de la propuesta

Para poder realizar el reconocimiento de rostros en vídeo vigilancia se debe establecer un *pipeline* de procesos para alcanzar dicha meta, a continuación se describe los pasos necesarios para el reconocimiento de rostros:

- [1] Realizar la detección de rostros en la escena de vídeo vigilancia.
- [2] Validar dichas detecciones para descarta falsos positivos.
- [3] Encontrar puntos fiduciales en el rostro detectado.

- [4] Aplicar pre-procesamiento a la imagen de rostros detectada.
- [5] Normalizar imágenes de rostros a un tamaño y posición predefinida.
- [6] Realizar el proceso de reconocimiento usando **EBGM**.
- [7] Mostrar el resultado del reconocimiento

En la Figura 4.1 se puede observar la lista de procesos o *pipeline* del procedimiento general de la propuesta desde adquisición de la imagen hasta el resultado del reconocimiento.

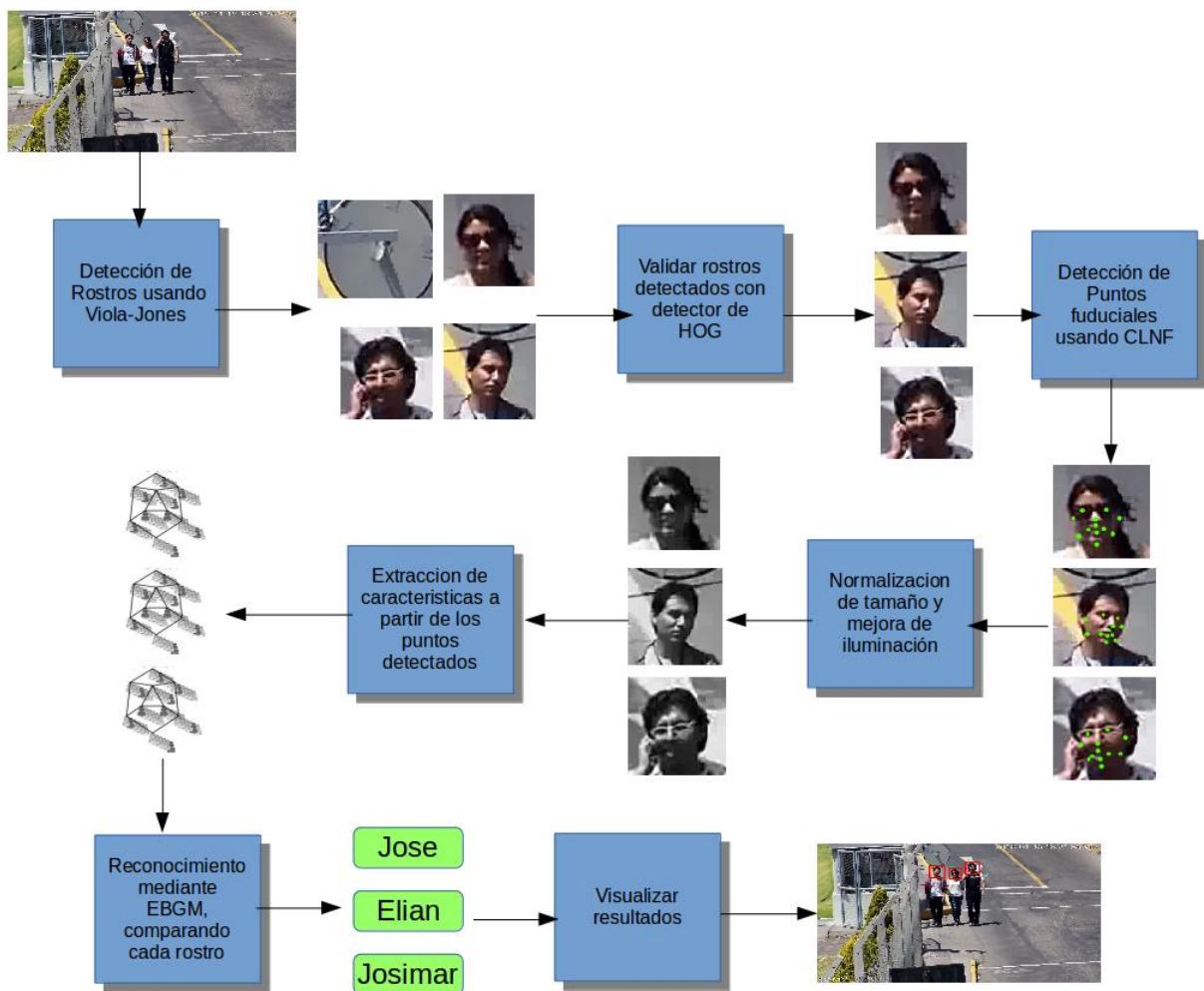


Figura 4.1: *Pipeline* de propuesta, muestra el proceso de reconocimiento de rostros desde la detección en una escena hasta la muestra del resultado.

En el Algoritmo 1 se describe el funcionamiento de nuestra propuesta donde se trata cada punto relevante:

Algoritmo 1: Pipeline de propuesta de reconocimiento de rostros

```

1 names ← crear lista para almacenar nombres de imágenes de entrenamiento;
2 graphs ← crear una lista de FaceGraph;
3 filter ← crear un filtro de puntos fiduciales;
4 clnf_model ← leer y cargar modelo de entrenamiento para CLNF;
5 mask ← máscaras de Gabor a partir de un conjunto con parámetros;
6 trainSet ← leer lista de direcciones y nombres de imágenes de
    entrenamiento;
7 para cada element en trainSet hacer
8     trainIm ← leer imagen a partir de element;
9     success ← buscar puntos fiduciales en trainIm con clnf_model;
10    si success es cierto entonces
11        faceLandmarkTrain ← asignar lista de puntos fiduciales encontrados
            con clnf_model;
12        añadir nombre de la imagen a la lista names;
13        graph←ConvertirImagenGrafo(trainIm,faceLandmarkTrain,mask,filter);
14    en otro caso
15        escribir mensaje de error en entrenamiento;
16 cascade ← entrenar un clasificador basado en el algoritmo de Viola-Jones;
17 frame ← leer frame de vídeo de fuente;
18 si fuente de vídeo no existe entonces
19     escribir mensaje de error en fuente de vídeo y salir de la aplicación;
20 mientras siempre cierto hacer
21     frame ← leer desde fuente de vídeo;
22     si frame esta vacío entonces
23         escribir mensaje de error;
24         salir de la aplicación;
25     frameGray ← convertir frame a escala de grises;
26     detectar rostros en frameGray usando cascade;
27     faces ← guardar coordenadas de rostros detectados por cascade;
28 para cada element en faces hacer
29     roi ← recortar región de rostro en detectada en frameGray y validar
        detección con un detector de HOG;
30     success ← buscar puntos fiduciales en roi con clnf_model;
31     si var es cierto entonces
32         var ← asignar lista de puntos fiduciales encontrados con clnf_model;
33         detectedFace ←
            ConvertirImagenGrafo(roi,faceLandmark,mask,filter);
34     ajustar faceLandmark a posición real en frame;
35     dibujar faceLandmark en frame;
36     dibujar recuadro de roi en frame;
37     name ← ReconocerFaceGraph(detectedFace,graphs,names);
38     dibujar name sobre recuadro de roi;
39     visualizar frame;

```

4.2. Detección y validación de rostros

Para el proceso de detección de rostros se utilizó el algoritmo de Viola-Jones, el cual devuelve regiones donde se detectaron rostros, es conocido que dicho algoritmo funciona bien en imágenes de alta resolución y condiciones de iluminación controladas. Como se observa en la Figura 4.2, pero en el contexto de imágenes de vídeo vigilancia las cámara tiene menor resolución y no controlan la condiciones de iluminación, motivo por el cual nos da como resultados regiones que no necesariamente corresponden a un rostro (falsos positivos).

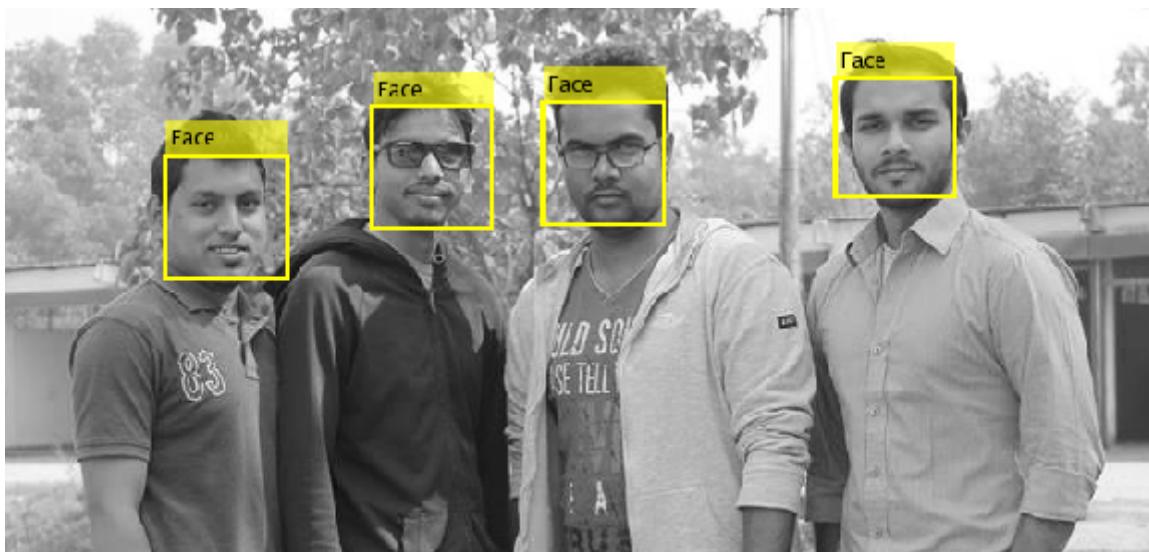


Figura 4.2: Muestra de una imagen optima detector de Viola-Jones, todos los sujetos observan a la cámara y es una imagen en buena calidad, extraída de Internet

Para que la propuesta sea robusta este trabajo presenta un validación mediante el descriptor de **HOG**, de esta manera se mejora el proceso de detección a la vez de validar los resultados del detector de Viola-Jones. En general mejora la detección de rostros en sistemas de vídeo vigilancia.

4.2.1. Detector de Viola-Jones

El proceso de detección comienza con la transformación de la escena al espacio de imagen integral, como es explicado en la Sección 3.1.1. Luego se analiza todas las regiones de la escena a través de una ventana para encontrar patrones de características Haar que cumplan con el entrenamiento del clasificador en cascada, dando como resultado las coordenadas de las regiones con rostros detectados. Este proceso se repite por varios tamaños de ventana diferente para detectar rostros en varios tamaños.

Como se observa en la Figura 4.3 y 4.4, Viola-Jones tiene problemas en condiciones de iluminación no controladas, y se puede apreciar claramente que hay varios rostros

detectados y un falso positivo. Motivo por el cual es necesario validar con el proceso siguiente.



Figura 4.3: Muestra de una escena de vídeo vigilancia usando el detector de Viola-Jones



Figura 4.4: Ejemplo de detecciones usando el detector de Viola-Jones, donde la ultima imagen es un falso positivo

4.2.2. Validación con detector de HOG

Para filtrar los falsos positivos que puede entregar el algoritmo de Viola-Jones se propone usar el detector de **HOG** para rostros.

El proceso de detección de rostros usando **HOG** toma una imagen y la transforma con el descriptor de **HOG** donde se resalta las líneas de contraste y entrega información sobre la gradiente de dicho contraste como se observa en la Figura 4.5

Para analizar las imágenes representadas por el descriptor de **HOG** se utiliza un **SVM** que se ha sido entrenado con ejemplos positivos y negativos de rostros, con ello se valida los resultados entregados por Viola-Jones vistos en la Figura 4.4.

Las información proporcionada por el descriptor de **HOG** es entregada a la siguiente parte del proceso como la primera estimación de puntos fiduciales.



Figura 4.5: Muestra de representación del descriptor de **HOG**(derecha) en una imagen de rostro(izquierda), donde el total de la imagen es transformada a un vector de características.

4.3. Detector de puntos fiduciales

A partir de los resultados obtenidos por el proceso de detección y validación se tiene una imagen la cual es utilizado en el calculo de los puntos fiduciales mediante *Constrained Local Neural Field* (**CLNF**).

El método de **EBGM** se basa en modelos de rostros, esto modelos son los puntos encontrados manualmente (ojos, nariz, boca, etc) como se observa en la Figura 4.8, dichos modelos se obtienen de un conjunto variado de rostros, a partir de los cuales se calcula un modelo promedio, que se utiliza como estimación inicial para un nuevo rostro mediante el alineamiento con el conjunto de modelos en base a las coordenadas de los ojos

Se extraen características de la nueva imagen en base a las coordenadas de puntos del modelo promedio, y por cada punto se realiza una comparación del Gabor Wavelet en la nueva imagen con su equivalente en todos los modelos, hasta encontrar el punto del modelo con mayor similitud con ello se obtiene la estimación final de los puntos. Este es un proceso de fundamental ya que una mala estimación de los puntos repercute en el proceso de reconocimiento.

Se propone el uso de *Constrained Local Neural Field* (**CLNF**) en reemplazo de todo el proceso detección de puntos usado en **EBGM**, debido a que **CLNF** es un detector robusto y probado en ambiente no controlados, con una buena tolerancia a variaciones de pose.

CLNF recibe el resultado del detector de **HOG** y realiza una estimación de puntos, a partir de ahí se restringe cada punto a un área de vecindad donde se aplica una red neuronal que entrega información espacial sobre cuales son las posiciones que tiene mayor probabilidad de ser el verdadero punto a detectar, el resultado de los puntos detectados es evaluado en conjunto para encontrar la configuración de puntos que resulte mas probable en conjunto, de esta manera se eliminan expresiones y formas de

rostro poco probables. Finalmente se entrega las posiciones de 68 puntos en el rostro.

Todo ello permite llevar a **EBGM** a un entorno de vídeo vigilancia sin ayuda de un factor humano, lo que antes no era posible. La totalidad de este proceso se puede apreciar en los algoritmos 1 y 2.

4.4. Pre-procesamiento y normalización de imágenes

Una vez el rostro ha sido detectado, validado y con los puntos fiduciales encontrados, a la imagen del rostro se le aplica una mejora en iluminación a través de ecualización de histograma, transformada de logaritmo y transformada discreta de coseno, presentada en el trabajo de Manjula ([Manjula \(2014\)](#)), siendo la siguiente Ecuación 4.1:

$$F(x, y) = c_1 * DCT + c_2 * Lg + c_3 * HE \quad (4.1)$$

Donde c_1, c_2, c_3 son valores tipo peso para equilibrar el efecto de las técnicas usadas, sus valores son de 0.3 , 0.2 y 0.5 respectivamente, donde estos valores fueron calculados después de permutar todos las combinaciones posibles. El resultado de este proceso puede verse en la Figura 4.6.



Figura 4.6: Muestra del pre-procesamiento aplicado a imágenes, comparando antes y después

Junto con ello se aplica el proceso normalización en tamaño que usa **EBGM** que mueve las coordenadas de los ojos a coordenadas pre establecidas (52, 64) y (76, 64) y cambia su resolución a un tamaño de 128×128 a través de una matriz de transformación de perspectiva, donde se incluye un borde de 30 píxeles, por lo que el tamaño efectivo de rostros en la imagen final es aproximadamente 90 píxeles de ancho.

Se mantiene la resolución del algoritmo original de [Bolme \(2003\)](#) debido a que es necesario逼近arse a resoluciones de rostros que se pueden encontrar en vídeos de

vigilancia. Este proceso y el descrito en la siguiente sección puede ser observado en el algoritmo 2

Algoritmo 2: Función para convertir una imagen a Face Graph

```
1 ConvertirImagenGrafo (image,faceLandmark,mask,filter);
2 source  $\leftarrow$  extraer coordenadas de ojos de faceLandmark;
3 M  $\leftarrow$  Calcular una matriz de transformación de perspectiva para que las
    coordenadas de source correspondan a las coordenadas (52, 64) y (76, 64),
    y el tamaño de la imagen sea cambiado a  $128 \times 128$ ;
4 geo  $\leftarrow$  aplicar la matriz de transformación M a image;
5 geo  $\leftarrow$  aplicar mejora de iluminación;
6 faceLandmark  $\leftarrow$  usar filter para filtrar puntos fiduciales de interés;
7 graph  $\leftarrow$  generar grafo de puntos a partir de faceLandmark;
8 graph  $\leftarrow$  aplicar la matriz de transformación M a coordenadas de vértices de
    graph;
9 para cada vertice en graph hacer
10   | extraer Gabor Jet de las coordenadas del vértice usando el conjunto de
      | máscaras de Gabor incluidas en mask;
11   | Almacenar Gabor Jet dentro de graph en relación con vertice;
12 devolver graph;
```

4.5. Filtro y reajuste de puntos detectados

Mientras que **EBGM** propuesto en [Bolme \(2003\)](#) funciona con 25 puntos fiduciales, de los cuales 3 de ellos se refieren al cabello, usando el detector de puntos **CLNF** se determina 68 puntos de los cuales muchos son cercanos unos a otros por lo que extraer características en todos ellos resulta redundante, ademas ninguno de los puntos obtenidos a través de **CLNF** corresponden a los 3 puntos que se refieren al cabello.

Por lo que es necesario un filtrado de puntos para realizar una correspondencia con los puntos con los que trabaja **EBGM**, por ello se deja de trabajar con estos 3 puntos fiduciales, siendo la cantidad efectiva de trabajo reducida de 25 a 22 puntos, Figura 4.8.

Después del filtrado toda la información se completa a un grafo y se aplica una transformada de perspectiva obtenida en el paso anterior para que los vértices correspondan a la imagen normalizada y se continua el proceso de extracción de características de **EBGM**.

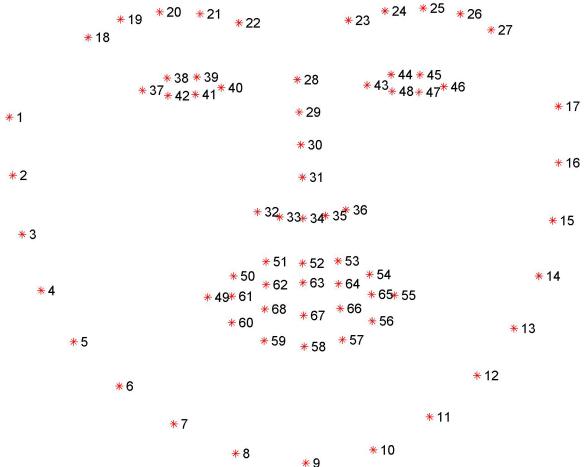


Figura 4.7: Puntos fiduciales entregados por CLNF, extraído del proyecto “Open Face”

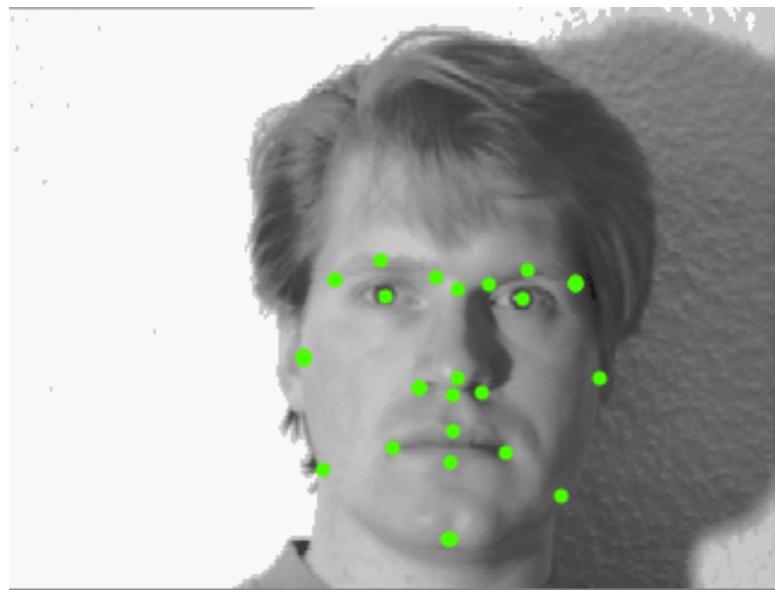


Figura 4.8: Puntos fiduciales después del proceso de filtrado

4.6. Proceso de EBGM

Después de reemplazar el proceso de localización de puntos, se continua con el proceso de reconocimiento explicado en la Sección 3.3 el cual se puede entender en el Algoritmo 2. El proceso empiza cuando extraen características de los puntos fiduciales a través de las convoluciones con las máscaras de Gabor, donde el grupo de coeficientes de resultado son almacenados en una estructura tipo grafo.

Finalmente cada imagen representada como un grafo que es comparada con las imágenes de entrenamiento en una comparación una a uno usando las Ecuaciones 3.20 y 3.21, y eligiendo al grafo con el cual posee mayor similitud. Este proceso es descrito en el Algoritmo 3.

Una vez obtenido un resultado de reconocimiento este es visualizado mostrando el nombre del sujeto reconocido sobre el área de su rostro detectado.

Algoritmo 3: Función para comparar un Face Graph con el conjunto de Face Graph de entrenamiento

```
1 ReconocerFaceGraph(faceGraph, graphs, names);
2 menor ← 10;
3 para i ← 0, i<tamaño de graphs hacer
4     distance ← medir similitud entre faceGraph y graphs[i];
5     si distance<menor entonces
6         name ← names[i];
7 devolver name;
```

4.7. Consideraciones finales

El aporte de la propuesta es todo el *pipeline* donde se ha logrado aplicar **EBGM** para su uso de vídeo vigilancia gracias al uso de una de las ultimas propuestas en una linea de investigación para el reconocimiento de puntos de interés (**CLNF**), donde el detalle de toda la propuesta puede observarse en la Figura 4.1.

Se resalta el uso de mejoras para la iluminación usando una unión de varios procesos conocidos, así mismo las transformaciones de tamaño sobre el rostro detectado para poder aproximarnos a resoluciones que son comunes en este escenario particular.

Se realizaron varias pruebas para llegar a la propuesta final, muchas de las cuales demuestran las dificultades que presenta en el reconocimiento de rostros en vídeo vigilancia y la imposibilidad de usar algunas técnicas en este contexto

Capítulo 5

Resultados

En este capítulo se muestran los resultados de investigación de la presente tesis, el presente capítulo se divide en dos partes. La primera parte son las base de datos con las que se ha trabajado, donde las tres primeras fueron utilizadas para las pruebas iniciales de la investigación, para realizar comparativas entre métodos de reconocimiento de rostro y el análisis paramétrico que se realizó después, mientras que la ultima base de datos fue hecha para las pruebas finales de la propuesta, finalizando la primera parte se explica el método de experimentación que se uso para todas las pruebas

La segunda parte muestra varias secciones de resultados como: Comparación de **EBGM** con otros métodos holísticos, Evaluación paramétrica de **EBGM**, Evaluación en el *pipeline* y la pruebas de la propuesta. Las dos primeras secciones se enfocan en **EBGM**, en comparación con otros métodos y el estudio de su funcionamiento a traves de una evaluación paramétrica, la tercera sección se enfoca en prueba de modificaciones propuestas al *pipeline* explicado en el Capítulo 4, y finalmente las ultimas secciones se dedican a mostrar los resultados finales de la propuesta en imágenes obtenidas de cámaras de seguridad y su aplicación para reconocer imágenes ofuscadas.

5.1. Base de datos

A continuación se menciona una descripción de las bases de datos usadas en las pruebas de este capítulo y una creada a partir de imágenes obtenidas de una cámara de seguridad para la prueba de la propuesta final.

5.1.1. AT&T

Cuenta con diez imágenes diferentes por cada uno de los 40 sujetos que componen dicha base de datos([Cambridge \(2002\)](#)) . En algunos sujetos, las imágenes fueron

tomadas en diferentes momentos , las imágenes presentan variación de la iluminación, de las expresiones faciales (ojos abiertos / cerrados, sonriendo / sin sonreír) y de los detalles faciales (lentes / sin gafas) . Todas las imágenes fueron tomadas contra un fondo homogéneo oscuro con los sujetos en posición vertical , frontal (con tolerancia para un cierto movimiento lateral). Se puede observa una muestra de esta base de datos en la Figura 5.1.



Figura 5.1: Muestra de imágenes que conforman la base de datos AT&T

5.1.2. Yale A

Esta base de datos([Georghiades et al. \(1997\)](#)) contiene 165 imágenes en escala de grises en formato GIF de 15 individuos. Hay 11 imágenes por sujeto, una por cada diferente expresión facial o cambio de iluminación: iluminación central , con gafas, feliz, iluminación izquierda, sin gafas ,expresión normal , iluminación derecha , triste , somnoliento , sorprendido , y guiño. Algunas de estas configuraciones pueden apreciarse en la imagen 5.2

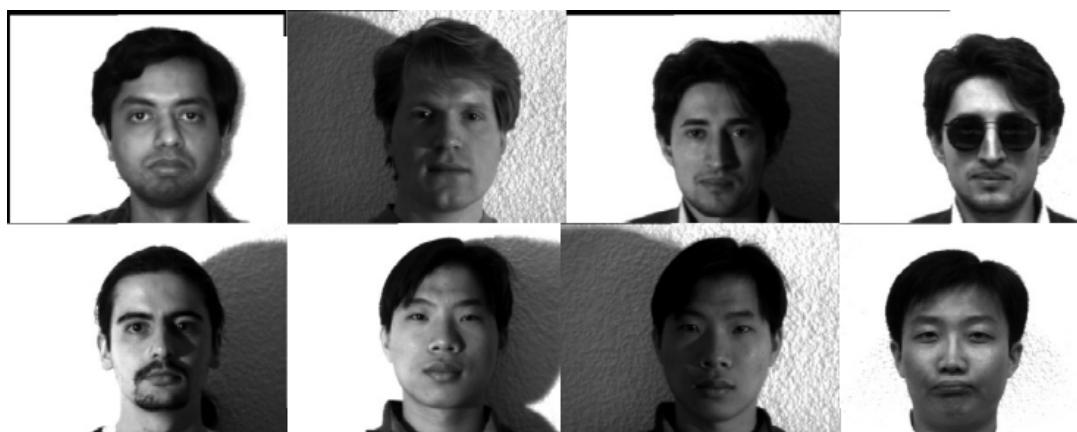


Figura 5.2: Muestra de imágenes que conforman la base de datos Yale A

5.1.3. Georgia Tech

La base de datos([Tech \(1999\)](#)) contiene imágenes de 50 personas y se almacena en formato JPEG. Para cada individuo, hay 15 imágenes a color capturadas entre el 06/01/99 y el 15/11/99. La mayoría de las imágenes fueron tomadas en dos sesiones diferentes para tener en cuenta las variaciones en las condiciones de iluminación, la expresión facial y la apariencia. Además de esto, los rostros fueron capturados en diferentes escalas y orientaciones.

Esta base de datos a diferencia de las dos descritas anteriormente es especialmente desafiante por que no es normalizada ni recortada y como se puede observar en la Figura [5.3](#) donde el proceso de adquisición es más informal.



Figura 5.3: Muestra de imágenes que conforman la base de datos Georgia Tech

5.1.4. Base de datos generadas a partir de imágenes de cámara de vigilancia

Se produjo una base de datos de una cámara de seguridad enfocando la escena vista en la Figura [5.18](#), con el objetivo de tener un referente de prueba lo mas cercano a la realidad.

La base de datos consiste en 24 sujetos con 8 imágenes cada uno separadas en dos grupos mañana y medio día para probar la variaciones en iluminación, en la Figura [5.4](#) podemos ver una muestra de la base de datos.

5.2. Método de experimentación

Como método de experimentación aplicamos un proceso en cada base de datos que consiste en dividir las imágenes de cada individuo en grupos de 3 a 4 imágenes por sujeto, dependiendo de la base de datos y cada grupo fue usado como entrenamiento mientras el resto es usado como prueba. Este proceso se repite hasta que todos los



Figura 5.4: Muestra de imágenes que conforman la base de datos obtenida a partir de una cámara de seguridad

grupos hayan sido usados como entrenamiento, siendo la cifra final el promedio de los resultados de cada grupo.

La razón para la elección de grupos de entrenamiento tan reducidos es poder acercarnos al escenario de video vigilancia donde en raras ocasiones se cuenta con varias imágenes de los sujetos a identificar.

A continuación se realiza una comparativa del método de reconocimiento usado en la propuesta y otros métodos extraídos del estado del arte.

5.3. Comparación de **EBGM** con métodos holísticos

Se realizó una comparación de **EBGM** con otros métodos de reconocimiento holísticos usados en la literatura como; PCA([Turk and Pentland \(1991\)](#)), KFA([/\(Yang \(2002\)\)](#) y LDA([Zhao et al. \(1999\)](#)), todos ellos implementados en matlab por The PhD Toolbox [Struc \(2012\)](#) para demostrar el rendimiento de estos algoritmos en varias situaciones. Esta comparación se realizo con las tres primeras bases de datos para establecer una diferencia entre el método escogido en la propuesta y otros métodos del estado del arte.

Como se puede apreciar en el Cuadro 5.1 y mejor aun en la Figura 5.6. **EBGM** tiene una tasa de aciertos igual o en algunos casos mejor que los demás métodos comparados. En ninguna de las pruebas realizadas con las tres bases de datos **EBGM** cae por debajo de la tasa de aciertos promedio, esta es una de las razones por la cual se elige a **EBGM** como el método de reconocimiento de rostros en el cual se centra el

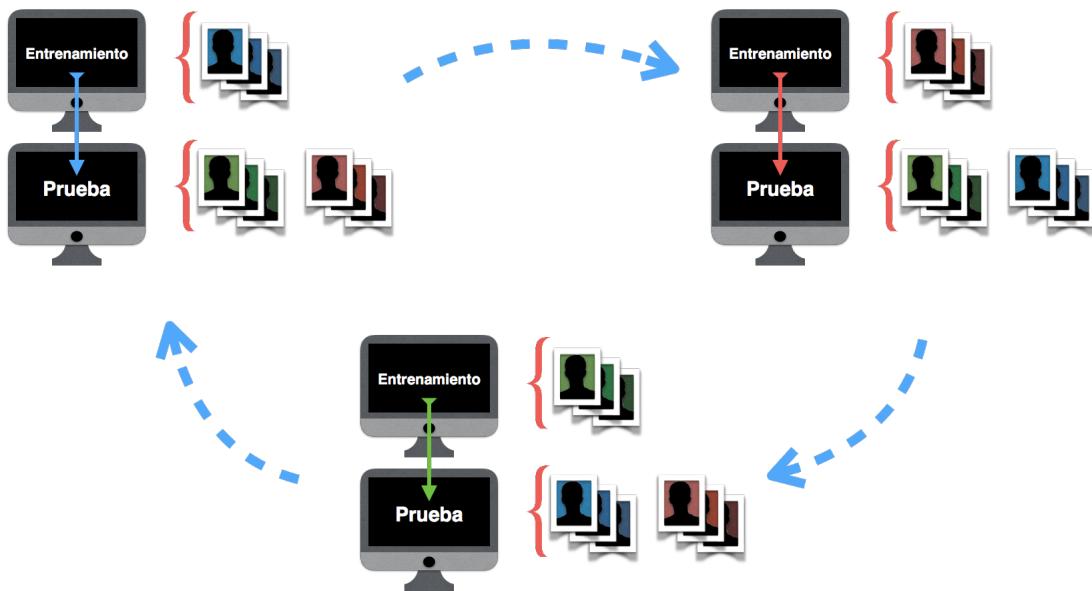


Figura 5.5: Muestra del proceso de experimentación, donde las imágenes se dividen por grupos y todos los grupos rotan hasta que todas las combinaciones de prueba y entrenamiento sean probadas

Cuadro 5.1: Comparación con bases de datos ATT, Yale A y Georgia

	ATT/ORL	YALE A	Georgia
PCA	88.03 %	88.18 %	76.84 %
KFA	87.78 %	91.72 %	75.03 %
LDA	86.02 %	90.21 %	75.50 %
EBGM	91.43 %	97.94 %	77.80 %

trabajo de esta tesis.

5.4. Evaluación paramétrica de EBGM

El método elegido para el proceso de reconocimiento de rostros es **EBGM**, este ha sido elegido por sus resultado en la comparativa realizada en la Sección 5.3, también por ser un método que es considerado biométrico por lo expuesto en el Capítulo 3.

Parte de las contribuciones de este trabajo de tesis consiste en una evaluación de parámetros de **EBGM** para poder incrementar la tasa de aciertos, entiéndase como tasa de acierto al porcentaje respecto a los verdaderos positivos. Finalmente probar el resultado en un ambiente de video vigilancia.

Se puede observar el curso normal de **EBGM** en la Figura 5.7, y las evaluaciones que se plantean son las siguientes:

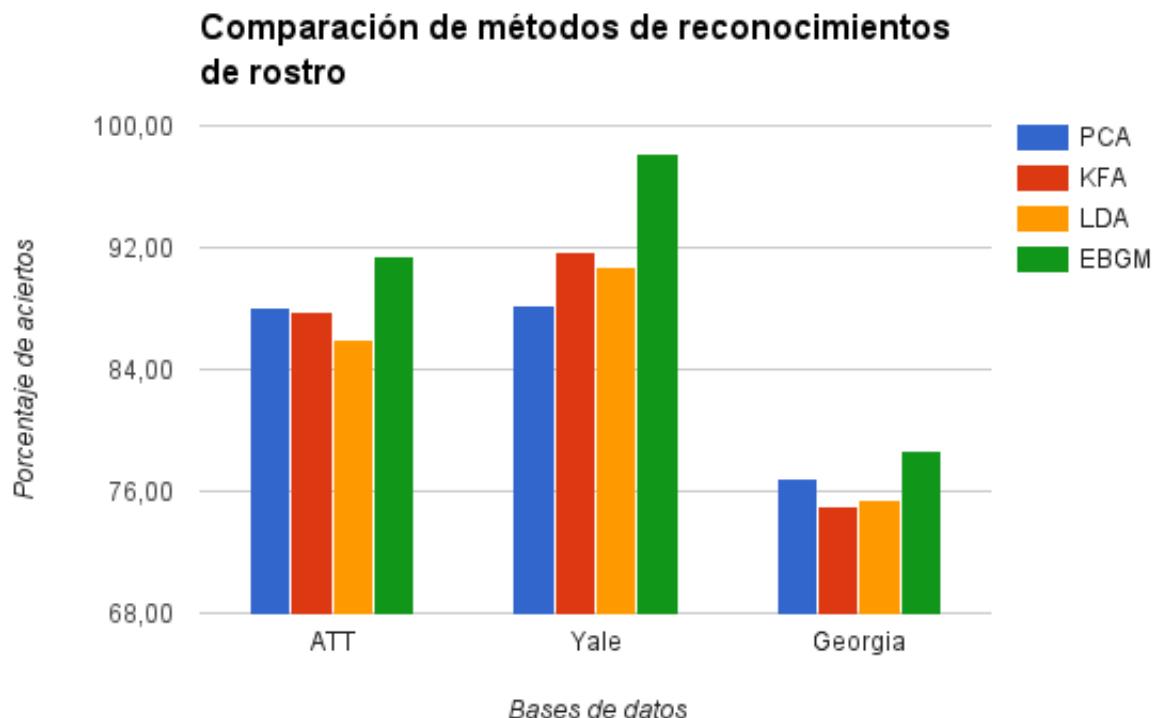


Figura 5.6: Comparación de EBGM con otros algoritmos, la escala empieza en 68,00 para que pueda apreciarse las diferencias entre los métodos

5.4.1. Incrementar la cantidad de modelos

En el trabajo de Bolme (2003) se usa 70 modelos de rostros de la base de datos FERET gray, el numero de modelos es el mismo que se usa en el trabajo de Wiskott et al. (1997) al que hace referencia.

Para la creación de los modelos se usó la nueva versión de FERET llamada FERET COLOR y se ha elegido nuevas imágenes aleatoriamente para extraer manualmente las coordenadas de los puntos fiduciales, así mismo se ha incrementado el numero de modelos a 150 con el fin de probar si dicho incremento tiene una relación directa con la tasa de aciertos.

Cuadro 5.2: Resultado aciertos con incremento de modelos en EBGM

	AT&T/ORL	Yale A	Georgia
EBGM original	91.43 %	97.94 %	77.80 %
150 Modelos	91.43 %	98.21 %	78.73 %

Como se puede observar en el Cuadro 5.2 existe una mejora en los aciertos, no obstante un incremento en el numero de modelos no implica una mejora proporcional en la tasa de aciertos, lo cual indica que existe un punto en que la cantidad de modelos es suficiente para realizar una estimación precisa.

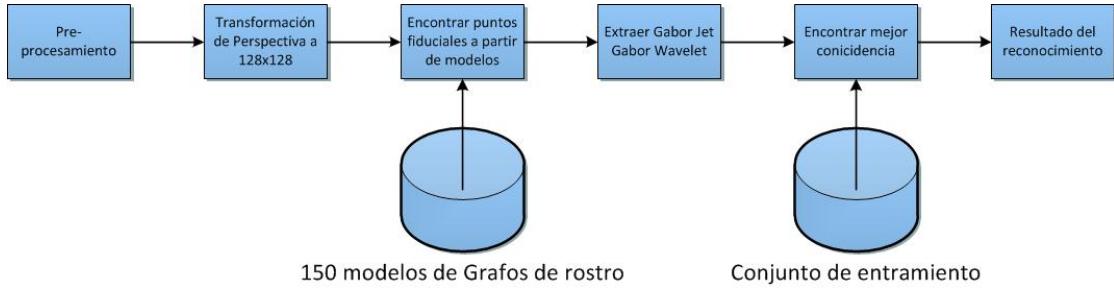


Figura 5.7: Curso de procesos que forman parte del reconocimiento de rostros en [EBGM](#)

5.4.2. Modificar la función de similitud

Usando como base el trabajo de [Bolme \(2003\)](#), en este trabajo se tiene como parte de sus conclusiones y recomendaciones la modificación de la función de similitud con la que se compara los Face Graph que son las representaciones de un rostro en dicho algoritmo.

La modificación que se sugiere investigar es la de añadir pesos a cada punto fiducial que se compara, ya que se puede concluir que no todos los puntos de un rostro son igualmente importantes para realizar un reconocimiento.

Un ejemplo, es que los puntos del contorno del rostro o el cabello pueden ser menos importantes que los puntos de los ojos o la nariz, así que se añade un vector de pesos a la función de similitud descrita en la ecuación 3.20 y 3.21 y probamos varias combinaciones empíricas bajo el concepto que la información con mayor relevancia se encuentra en la boca, nariz y ojos .

$$L_{jet}(G, G', W) = \frac{1}{N} \sum_{i=0}^N S(J_i, J'_i, w_i) \quad (5.1)$$

$$S(J, J', w) = w * \frac{\sum_{j=1}^N a_j a'_j \cos(\phi_j - \phi'_j)}{\sqrt{\sum_{j=1}^N a_j^2 \sum_{j=1}^N a'^2_j}} \quad (5.2)$$

En las Ecuaciones 5.1 y 5.2 podemos observar la adición de un vector W de tamaño N con elementos w , lo que permite dar una valoración diferente a cada punto fiducial.

En el Cuadro 5.3 se observa la agrupación que se realizan a los puntos fiduciales y las distribuciones mostradas como porcentajes, donde se le da importancia a áreas que se ubican en el centro del rostro, en el cuadro 5.4 se puede ver su correspondencia en valores numéricos para la conformación del vector W usado en la Ecuación 5.1.

En el Cuadro 5.5 se observan los resultados de las configuraciones propuestas para modificación la función de similitud de [EBGM](#), en la figura 5.8 vemos que la primera configuración ofrece una mejora de 2 % en la base de datos Georgia y mantiene el rendimiento en el resto de las bases de datos. Esta configuración distribuye los pesos

Cuadro 5.3: Tabla de distribución de pesos propuesta

Área	Nro puntos	Conf. 1	Conf. 2	Conf. 3
Ojos y puente de la nariz	3	20 %	50 %	15 %
Cejas	6	20 %	10 %	15 %
Nariz	4	20 %	20 %	20 %
Boca	4	20 %	10 %	25 %
Bordes de la cabeza	5	10 %	5 %	5 %
Barbilla y quijada	3	10 %	5 %	20 %

uniformemente a todas las áreas del centro del rostro en detrimento de los bordes como son la barbilla y la quijada.

En el resto de las configuraciones se observan resultados variados donde se presentan mejoras del 1% en las tres bases de datos con excepción de la tercera configuración donde se le resta importancia a los ojos, para probar la importancia del centro del rostro como fuente de información.

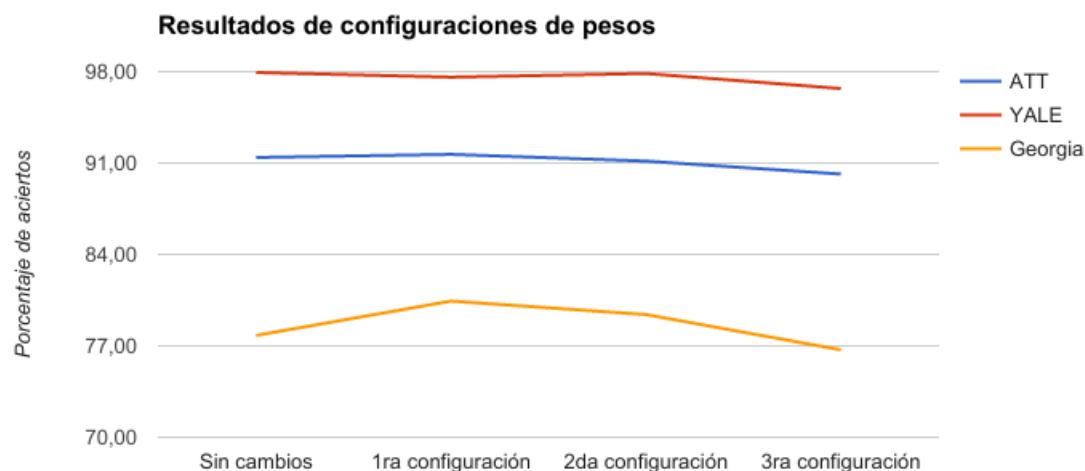


Figura 5.8: Comparación entre las configuraciones propuestas para los pesos en la función de similitud de **EBGM**

5.4.3. Incrementar la cantidad de imágenes de entrenamiento a través de transformaciones de perspectiva

Una de las restricciones más grandes que conlleva el reconocimiento de rostros en vídeo vigilancia, es el hecho de que en muchos sistema de vigilancia no se cuenta con varias imágenes por sujeto para realizar un entrenamiento adecuado. En muchos casos solo se cuenta con una imagen, ya puede ser de un carnet o una fotografía de perfil y en varios casos son imágenes que no están al día.

CAPÍTULO 5. Resultados

Cuadro 5.4: Valores del vector W de pesos derivado del Cuadro 5.3

Puntos Fiduciales	1ra Config	2da Config	3ra Config
ojo izquierdo	0.07	0.07	0.05
ojo derecho	0.07	0.07	0.05
puente de la nariz	0.06	0.07	0.05
pico ceja derecha	0.03	0.025	0.025
pico ceja izquierda	0.03	0.025	0.025
interior ceja derecha	0.04	0.025	0.025
interior ceja izquierda	0.04	0.025	0.025
exterior ceja derecha	0.03	0.025	0.025
exterior ceja derecha	0.03	0.025	0.025
punta de la nariz	0.05	0.08	0.05
centro base de la nariz	0.05	0.07	0.05
base derecha de la nariz	0.05	0.07	0.05
base izquierda de la nariz	0.05	0.07	0.05
parte central superior de la boca	0.05	0.0375	0.0625
parte central inferior de la boca	0.05	0.0375	0.0625
esquina izquierda de la boca	0.05	0.0375	0.0625
esquina derecha de la boca	0.05	0.0375	0.0625
parte central superior de la cabeza	0.02	0.02	0.01
parte izquierda superior de la cabeza	0.02	0.02	0.01
parte derecha superior de la cabeza	0.02	0.02	0.01
borde izquierdo de la cara	0.02	0.02	0.01
borde derecho de la cara	0.02	0.02	0.01
centro de la barbilla	0.04	0.04	0.06
parte izquierda de la quijada	0.03	0.03	0.07
parte derecha de la quijada	0.03	0.03	0.07

Por ello se plantea crear más imágenes de entrenamiento a partir de una sola imagen a través de varias transformaciones de perspectiva para probar si con una imagen frontal podemos crear imágenes de rostros que parezcan mirar a otros sentidos y así poder aumentar el numero de entrenamiento y con ello el porcentaje de aciertos.

Para lograr ello generamos una matriz de transformación en 3D donde realizamos las rotaciones necesarias en el eje Z para crear la ilusión de cambio de pose, después realizamos un conversión a un matriz de transformación 2D y la aplicamos a la imagen. Los resultados de dicha transformación se pueden apreciar en la Figura 5.10.

En las pruebas hechas a **EBGM**, no se muestra ningún cambio y en los resultados experimentales. el uso de las trasformaciones no ayudan a mejorar la tasa de aciertos pero tampoco la empeoran. En ningnú caso una imagen generada por la transformación de perspectiva es elegida como resultado del proceso de reconocimiento.

Cuadro 5.5: Resultados de adición de pesos **EBGM**

Propuestas	AT& T	Yale	Georgia
EBGM sin cambios	91.43 %	97.94 %	77.80 %
Primera configuración de pesos	91.67 %	97.58 %	80.43 %
Segunda configuración de pesos	91.15 %	97.86 %	79.40 %
Tercera configuración de pesos	90.16 %	96.71 %	76.70 %

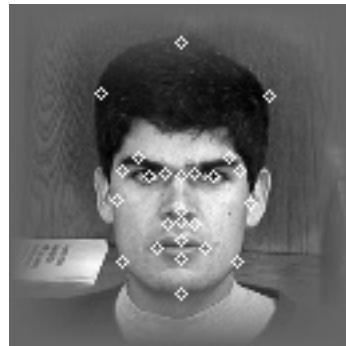


Figura 5.9: Imagen de entrenamiento original

5.4.4. Ajuste de tamaño a máscaras de Gabor

La Ecuación 3.14 define el Gabor Wavelet que se usa para crear máscaras de Gabor. Con esta ecuación podemos crear varios tamaños de máscaras, donde la configuración original es $N \in \{25, 37, 51, 71, 101\}$. En conjunción al resto de parámetros explicados en la Sección 3.3.1 se obtienen 80 configuraciones de Gabor Wavelet y siendo efectivas 40 máscaras por punto fiducial, debido a que existe una mascara que extrae la parte imaginaria y otra la parte real del Wavelet.

La propuesta de modificación plantea el cambio del tamaño de las Gabor Wavelet, debido a que tamaños de 71 y 101 son demasiado grandes teniendo en cuenta que la imagen a convolucionar es de 128×128 y el tamaño efectivo del rostro es aun menor siendo los tamaños propuestos $N \in \{13, 19, 25, 35, 51\}$ los resultados pueden observarse en el Cuadro

Cuadro 5.6: Resultados de pruebas en tamaños de máscaras de Gabor

	AT&T	Yale	Georgia
Sin cambios	91.43 %	97.94 %	77.80 %
Propuesta de máscaras de Gabor	93.18 %	97.58 %	79.43 %

Se observa mejoras de 2 % en las bases de datos AT&T y Georgia, mientras que en Yale mantiene el porcentaje de aciertos. Un efecto a parte de la mejora en el acierto es que al tener un tamaño de máscaras menor el tiempo en convoluciones también se disminuye.



Figura 5.10: Resultados de las transformaciones de perspectiva

5.5. Evaluaciones en el *pipeline* de reconocimiento

Como parte de la propuesta es necesario encontrar un método de validación para confirmar las detecciones de rostros proporcionadas por el algoritmo de Viola-Jones, en esta sección se muestran resultados sobre las opciones exploradas para afrontar este problema

5.5.1. Evaluación del detector de Viola-Jones para detección de ojos

Para poder usar **EBGM** tal como se presenta en [Bolme \(2003\)](#) y [Wiskott et al. \(1997\)](#) en un *pipeline* de vídeo vigilancia es necesario proporcionar las coordenadas de los ojos.

Para ello se necesita establecer un proceso o una forma de detección de ojos con una gran robustez, ya que un error en la detección de ojos resulta en un fallo de todo el proceso de reconocimiento

Como parte de este trabajo se realizó pruebas de un detector de Viola-Jones entrenado para detectar ojos como opción para encontrar coordenadas de ojos y empezar el proceso de reconocimiento. Para ello se probó el Algoritmo 4.



Figura 5.11: Ejemplo de proceso de detección de ojos Nro 1

Como se puede observar en las Figuras 5.11 y 5.12 el proceso que propone el

Algoritmo 4: Proceso de detección de ojos en rostro para **EBGM**

```

1 face ← rostro detectado por un detector de rostros;
2 eyes ← lista de ojos detectados en face;
3 para  $i \leftarrow 0, i < \text{tamaño de } eyes$  hacer
4   si la posición de  $eyes[i]$  no se encuentra en la región superior central de
      face entonces
5     descartar  $eyes[i]$  de eyes;
6 region ← dividir la región superior central de face en cuatro pares de
    sub-regiones ;
7 agrupar todos los ojos detectados en eyes segun su posicion dentro de
    region;
8 para cada region hacer
9   calcular el punto promedio a partir todos los eyes en region;
10 Elegir el par de puntos perteneciente al par de region según la mayor
    cantidad de puntos que posee;
```

algoritmo 4 ofrece resultados variados, el motivo de ello es la naturaleza del algoritmo de Viola-Jones, los ojos poseen menos variaciones de sombras si se lo compara con un rostro y ademas el calculo imagen integral es dependiente a la iluminación de la imagen.

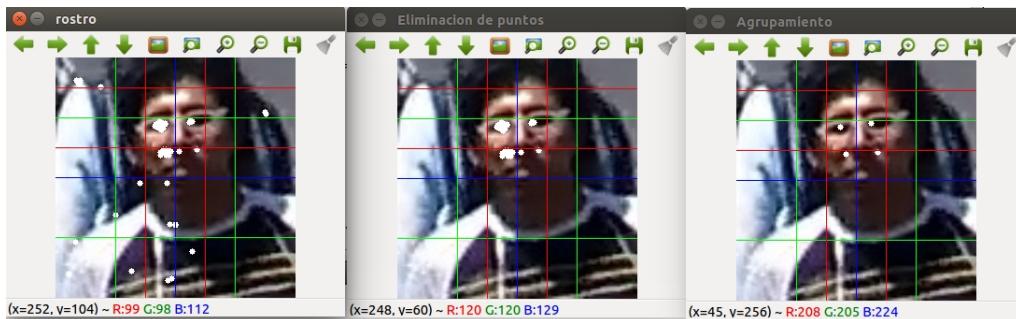


Figura 5.12: Ejemplo de proceso de detección de ojos Nro 2

5.5.2. Evaluar validación de detecciones mediante color de piel

Se probó como método de validación de las detecciones proporcionadas por Viola-Jones, el uso de un método de confirmación a través de detectores de color de piel el cual se detalla en el algoritmo 5.

Realizando pruebas en videos como se puede ver en la figura 5.13, se observan problemas de validación debido a que los colores de piel están entre 0 y 70 en canal Hue, pero también están otros colores como el naranja se encuentran en ese mismo rango debido a la variación de los otros dos canales. Así mismo todos los colores de piel humana no se encuentran agrupados en una sola región del rango de todo el canal, sobre todo colores oscuros de piel.

Algoritmo 5: Validación de detecciones a través de color de piel

```

1 face ← rostro detectado;
2 imageHSV ← convertir face a espacio de color HSV;
3 binaryImage ← binarizar imagen bajo un threshold de 70 en el canal Hue
   de imageHSV;
4 count ← contar pixeles blancos de binaryImage;
5 si count>al 50 % del total de pixeles de face entonces
6   | validar face como imagen de un rostros;
7 en otro caso
8   | descartar face como imagen de un rostros;

```

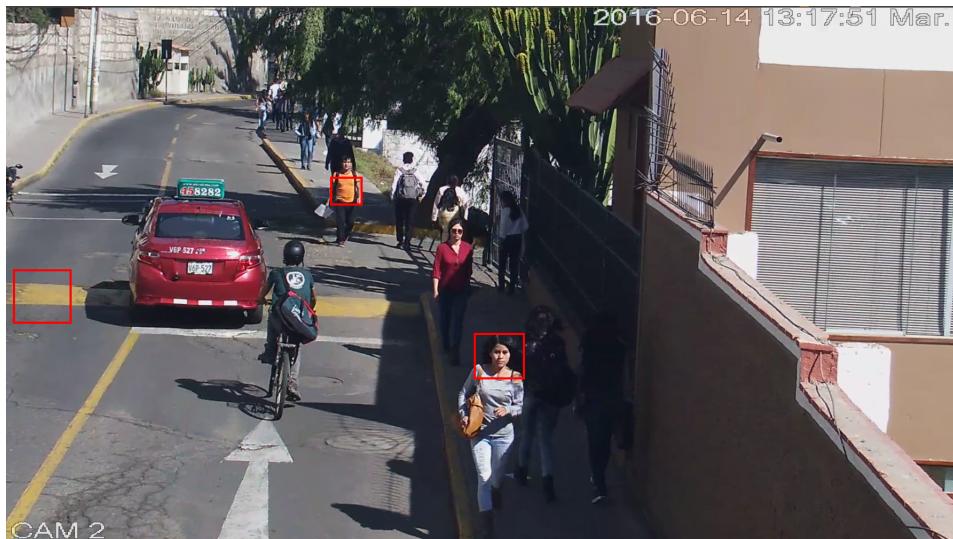


Figura 5.13: Ejemplo de falsas validaciones en colores cercanos al rojo y naranja.

5.5.3. Mejora de la iluminación en el pre-procesamiento

EBGM tiene un pre-procesamiento para normalizar la información introducida, pero este pre-procesamiento solo consiste en una normalización de los valores de los pixeles de las imágenes, una transformación de tamaño a un estándar de 128×128 y la añadidura de bordes de 30 pixeles a la imagen. El uso de esto bordes se justifican en el trabajo de Bolme (2003) como una forma de ayudar a que su ajuste de puntos fiduciales converja en el centro de la imagen y no tienda a alejarse a los borde, y como método para filtrar la información del fondo que pueda quedar después de la transformación de perspectiva aplicada.

Para cubrir esta falta de una mejora en iluminación, la propuesta explicada en la Sección 4.4 implica la modificación de este proceso para así volverlo más robusto a cambios de luz.

La unión de procesos expuesta en la ecuación 4.1 es comparada con el algoritmo sin cambios y con la ecualización de histograma, los resultados se pueden observar en el

Cuadro 5.7: Resultados de la propuesta de iluminación

	AT&T	Yale	Georgia
EBGM sin cambios	91.43 %	97.94 %	77.80 %
Ecualización de histograma	92.78 %	96.67 %	79.13 %
Propuesta de iluminación	92.82 %	98.53 %	78.93 %

Cuadro 5.7 y la Figura 5.14 donde existen mejoras del 1 % en comparación al algoritmo original y en la base de datos Yale donde la presencia de grandes contrastes afectan el reconocimiento, la ecualización de histograma obtiene un resultado inferior al algoritmo original, mientras que la propuesta logra una mejora de 1 %.

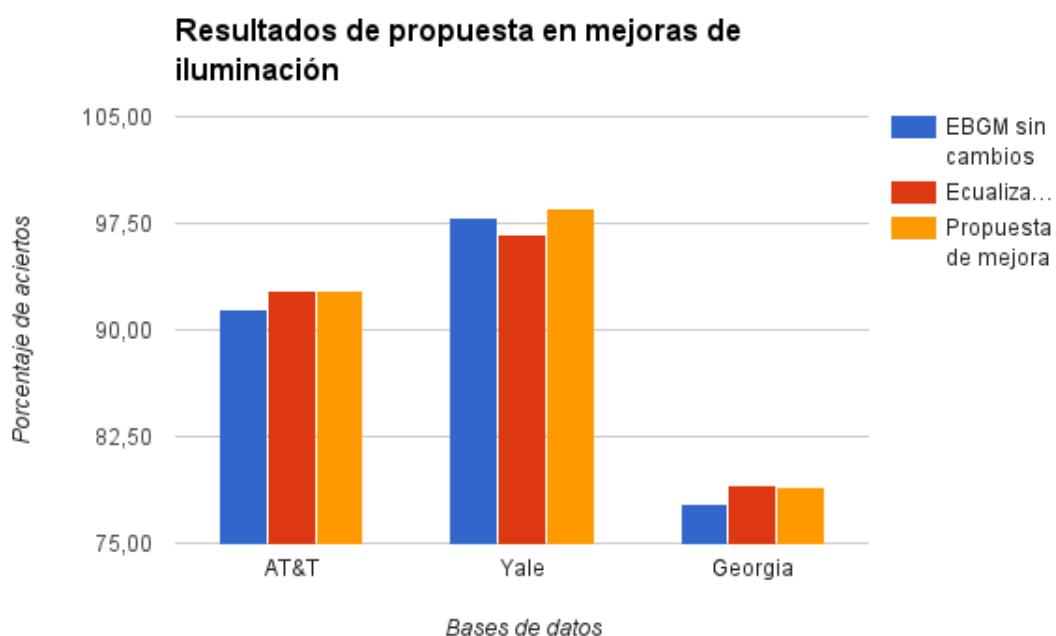


Figura 5.14: Comparación entre propuesta, ecualización de histograma y EBGM

5.5.4. Tracking de rostros

Se hizo pruebas con el algoritmo de Camshift como método de *tracking*, al igual que el detector de piel presenta los problemas relacionados al color, ya que se basa en información de color proporcionada por la región inicial y en la siguiente escena busca la misma proporción de información en regiones cercanas.

En las figuras 5.15 y 5.16 se puede observar otro comportamiento de camshift, si objetos del mismo color se acercan al área de *tracking*, dicha área se incrementa e incluso puede ser tomada por otro patrón de color

Por las razones explicadas, se opta por usar un enfoque de *tracking* a través de



Figura 5.15: Ejemplo de tracking usando Camshift

detecciones continuas usando el algoritmo de Viola-Jones donde errores ocasionados por los cambios de iluminación son menores que los que presenta camshift.

5.5.5. Evaluar la detección de puntos fiduciales

EBGM usa modelos hechos a mano como base para encontrar puntos fiduciales, este proceso empieza con un grafo promedio generado a partir de todos los modelos, es necesario proveer las coordenadas de los ojos, a partir de esto dos punto se calcula un tercero a partir de las distancias proporcionadas por el grafo promedio y el proceso se repite a través de todos los puntos, después hace una comparación de similitud por cada punto ubicado con el mismo punto en todos los moldes, se elige la información de coordenada del molde más similar y se estima el nuevo valor del punto calculando una desviación entre el punto del grafo promedio y del obtenido a partir del modelo incluyendo el grado de similitud dando el punto final donde se extraerá las características.

5.5.5.1. Comprobación entre punto manuales y puntos encontrados por **EBGM**

EBGM depende de los modelos manuales, si no existe un modelo lo suficientemente parecido a la imagen es probable que no se realice una buena estimación de los puntos fiduciales, otro problema es que en un ambiente de vídeo se depende de un detector rostros y probablemente un detector de ojos, o uno que realice ambos trabajos lo que hace depender el resultado del reconocimiento de la correcta detección de los rostros y ojos en vídeo.

Para probar en cuanto afecta una correcta localización de puntos fiduciales se

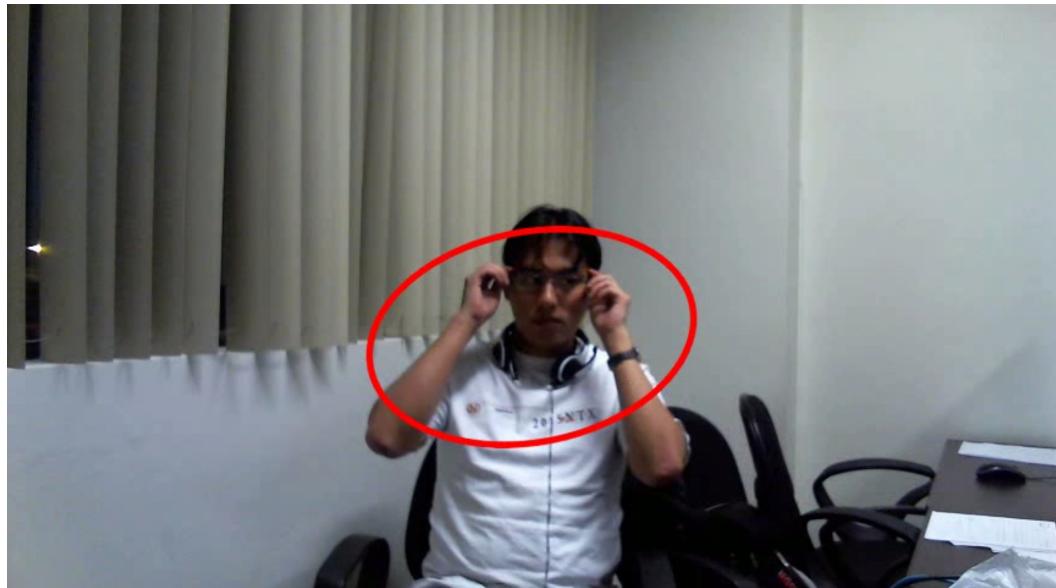


Figura 5.16: Ejemplo de como el área de tracking se expande debido a áreas circundantes del mismo color

realizo la siguiente prueba: de los resultados obtenidos de la base de datos Georgia elegimos 8 sujetos con los peores resultados de aciertos y detectamos los punto fiduciales manualmente, después comparamos con el resultado si los puntos fiduciales hubiera sido localizados con los moldes.

Cuadro 5.8: Comparación de puntos fiduciales

	3 Rostros	2 Rostros	1 Rostro
Sin Cambios con modelos	65.75 %	66.62 %	57.14 %
Puntos manuales Entrenamiento aleatorio	77.92 %	74.31 %	64.11 %
Puntos manuales Entrenamiento escogido	91.67 %	81.73 %	80.36 %

Como se observa en la Cuadro 5.8 tenemos la primera fila que es el resultado de **EBGM** sin ningún cambio, en la segunda tenemos los resultados usando una localización manual en vez de los moldes originales y la ultima fila es el mismo resultado donde elegimos una imagen de perfil y dos imágenes mirando a ambos lados, las columnas se refieren a la cantidad de rostro que se usa como entrenamiento. Para esta prueba se utilizó solo la base de datos Georgia debido a como muestran las pruebas anteriores esta base de datos muestra el porcentaje de aciertos más bajo.

Como se puede ver en las columnas del Cuadro 5.8 la primera representa el método de experimentación con el cual se ha llevado a cabo todos los experimentos, Sección 5.2, y las dos siguientes columnas son pruebas de cuanto mas se puede reducir el entrenamiento. Cabe resaltar que los resultados entre el algoritmo original y el uso de **CLNF** con el método de experimentación de las demás pruebas es de 65.75 % y 77.92 %

y cuando se escoge especialmente las imágenes de entrenamiento la tasa de aciertos aumenta a 91.67%, por lo que se puede observar que la forma de detección de puntos fiduciales es el punto mas débil de **EBGM**.

5.5.5.2. Uso de **CLNF** como detector de puntos fiduciales

La propuesta de esta tesis es usar **CLNF** como método de detección de puntos fiduciales en reemplazo para ello realizamos las pruebas solo en la base de datos Georgia por ser la base de datos que muestra el menor porcentaje de aciertos. En la sección anterior se probó que la localización manual es superior al método usado por el algoritmo original de **EBGM**.

Para probar que la propuesta es una mejor alternativa se hizo una comparación del algoritmo sin cambios y reemplazando el proceso de detección de puntos que usa **EBGM** con el algoritmo de **CLNF** esta vez usando la totalidad de la base de datos Georgia, en la figura 5.17 se aprecia una mejora del 1% con respecto al algoritmo original, ademas con el uso de **CLNF** se puede dar uso a **EBGM** sin los problemas mencionados en la sección 5.5.1

Cuadro 5.9: Resultado de uso de **CLNF** como detector de puntos fiduciales para **CLNF**

	Georgia
EBGM sin cambios	77.80 %
CLNF con 68 puntos	78.73 %
CLNF con 22 puntos	79.33 %

El uso de **CLNF** tal como lo propone [Baltrusaitis et al. \(2013\)](#) entrega un grupo de 68 puntos como se puede observar en la Figura 4.7 mientras que **EBGM** usa 25, Cuadro 3.1, pero entre estos dos grupos no existe correspondencia para los tres puntos que se refieren a cabello. Por lo que se realiza un filtrado para solo corresponder a 22 puntos fiduciales.

En los resultados expuestos en la tabla 5.9 y la figura 5.17 se observa que después del filtrado de puntos obtenemos mejores resultado que con la totalidad de puntos entregados por**CLNF**, esto se puede explicar por la eliminación de datos redundantes, gracias a ello se obtienen mejoras de 2% con una menor cantidad de puntos a convolucionar.

5.6. Prueba con base de datos de vídeo vigilancia

La propuesta final fue probada con imágenes extraídas de una cámara de video vigilancia, sección 5.1.4, para ello creamos dos grupo de datos de 12 sujetos con 8 imágenes cada uno, ambos grupos obtenidos en diferentes momentos del día uno en la mañana y otro en medio día, en la escena mostrada en la figura 5.18.

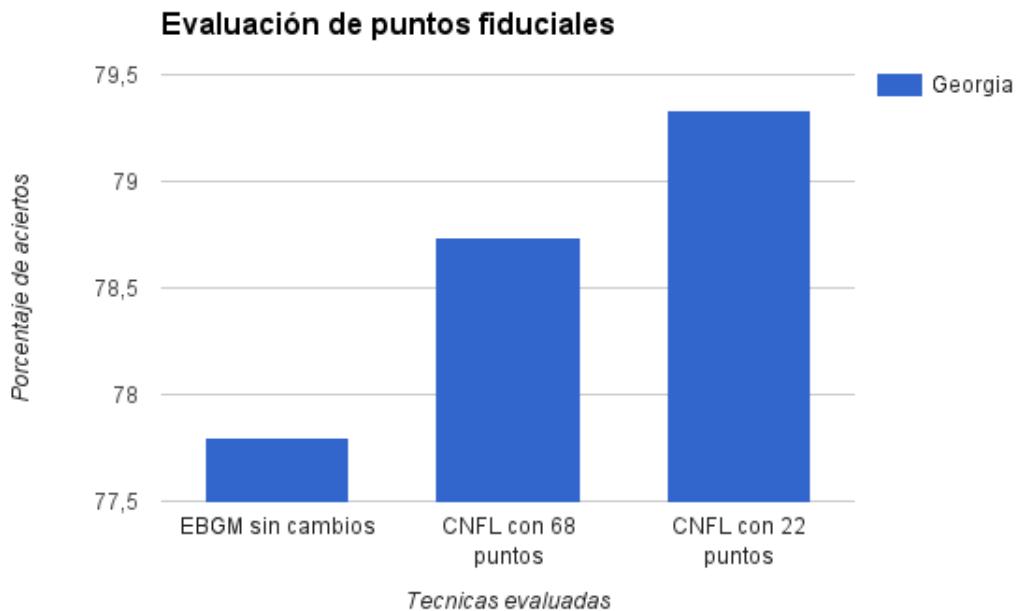


Figura 5.17: Resultado de pruebas de puntos fiduciales, donde CLNF con 22 es la propuesta final y CLNF con 68 es el resultado sin un filtrado de puntos

La adquisición de datos se realizó sin la colaboración de los sujetos para que los resultados de las pruebas se acerquen lo más posible a la realidad, a diferencia de otras bases de datos normalizadas. En la tabla 5.10 podemos ver los resultados de la propuesta comparados con el algoritmo original.

Cuadro 5.10: Resultados de la propuesta en comparación a EBGM sin cambios

	Mañana	Medio día
EBGM sin cambios	78.33 %	70.18 %
Propuesta	88.88 %	76.66 %

Por lo demostrado en la Sección 5.5.5 donde se muestra que el punto más débil de EBGM es la detección de punto fiduciales y la propuesta de usar CLNF obtiene una mejora del 2 %. En esta base de datos las mejora se puede apreciar mejor ya que a diferencia del resto de las bases los individuos en las imágenes no muestran cooperación alguna con el proceso de reconocimiento también se prueba en un ambiente no controlado donde si existe una mayor variación en lo concerniente a expresiones faciales y cambios de pose, logrando una mejora de 10 % y 6 % respectivamente.

La propuesta tiene un mejor desempeño que el algoritmo original ya que la forma de encontrar puntos propuesta es mejor debido a que mientras el algoritmo original solo hace una estimación simple del punto fiducial usando los moldes como referencia sin importar que tipo de punto sea, la propuesta usa una red neuronal especializada en cada punto. Finalmente la propuesta hace una evaluación de los puntos en conjunto para que puedas conformar un rostro factible cosa que el algoritmo original no hace.

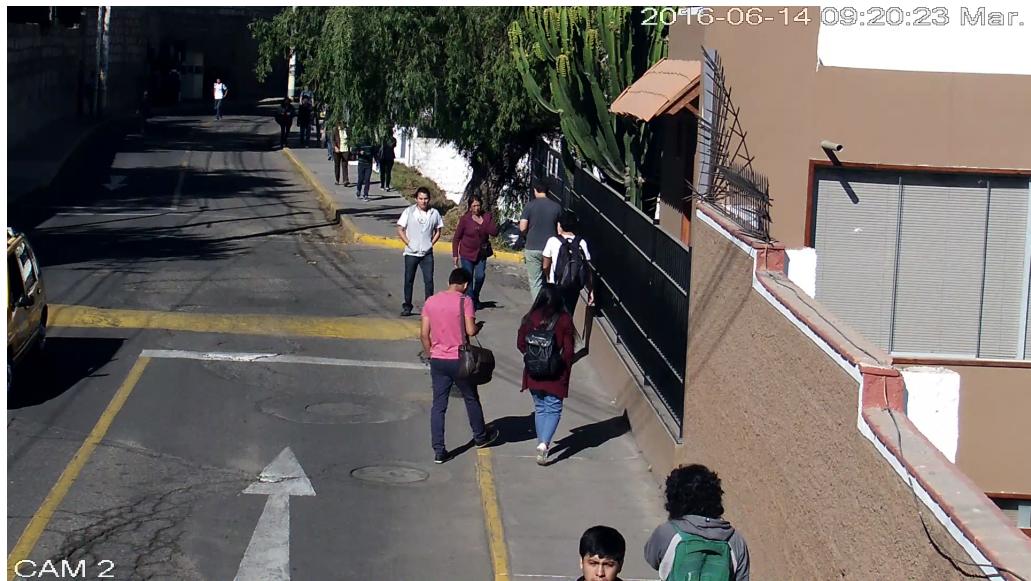


Figura 5.18: Escena de captura de imágenes para pruebas

5.7. Uso de la propuesta para reconocer imágenes ofuscadas

A finales de la investigación de esta tesis, se presento el trabajo de [McPherson et al. \(2016\)](#) donde se utilizan redes neuronales para reconocer rostros ofuscados, teniendo resultados alrededor del 95 % cuando la técnica de ofuscación es *mosaicing* es decir un pixelease pero obtiene un 57.75 % cuando se trata de un *blurring*.

Se ha realizado pruebas para demostrar que nuestra propuesta puede seguir reconociendo rostros al ser aplicada imágenes donde la información del rostro a sido ofuscada, es decir ocultada a través de algún tipo de técnica como lo es el "blurring".

5.7.1. Blurring

Oficialmente conocido como "Gaussian Blur", remueve detalles de una imagen aplicando un kernel Gaussiano a cada pixel. El resultado que se obtiene es una imagen suavizada, lo que impide que las personas puedan reconocer un rostro, como se aprecia en la figura 5.19.

5.7.2. Comparación de redes neuronales

EL resultado de nuestra prueba se compara con el trabajo de [McPherson et al. \(2016\)](#) donde se muestra resultados en TOP 1 donde solo se evalúa al primer resultado mas cercano. La única base de datos que usa es AT&T descrita en la sección 5.1.



Figura 5.19: Imagen antes y después de un proceso de *blurring*

Cuadro 5.11: Comparación entre propuesta y método propuesto en [McPherson et al. \(2016\)](#)

Metodo	Blurred Top 1
McPherson	57.75 %
Propuesta	83.42 %

Para replicar su experimento usamos la misma cantidad de datos de entrenamiento (8 imágenes por sujeto) y prueba (2 imágenes por sujeto). Todos los rostros fueron ofuscados antes de las pruebas, usando un Kernel Gaussiano de tamaño 64 para asemejarse al ofuscamiento usado por McPherson.

En este proceso de experimentación se localizan los puntos fiduciales en la imagen de entrenamiento y se alinean en la imagen ofuscada, debido a que en la imagen ofuscada no es posible encontrar puntos con la propuesta. Luego de la alineación de puntos se extraen las características siguiendo el curso normal del proceso de reconocimiento teniendo una mejora 26 % en el reconocimiento.

Como se observa en la tabla 5.11 la propuesta de esta tesis es superior al resultado obtenido por [McPherson et al. \(2016\)](#), esto se puede explicar al hecho que el proceso de *blurring* dispersa la información con el uso del kernel Gaussiano y las máscaras de Gabor las reúne para el proceso de reconocimiento. También la aplicación de *blurring* sobre la imagen del rostro a procesar no parece afectar la recolección de información en el espacio de las señales obtenidas tras la aplicación de las máscaras de Gabor. Finalmente hay que destacar que a diferencia del resto de pruebas, en esta se tienen 8 imágenes de entrenamiento un número mucho mayor que permite una mayor tasa de aciertos.

5.8. Consideraciones Finales

La propuesta ha logrado adaptar **EBGM** a un contexto de vídeo vigilancia gracias al reemplazo del uso de modelos para detectar los puntos de donde extraer características con **CLNF**, que es un detector basado en **CLM** donde la detección de puntos se realiza con una red neuronal en cada punto y el resultado final es evaluado en conjunto

CAPÍTULO 5. Resultados

para darle mayor robustez.

La adopción de nuestra propuesta mejora el porcentaje de 2 % de aciertos en las bases de datos Georgia y de 10 % y 6 % en las bases de datos obtenidas a partir de la cámara de seguridad.

Se ha realizado una evaluación de parámetros de **EBGM** donde se ha detallado varias formas de influir en el resultados final de **EBGM**.

Finalmente se menciona todas las opciones experimentadas para producir el *pipeline* final que presenta este trabajo de tesis.

Capítulo 6

Conclusiones, publicaciones y trabajos futuros

Finalmente en este capítulo se menciona las conclusiones, publicaciones y propuestas para trabajos futuro derivados del trabajo de tesis expuesto.

6.1. Conclusiones

En este trabajo de tesis se logró presentar un *pipeline* de reconocimiento de rostros en vídeo vigilancia, lográndolo mediante la modificación de **EBGM** con **CLNF**. Junto a ello se realizó una comparativa de varios métodos holísticos con **EBGM** probando que puede competir con ellos, también se realizó una evaluación de los parámetros que influyen en **EBGM** para mejorar el proceso de reconocimiento.

Se probó que las transformaciones de perspectiva para incrementar el conjunto de entrenamiento no tienen ningún efecto en el aumento de aciertos del reconocimiento.

Se demostró que la modificación propuesta usando **CLNF** como reemplazo para la función de detección de puntos permite el uso de **EBGM** en vídeo, cubriendo el punto mas débil del algoritmo.

La propuesta del *pipeline* en mejora de iluminación enfrenta parte del problema presentado por el ambiente no controlado, según los resultados experimentales el medio día es el momento del día mas difícil para el reconocimiento debido a la posición del sol.

El uso de gafas dificulta el reconocimiento por acentuar la oclusión del rostro. Mientras que en días extremadamente soleados y causan que las persona cambien su comportamiento, como ocultar su rostros y provoca ciertas sombras, como las que se proyectan con la nariz y la cuenca de los ojos se pronuncien más.

Aun no es posible realizar el reconocimiento con una sola imagen por persona como conjunto de entrenamiento, por ello es necesario la elección de un conjunto de entrenamiento que cubra por los menos vistas frontales, de izquierda y de derecha para un reconocimiento adecuado.

Finalmente en lo que respecta al reconocimiento de imágenes ofuscadas las máscaras de Gabor pueden seguir recolectado información relevante aun si la imagen ha sido modificado por un *blurring*, esto se debe a que las máscaras recolectan información regional en el espacio de las señales para poder hacer un reconocimiento.

6.2. Trabajos futuros

A raíz de este trabajo de tesis se ha desarrollado varias propuestas para trabajos futuros siendo los mas relevantes los siguientes:

- Resolver el problema de hallar una configuración optima para los pesos de la función de similitud de **EBGM**.
- Probar técnicas de súper resolución para mejorar el reconocimiento y poder detectar rostros pequeños en baja resolución.
- Establecer un método de *tracking* que sea robusto en los ambientes no controlados.
- Realizar un análisis del coste computacional del proceso de *pipeline* e identificar que procesos se pueden optimizar
- Es necesario seguir la investigación de McPherson [McPherson et al. \(2016\)](#) para poder replicar el experimento y probar con más bases de datos, situaciones de video, y uniformizar las condiciones de las pruebas.

6.3. Publicaciones

A lo largo de este trabajo de tesis se presentó los siguientes artículos:

- An Improved Face Recognition Based on Illumination Normalization Technique and Elastic Bunch Graph Matching en ICFIP 2017 International Conference on Frontiers of Image Processing (Artículo aceptado)
- Modifications on Illumination, Distance function and Gabor masks for Elastic Bunch Graph Matching en la 35th International Conference of the Chilean Computer Science Society (SCCC 2016) organizado con la 42th Latin American Computing Conference (CLEI 2016) [Caceres et al. \(2016\)](#).

Bibliografía

- Alice, I. (2003). Biometric recognition: Security and privacy concerns. *IEEE Security & Privacy*.
- Baltrušaitis, T., Robinson, P., and Morency, L.-P. (2013). Constrained local neural fields for robust facial landmark detection in the wild. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 354–361.
- Baltrušaitis, T., Robinson, P., and Morency, L.-P. (2016). Openface: an open source facial behavior analysis toolkit. In *IEEE Winter Conference on Applications of Computer Vision*.
- Bolme, D. S. (2003). *Elastic bunch graph matching*. PhD thesis, Colorado State University.
- Bruce, V., Hancock, P. J., and Burton, A. M. (1998). Human face perception and identification. In *Face Recognition*, pages 51–72. Springer.
- Burton, A. M., Wilson, S., Cowan, M., and Bruce, V. (1999). Face recognition in poor-quality video: Evidence from security surveillance. *Psychological Science*, 10(3):243–248.
- Caceres, J. C. G., Chavez, J. R. G., and Riveros, E. L. (2016). Modifications on illumination, distance function and gabor masks for elastic bunch graph matching. *CLEI Electronic Journal*, 19(2).
- Cambridge, A. L. (2002). Att the database of faces (formerly the orl database of faces).
- Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.
- Cottrell, G. W. and Fleming, M. (1990). Face recognition using unsupervised feature extraction. In *Proc. of the Int. Neural Network Conf*, pages 322–325.
- Dalal, N. and Triggs, B. (2005). Histograms of oriented gradients for human detection. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, volume 1, pages 886–893. IEEE.
- Er, M. J., Wu, S., Lu, J., and Toh, H. L. (2002). Face recognition with radial basis function (rbf) neural networks. *IEEE transactions on neural networks*, 13(3):697–710.

- Georghiades, A., Belhumeur, P., and Kriegman, D. (1997). Yale face database. *Center for computational Vision and Control at Yale University*, <http://cvc.yale.edu/projects/yalefaces/yalefa>, 2.
- Givens, G., Beveridge, J. R., Draper, B. A., Grother, P., and Phillips, P. J. (2004). How features of the human face affect recognition: a statistical comparison of three face recognition algorithms. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–381. IEEE.
- Gorodnichy, D., Granger, E., Radtke, P., Authority, C. S., and Meunier, P. (2014). Survey of commercial technologies for face recognition in video. *CBSA, Border Technology Division, Tech. Rep*, 22.
- Gross, R. and Brajovic, V. (2003). An image preprocessing algorithm for illumination invariant face recognition. In *International Conference on Audio-and Video-Based Biometric Person Authentication*, pages 10–18. Springer.
- Gross, R., Shi, J., and Cohn, J. F. (2001). *Quo vadis face recognition?* Carnegie Mellon University, The Robotics Institute.
- Harvey, A. (2014). Cv dazzle (2010–2012) <http://cvdazzle.com/> and <http://ahprojects.com/projects/cv-dazzle>.
- Hill, H. and Bruce, V. (1996). The effects of lighting on the perception of facial surfaces. *Journal of Experimental Psychology: Human Perception and Performance*, 22(4):986.
- Hill, H., Schyns, P. G., and Akamatsu, S. (1997). Information and viewpoint dependence in face recognition. *Cognition*, 62(2terzopoulos1988constraints):201–222.
- Huang, J., Heisele, B., and Blanz, V. (2003). Component-based face recognition with 3d morphable models. In *International conference on audio-and video-based biometric person authentication*, pages 27–34. Springer.
- Johnston, A., Hill, H., and Carman, N. (1992). Recognising faces: effects of lighting direction, inversion, and brightness reversal. *Perception*, 21(3):365–375.
- Kalocsai, P., Zhao, W., and Elagin, E. (1998). Face similarity space as perceived by humans and artificial systems. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on*, pages 177–180. IEEE.
- Lin, S.-H., Kung, S.-Y., and Lin, L.-J. (1997). Face recognition/detection by probabilistic decision-based neural network. *IEEE transactions on neural networks*, 8(1):114–132.
- Liu, C. (2006). Capitalize on dimensionality increasing techniques for improving face recognition grand challenge performance. *IEEE transactions on pattern analysis and machine intelligence*, 28(5):725–737.

BIBLIOGRAFÍA

- Manjula, V. (2014). Image normalization robust using histogram equalization and logarithm transform frequency dct coefficients for illumination in facial images. *International Journal of Science and Research (IJSR)*.
- McPherson, R., Shokri, R., and Shmatikov, V. (2016). Defeating image obfuscation with deep learning. *arXiv preprint arXiv:1609.00408*.
- Nazare, A. C., Dos Santos, C. E., Ferreira, R., and Schwartz, W. R. (2014). Smart surveillance framework: A versatile tool for video analysis. In *IEEE Winter Conference on Applications of Computer Vision*, pages 753–760. IEEE.
- Odobez, J.-M., Carincotte, C., Emonet, R., Jouneau, E., Zaidenberg, S., Ravera, B., Bremond, F., and Grifoni, A. (2012). Unsupervised activity analysis and monitoring algorithms for effective surveillance systems. In *European Conference on Computer Vision*, pages 675–678. Springer.
- Orlova, E. V. (2002). The image processing handbook, john c. russ: Crc press and ieee press (1998), 771 pages. isbn 0849325323, \$ 139.95.
- Pandya, J. M., Rathod, D., and Jadav, J. J. (2013). A survey of face recognition approach. *International Journal of Engineering Research and Applications (IJERA)*, 3(1):632–635.
- Parmar, D. N. and Mehta, B. B. (2014). Face recognition methods & applications. *arXiv preprint arXiv:1403.0485*.
- Schwartz, A. (2012). Chicago’s video surveillance cameras: A pervasive and poorly regulated threat to our privacy. *Nw. J. Tech. & Intell. Prop.*, 11:ix.
- Sirovich, L. and Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. *JOSA A*, 4(3):519–524.
- Struc, V. (2012). The phd toolbox: Pretty helpful development functions for face recognition.
- Szeliski, R. (2010). *Computer vision: algorithms and applications*. Springer Science & Business Media.
- Tech, G. (1999). Georgia tech face database.
- Thamizharasi, A. and Jayasudha, J. (2015). A literature survey on various illumination normalization techniques for face recognition with fuzzy k nearest neighbour classifier. *ICTACT Journal on Image & Video Processing*, 5(4).
- Tian, Y.-L., Lu, M., and Hampapur, A. (2005). Robust and efficient foreground analysis for real-time video surveillance. In *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, volume 1, pages 1182–1187. IEEE.
- Tseng, S. (2003). *Comparison of holistic and feature based approaches to face recognition*. PhD thesis, School of Computer Science and Information Technology, Faculty of Applied Science, Royal Melbourne Institute of Technology.

- Turk, M. and Pentland, A. (1991). Eigenfaces for recognition. *Journal of cognitive neuroscience*, 3(1):71–86.
- Viola, P. and Jones, M. (2001). Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511. IEEE.
- Vishwakarma, V. P. (2015). Illumination normalization using fuzzy filter in dct domain for face recognition. *International Journal of Machine Learning and Cybernetics*, 6(1):17–34.
- Wiskott, L., Fellous, J.-M., Kuiger, N., and Von Der Malsburg, C. (1997). Face recognition by elastic bunch graph matching. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):775–779.
- Wood, D. M., Ball, K., Lyon, D., Norris, C., and Raab, C. (2006). A report on the surveillance society. *Surveillance Studies Network, UK*.
- Yang, M.-H. (2002). Kernel eigenfaces vs. kernel fisherfaces: Face recognition using kernel methods. In *fgr*, volume 2, page 215.
- Zhao, L. and Yang, Y.-H. (1999). Theoretical analysis of illumination in pca-based vision systems. *Pattern recognition*, 32(4):547–564.
- Zhao, W., Chellappa, R., and Phillips, P. J. (1999). *Subspace linear discriminant analysis for face recognition*. Citeseer.
- Zhao, W., Chellappa, R., Phillips, P. J., and Rosenfeld, A. (2003). Face recognition: A literature survey. *Acm Computing Surveys (CSUR)*, 35(4):399–458.