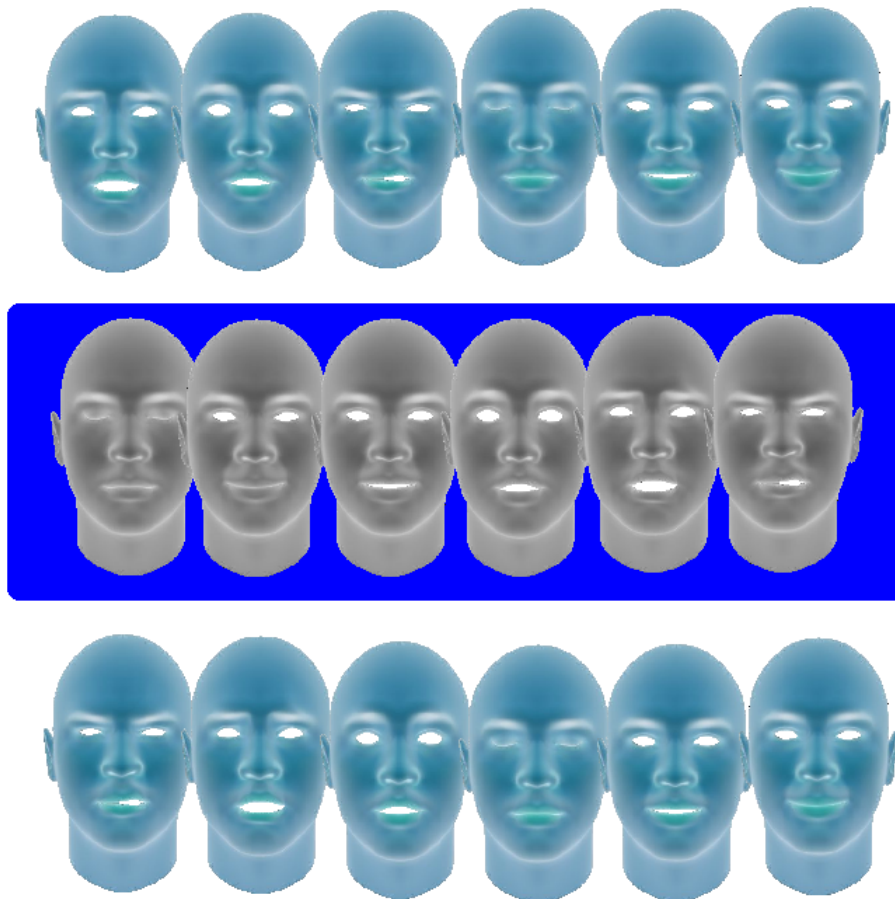


Seguimiento Dinámico de Expresiones Faciales Usando Tratamiento Digital de Imágenes



Alejandro Parada Mayorga



SEGUIMIENTO DINÁMICO DE EXPRESIONES FACIALES USANDO TRATAMIENTO DIGITAL DE IMÁGENES

Alejandro Parada Mayorga

Trabajo de grado para optar por el título de **Magister en Ingeniería Electrónica**

Director
Ph.D Arturo Plata Gómez

Co-director
Msc. Jaime Barrero

UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y TELECOMUNICACIONES
MAESTRÍA EN INGENIERÍA ELECTRÓNICA
Bucaramanga 2012

SUSTENTACIÓN TRABAJO DE INVESTIGACIÓN

LOS SUSCRITOS DIRECTOR Y CALIFICADORES DEL TRABAJO DE INVESTIGACIÓN DE MAESTRÍA TITULADO

"SEGUIMIENTO DINÁMICO DE EXPRESIONES FACIALES USANDO TRATAMIENTO DIGITAL DE IMÁGENES."

Realizado por el estudiante: **ALEJANDRO PARADA MAYORGA**, código 2097651, como parte de los requisitos de grado para la **Maestría en Ingeniería Electrónica** de la Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones, Facultad de Ingenierías Físico-Mecánicas,

ACUERDAN

Dar la calificación de **APROBADO**, después de realizada, la evaluación y sustentación oral del Trabajo de Grado (según Artículo 104 del Reglamento de Posgrado).

Para constancia firman en Bucaramanga a los 1 días del mes de Diciembre del año 2011.

ARTURO PLATA GÓMEZ.

Director del trabajo de investigación

JAIME BARRERO PÉREZ.

Codirector del trabajo de investigación

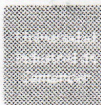
DANIEL SIERRA BUENO.

Evaluador

OSCAR GUALDRÓN GONZÁLEZ.

Evaluador

ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE TELECOMUNICACIONES (E³T)
Ciudad Universitaria, Carrera 27 – Calle 9, Edificio Ingeniería Eléctrica, IE-101
PBX: (7) 6344000 Ext. 2360 FAX: 6359622 A.A. 678 Bucaramanga, Colombia
Correo-e: e3t@uis.edu.co URL: http://www.e3t.uis.edu.co/



Recu 12/06/2011 5:14 PM

ACUERDO No. 164 DE 2003
(diciembre 16)



**ENTREGA DE TRABAJOS DE GRADO, TRABAJOS
DE INVESTIGACION O TESIS Y AUTORIZACIÓN
DE SU USO A FAVOR DE LA UIS**

Yo, Alejandro Parada Mayorga, mayor de edad, vecino de
Bucaramanga, identificado con la Cédula de Ciudadanía No. 1098602120
de B/manga, actuando en nombre propio, en mi calidad de autor del trabajo
de grado, del trabajo de investigación, o de la tesis denominada(o):

Seguimiento Dinámico de Expresiones Faciales usando Tratamiento
digital de Imágenes

hago entrega del ejemplar respectivo y de sus anexos de ser el caso, en formato digital o electrónico (CD o DVD) y autorizo a LA UNIVERSIDAD INDUSTRIAL DE SANTANDER, para que en los términos establecidos en la Ley 23 de 1982, Ley 44 de 1993, decisión Andina 351 de 1993, Decreto 460 de 1995 y demás normas generales sobre la materia, utilice y use en todas sus formas, los derechos patrimoniales de reproducción, comunicación pública, transformación y distribución (alquiler, préstamo público e importación) que me corresponden como creador de la obra objeto del presente documento. PARÁGRAFO: La presente autorización se hace extensiva no sólo a las facultades y derechos de uso sobre la obra en formato o soporte material, sino también para formato virtual, electrónico, digital, óptico, uso en red, Internet, extranet, intranet, etc., y en general para cualquier formato conocido o por conocer.

EL AUTOR – ESTUDIANTE, manifiesta que la obra objeto de la presente autorización es original y la realizó sin violar o usurpar derechos de autor de terceros, por lo tanto la obra es de su exclusiva autoría y detenta la titularidad sobre la misma. PARÁGRAFO: En caso de presentarse cualquier reclamación o acción por parte de un tercero en cuanto a los derechos de autor sobre la obra en cuestión, EL AUTOR / ESTUDIANTE, asumirá toda la responsabilidad, y saldrá en defensa de los derechos aquí autorizados; para todos los efectos la Universidad actúa como un tercero de buena fe.

Para constancia se firma el presente documento en dos (02) ejemplares del mismo valor y tenor, en Bucaramanga, a los 13 días del mes de Febrero de Dos Mil doce. 200

EL AUTOR / ESTUDIANTE:

(Firma)

Alejandro Parada Mayorga

Nombre

Alejandro Parada Mayorga

*Este trabajo está dedicado a mi hermano Salvador, cuya existencia física se
extinguió llevándose una parte de mí y de mi familia ... pero cuya presencia sigue estando
vigente en nuestras vidas como un recordatorio de que los ángeles sí existen...*

AGRADECIMIENTOS

A ti señor Jesucristo por darme la pasión para enfrentar mis retos profesionales y llenarme de fe para superar día a día las eventualidades de la vida.

A mis padres por su apoyo económico. En especial agradezco a mi madre por su fe en la realización de mi proyecto de vida.

A mi hermana Pilar, ejemplo de vida, mi amiga, mi hermana, mi confidente... Gracias por creer en mí y estar a mi lado incondicionalmente.

Al doctor Arturo Plata Gómez, director de este proyecto de investigación, quién no solo brindó sus conocimientos sino también su amistad y apoyo a lo largo del proceso, dejando una huella grande en mi formación profesional.

Al doctor Oscar Gualdrón González, por atender algunas de mis inquietudes y dudas a pesar de su apretada agenda.

Al doctor Mikkel B. Stegmann en, The Technical University of Denmark, por atender algunas de mis dudas sobre *Active Appearance Models*.

Al doctor John Aldo Lee en, Université catholique de Louvain, por las valiosas discusiones acerca de *Nonlinear dimensionality Reduction* y la atención de mis inquietudes sobre algunos de sus desarrollos.

Al doctor De-Chuan Zhan en, Nanjing University, por atender mis inquietudes acerca de sus desarrollos sobre s -**Isomap**.

A mis docentes de maestría

- Doctor Henry Lamos Díaz, por su motivación y enseñanzas hacia una formación matemática con profundidad, y atender en múltiples ocasiones mis dudas sobre diversos temas.
- Doctor Oscar Gualdrón González, por compartir con sus estudiantes no solo conocimientos sino también su amplia visión sobre la investigación.
- Doctor Daniel Sierra Bueno, por su impecable labor como docente y su interés en la formación de sus estudiantes.
- Doctor Ruben Darío Cruz, por sus fuertes exigencias sobre la elaboración de la propuesta de maestría. Todas sus enseñanzas y recomendaciones influyeron significativamente en el cumplimiento de este proyecto.

A la Universidad Industrial de Santander, por permitirme cursar mis estudios de posgrado con la beca de maestría.

Al doctor Gabriel Ordóñez, coordinador de la asignatura Tratamiento de Señales, por creer en mí y darme la maravillosa oportunidad de dictar esta asignatura.

Quiero agradecer al profesor Rafael Isaacs Giraldo, quién brindó su apoyo a mis constantes intentos por acercarme a las matemáticas en mis estudios de pregrado.... este trabajo tiene su huella, pues aún creo firmemente en las verdaderas matemáticas como el futuro de los principales desarrollos científicos..... y aun continúo buscando las matemáticas más abstractas y el por qué de muchos resultados matemáticos como una puerta para disfrutar el mundo y contribuir a la humanidad.

RESUMEN

TÍTULO: SEGUIMIENTO DINÁMICO DE EXPRESIONES FACIALES USANDO TRATAMIENTO DIGITAL DE IMÁGENES

¹

AUTOR:² ALEJANDRO PARADA MAYORGA

PALABRAS CLAVE: Expresiones Faciales, Tratamiento Digital de Imágenes, Detección de Rostro, Localización de Puntos Faciales, Extracción de Características, *features*, Reducción de dimensionalidad, Comportamiento Dinámico, Modelos de Apariencia Activa.

DESCRIPCIÓN:

En este trabajo se aborda el problema de seguimiento dinámico de las expresiones faciales usando tratamiento digital de imágenes. Se trata inicialmente el problema de la detección de rostro empleando las técnicas más efectivas para dicho fin de acuerdo al estado del arte, integrando herramientas cromáticas y de textura.

Con esto como punto de partida, se realiza la detección de puntos faciales empleando los *Active Appearance Models*, dando una descripción detallada de su construcción, entrenamiento y validación con imágenes faciales pertenecientes a la base de datos Cohn-Kanade.

La extracción de características es abordada siguiendo las propuestas más consolidadas de la literatura, dando como resultado una colección de cinco conjuntos de *features* encajados en espacios euclidianos de dimensiones fijas, cada uno de los cuales se procesa por separado.

Como etapa final se realiza un análisis a fin de determinar las técnicas más apropiadas para realizar la estimación de la dimensión y la reducción de dimensionalidad de la información extraída. Se plantea un esquema de reducción considerando las técnicas más apropiadas para los fines de esta investigación considerando calidad de reducción, fundamentación teórica y selección de parámetros.

Finalmente se presentan los resultados generales de la investigación, consignando las principales conclusiones y delimitando un escenario de aplicación para los desarrollos logrados, dando además recomendaciones para futuras investigaciones.

¹Proyecto de Grado de Maestría

²Facultad de Ingenierías Físico-Mecánicas. Escuela de Ingeniería Eléctrica, Electrónica y Telecomunicaciones. Director: Arturo Plata Gómez. email: alejandro_parada.m@hotmail.com

SUMMARIZE

TITLE: DYNAMIC TRACKING OF FACIAL EXPRESSIONS USING DIGITAL IMAGE PROCESSING

³

AUTHOR:⁴ ALEJANDRO PARADA MAYORGA

KEY WORDS: Facial Expressions, Digital Image Processing, Face Detection, Facial Point Detection, Feature Extraction, Nonlinear dimensionality reduction, dynamic behavior, Active Appearance Models.

DESCRIPTION:

This paper addresses the problem of dynamic tracking of facial expressions using digital image processing. Initially addressing the problem of face detection using the most effective techniques for this purpose according to the state of the art, integrating color and texture tools.

With this as a starting point, we make the facial point detection using the *Active Appearance Models*, giving a detailed description of its construction, training and validation using facial images belonging to the database Cohn-Kanade.

The feature extraction is treated according to the most important proposals of the literature, resulting in a collection of five sets of textit features embedded in Euclidean spaces of fixed dimensions, each of which is processed separately.

As this, final analysis is performed to determine the most appropriate techniques for estimating the size and dimensionality reduction of the extracted information. We propose a reduction scheme considering the most appropriate techniques for the purposes of this research considering the quality of reduction, theoretical foundation and parameter selection.

Finally, we present the overall results of the research, detailing the main findings and outlining an application scenario for the developments, besides giving recommendations for future research.

³Master Thesis

⁴Physics Mechanical Engineering Faculty. Electric, Electronic and Telecommunications School. Director:Arturo Plata Gómez. email: alejandro-parada.m@hotmail.com

Índice general

1. Introducción	19
1.1. Motivación	20
1.2. Consideraciones Generales	20
1.3. Organización del Documento	20
2. Detección de Rostro	23
2.1. Estado del Arte	23
2.2. Esquema Seleccionado	25
2.2.1. <i>Comprehensive Colour Image Normalization (CCIN)</i>	25
2.2.2. <i>Centroid Color Segmentation (CCS)</i>	25
2.2.3. <i>Adaboost</i>	27
2.3. Implementación	30
2.3.1. <i>Comprehensive Color Image Normalization (CCIN)</i>	30
2.3.2. <i>Color Centroids Segmentation (CCS)</i>	31
2.3.3. <i>Adaboost</i>	31
2.4. Resultados	31
2.5. Conclusión	31
3. Localización Automática de Puntos Faciales	37
3.1. Estado del Arte	37
3.2. Modelos de Apariencia Activa (AAM)	38
3.2.1. Formulación del Modelo de Forma	38
3.2.2. Formulación del Modelo de Textura	40
3.2.3. Formulación del Modelo Combinado	43
3.2.4. Búsqueda en <i>AAM</i>	44
3.3. Implementación	47
3.4. Resultados	47
3.4.1. Evaluación Cuantitativa	47

3.5. Discusión y Conclusiones	48
4. Extracción de Descriptores	65
4.1. Estado del Arte	65
4.2. Definición de Parámetros	66
4.2.1. Alternativa Planteada	66
4.2.2. Comportamiento Dinámico: Variedades de Datos	68
4.3. Implementación	71
4.4. Discusión y Conclusiones	71
5. Reducción de Dimensionalidad	73
5.1. Hipótesis Central	74
5.2. Motivaciones Prácticas	74
5.3. Motivaciones Teóricas	74
5.4. Aspectos a Explorar	76
5.5. Estado del Arte	76
5.6. Dimensión Intrínseca	76
5.6.1. Dimensión de Recubrimiento	77
5.6.2. Dimensión de Hausdorff	78
5.6.3. Dimensión de Capacidad (<i>Box-Counting Dimension</i>)	78
5.6.4. Dimensión de Información	78
5.6.5. Dimensión de Correlación	79
5.6.6. Relación entre algunas dimensiones	79
5.6.7. Estimación Práctica de la Dimensión Intrínseca	79
5.6.8. Algunos Resultados Sobre Variedades Diferenciales: Acerca de la Dimensión, <i>PCA local Modificado (PCA-ℓm)</i>	82
5.6.9. Algunos Resultados Sobre Variedades: Sobre el Teorema de Whitney, <i>un Nuevo Método para la Estimación de la Dimensión (PCA-ω)</i>	84
5.7. Técnicas Basadas en Preservación de la Distancia	88
5.7.1. Distancia Euclidiana	88
5.7.2. Técnicas Basadas en Distancias de Grafo	94
5.7.3. Otras Técnicas	108
5.8. Técnicas Basadas en Preservación de la Topología	108
5.8.1. Retícula definida por Datos	108
5.9. Comparación de los Métodos	117
5.9.1. Estimación de la Dimensión	117
5.9.2. Reducción de Dimensionalidad	118

5.10. Esquema propuesto para la Reducción de Dimensionalidad	118
5.11. Implementación	119
5.11.1. Estimación Dimensión Intrínseca	119
5.11.2. Reducción de Dimensionalidad	119
5.12. Discusión y Conclusiones	120
6. Resultados Generales	121
6.1. Escenario de Aplicación	121
6.2. Productos Entregados	122
6.2.1. Artículos de Divulgación	122
6.2.2. Códigos	124
6.2.3. Detección de Rostro	124
6.2.4. Localización de Puntos Faciales y Extracción de Características	124
6.2.5. Reducción de Dimensionalidad	124
6.2.6. Códigos Auxiliares	125
7. Conclusiones, Discusión y Trabajos Futuros	127
7.1. Conclusiones	127
7.1.1. Detección de Rostro	127
7.1.2. Localización de Puntos Faciales	128
7.1.3. Extracción de Características	128
7.1.4. Reducción de Dimensionalidad	128
7.2. Discusión	129
7.3. Recomendaciones sobre Trabajos Futuros	129
7.3.1. Detección de Rostro	129
7.3.2. Localización de Puntos Faciales	130
7.3.3. Extracción de Características y Reducción de Dimensionalidad	130
7.4. Comparación con el Trabajo de Otros Autores	130
Bibliografía	131

Índice de cuadros

3.1. Resultados AAM	49
5.1. Estimación de la Dimensión de Correlación de $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7].	81
5.2. Estimación de la Dimensión de $\widehat{\mathcal{M}}_{FSF}$ usando $\mathbf{PCA}-\ell m \left(\widehat{\mathcal{M}}_{FSF} \right)$ y $d_{corr} \left(\widehat{\mathcal{M}}_{FSF} \right)$, para el sujeto S055 de la base de datos Cohn-Kanade [7]	85
5.3. Estimación de la Dimensión de $\widehat{\mathcal{M}}_{FSF}$ usando $\mathbf{PCA}-\ell m \left(\widehat{\mathcal{M}}_{FSF} \right)$, $d_{corr} \left(\widehat{\mathcal{M}}_{FSF} \right)$ y $\mathbf{PCA}-\omega \left(\widehat{\mathcal{M}}_{FSF} \right)$ para el sujeto S055 de la base de datos Cohn-Kanade [7]	87
5.4. Estimación de la dimensión usando MDS sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7]. La dimensión estimada se basa en conservar el 98 % de la energía de los eigenvalores	90

Índice de figuras

2.1. Esquema Propuesto para la Detección de Rostro	24
2.2. Izquierda: Imagen Original $\mathcal{I}_{i,j}$. Derecha: Imagen Normalizada $\mathcal{I}_{i,j}^{final}$	26
2.3. Transformación del espacio RGB a \mathbb{R}^2	26
2.4. Distribución de la Piel en el plano (θ, r) . Hay 517880 pixeles de Piel tomadas de las bases de datos: FERET [29] [30], Cohn-Kanade [7] y PIE [31] junto con una miscelánea de 100 imágenes extraídas de Internet	28
2.5. Izquierda: $\mathcal{I}_{i,j}^{normalized}$. Derecha: Segmentación sobre $\mathcal{I}_{i,j}^{normalized}$ usando CCS, con los umbrales especificados en 2.8, 2.9, 2.10	29
2.6. Detección de Rostro usando <i>Adaboost</i> sobre una imagen de la base de datos FERET [29] [30]. Algoritmo de Viola y Jones [10]	31
2.7. Esquema general de Resultados Finales. Imagen de la base de datos FERET [29] [30]	32
2.8. Resultados Finales. Imágenes Base de Datos Cohn-Kanade [7]	33
2.9. Resultados Finales. Imágenes Base de Datos Cohn-Kanade [7]	34
2.10. Resultados Finales. Imágenes Base de Datos Cohn-Kanade [7]	35
2.11. Resultados Finales. Imágenes Base de Datos Cohn-Kanade [7]	35
3.1. Izquierda: Conjunto de marcas asignadas con el estándar de la base de datos Cohn-Kanade [7]. Derecha: Conjunto de marcas asignadas siguiendo el esquema de Cootes [1]	40
3.2. Diagrama que ilustra el <i>Image Warping</i> , en donde \mathbf{x}_r es una versión deformada de $\tilde{\mathbf{x}}_r$	42
3.3. De izquierda a derecha: a) Imagen Original, b) Imagen con Marcas asignadas, c) Enmallado construido usando las marcas, d) Textura en el interior del <i>convex hull</i> del enmallado mapeada sobre la forma media usando <i>image warping</i>	44
3.4. De izquierda a Derecha: a) Valor del modelo haciendo $c_1 = -\sqrt{\lambda_1}$, $c_i = 0 \quad \forall i > 1$, b) Textura media, sobre la forma media (<i>Mean Shape</i>) c) Valor del modelo haciendo $c_1 = \sqrt{\lambda_1}$, $c_i = 0 \quad \forall i > 1$, d) Representación de los cambios de forma por la variación de c_1 (Azul: <i>Mean Shape</i> , Verde: $c_1 = \sqrt{\lambda_1}$, Rojo: $c_1 = -\sqrt{\lambda_1}$)	45
3.5. Resultados Imágenes 1-7. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	50

3.6. Resultados Imágenes 8-14. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	51
3.7. Resultados Imágenes 15-21. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	52
3.8. Resultados Imágenes 22-28. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	53
3.9. Resultados Imágenes 29-35. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	54
3.10. Resultados Imágenes 36-42. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	55
3.11. Resultados Imágenes 43-49. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	56
3.12. Resultados Imágenes 50-56. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	57

3.13. Resultados Imágenes 57-63. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	58
3.14. Resultados Imágenes 64-70. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	59
3.15. Resultados Imágenes 71-77. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	60
3.16. Resultados Imágenes 78-84. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	61
3.17. Resultados Imágenes 85-91. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	62
3.18. Resultados Imágenes 92-98. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	63
3.19. Resultados Imágenes 99-100. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el <i>mean shape</i> en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1	64
4.1. Features \mathbf{m}_{FSF}	66

4.2. Features \mathbf{m}_{FST}	67
4.3. Features \mathbf{m}_{FIF}	68
4.4. Features \mathbf{m}_{FIS}	69
4.5. Features \mathbf{m}_{FIT}	70
5.1. Gráfica típica de \mathbf{df} vs el tamaño de la ventana \mathbf{Wd} en la aplicación de PCA — ℓm . La primera región plana asociada a la ordenada de valor 2 indica el valor estimado de la dimensión.	83
5.2. Dos comportamientos típicos en la curva AIC . Izquierda: Caso en el cual existe un n que minimiza la función $AIC(n)$. Derecha: Caso en el cual el valor mínimo de AIC se logra con el número total de eigenvalores, pero existe un marcado cambio de comportamiento con un número menor de los mismos	84
5.3. De izquierda a derecha: El rollo suizo, El Toro, La botella de Klein, La banda de Möbius.	87
5.4. Resultados de aplicar NLM sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], empleando la dimensión de correlación. Algoritmo de Vesanto [81]	92
5.5. Resultados de aplicar CCA sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], empleando la dimensión de correlación. Algoritmo de Vesanto [83]	95
5.6. Gráfica para encontrar el valor óptimo de K propuesta por Shao [86] para los datos $\widehat{\mathcal{M}}_{FSF}$ del sujeto S055 de la base de datos Cohn-Kanade [7] construyendo el grafo a partir de distancias euclidianas, el valor óptimo sería de 12	97
5.7. Resultados de aplicar Isomap sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], usando $K = 12$ seleccionado con el criterio asociado a $f(K)$. Algoritmo de Tenenbaum [84]	99
5.8. Resultados de aplicar s — Isomap sobre la variedad $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], usando $K = 11$ seleccionado con el criterio asociado a $f(K)$. Algoritmo Geng-Tenebaum [87]	101
5.9. Resultados de aplicar GNLM y s — GNLM sobre $\widehat{\mathcal{M}}_{FSF}$, empleando la dimensión de correlación. Con $K = 12, 11$ respectivamente, seleccionado con el criterio asociado a $f(K)$	104
5.10. Resultados de aplicar CDA y s — CDA sobre el conjunto $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], usando $K = 12, 11$ respectivamente seleccionado con el criterio asociado a $f(K)$	107
5.11. Resultados de aplicar LLE sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-kanade [7], empleando la dimensión de correlación. Algoritmo de Roweis [91]	111
5.12. Resultados de aplicar LE sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kande [7], empleando la dimensión de correlación. Algoritmo de Belkin [62]	113
5.13. Resultados de aplicar HLLE sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], empleando la dimensión de correlación. Algoritmo de Grimes [59]	116
5.14. Esquema General de Reducción de Dimensionalidad	119
6.1. Esquema General de Seguimiento Dinámico	123

1

Introducción



La extracción y procesamiento de información asociada a expresiones faciales es un problema que despierta interés en diversas áreas [1] [2]. Distintos desarrollos han sido realizados para la extracción, representación y análisis de dicha información, sin embargo la mayor parte de éstos se enfocan en un manejo estático asociado a un determinado instante de tiempo, quedando así por explotar aún mucha de la riqueza existente en el comportamiento dinámico [1].

En esta tesis se aborda el problema de realizar el seguimiento dinámico por medio de herramientas computacionales a las expresiones faciales, en donde la hipótesis central es

Existen conjuntos de expresiones faciales por individuo, para las cuales es posible afirmar que: el comportamiento dinámico está bien diferenciado de una expresión a otra

es decir las trayectorias en el espacio de evolución de la información son diferentes y disjuntas entre sí.

Para probar esta hipótesis, se aborda el problema de la extracción de información lidiando primero con la detección de rostro, la localización de puntos faciales y la extracción de los descriptores para finalmente tratar en forma detallada la reducción de dimensionalidad.

En este capítulo se da un panorama de las consideraciones generales involucradas en la investigación, dando primero una presentación de los hechos que motivan el desarrollo de la misma. Finalmente se dará una descripción de la organización del documento.

1.1. Motivación

La descripción y análisis de la información contenida en las expresiones faciales de un determinado individuo, ha sido de interés en áreas que van desde la psicología [3] hasta áreas más recientes como la de *comunicaciones futuras* en donde el principal interés recae en la interacción hombre-máquina [1].

El estudio de estos patrones faciales se ha desarrollado en primera instancia alrededor de un manejo estático [1], sin embargo contribuciones recientes indican la riqueza para descripción y representación que puede encontrarse en un análisis dinámico [2].

El trabajo de investigadores como Goleman [3] en donde se destaca la importancia de lo que hoy es conocido como *la inteligencia emocional*, generan un gran interés en pro de emitir conceptos con información que va más allá de un solo instante de tiempo, y así tener en cuenta el proceso de evolución de una expresión en sí misma. Ekman [4] [5] por su parte, ha realizado marcadas contribuciones gracias a las cuales ha sido posible entre otras cosas, establecer conclusiones generales y con carácter de universalidad sobre los patrones faciales de los individuos y sus respuestas a determinados estímulos. Estos resultados están basados en profundos análisis dentro de las ciencias de la psicología, sin embargo brindan un soporte riguroso para el análisis y extracción automático de información de los patrones faciales con un enfoque dinámico, pues tanto las estrategias de reconocimiento como de clasificación se centran en la identificación de las *AU (Action Units)* que hacen parte del sistema *FACS (Facial Action Coding System)* desarrollado por Ekman [6], teniendo cada una de las mimas un grado de evolución diferente dentro de cada expresión [1] [2].

En torno a estas ideas se realiza esta investigación, buscando el desarrollo de herramientas para *cuantificar la dinámica* de los patrones faciales en un determinado sujeto, teniendo como finalidad lograr un seguimiento dinámico de la expresión que sea confiable para fines de clasificación y/o síntesis.

1.2. Consideraciones Generales

El problema de la detección de rostro se aborda considerando condiciones controladas. Es decir, es posible manejar distintas condiciones de iluminación para cada una de las imágenes pero no tratando situaciones extremas que puedan ser consideradas oclusiones. Adicionalmente, se consideran imágenes de sujetos cuya información facial no es interferida por aspectos fisionómicos como la barba, anteojos y accesorios que puedan ser causantes de algún tipo de oclusión.

Si bien el interés de la investigación es en la descripción dinámica de la información, se analiza *en bloque* la totalidad de la información contenida en una secuencia de video, de manera que no se aborda el problema de procesamiento en tiempo real.

1.3. Organización del Documento

En el capítulo 2 se presenta el problema de la detección de rostro planteando un esquema de solución basado en segmentación cromática y de textura. Se da una descripción detallada de cada una, así como una presentación de los resultados obtenidos con las imágenes a Color de la base de Datos Cohn-Kanade [7].

En el capítulo 3 se aborda el problema de la detección de puntos faciales. Se presenta una solución basada en *Active Appearance Models (AAM)*, dando una descripción completa de estos modelos. Se dan los detalles de cómo se realiza la construcción y el entrenamiento de los mismos, así como los resultados obtenidos con una miscelánea de imágenes de la base de datos Cohn-Kanade [7].

La extracción de descriptores es abordada en el capítulo 4 planteando un esquema que integra las propuestas más relevantes de la literatura y modificaciones adicionales a fin de dar una descripción más completa. Se dan a conocer en detalle los descriptores empleados y la forma como se maneja esta información para su posterior procesamiento.

En el capítulo 5 se aborda el problema de la reducción de dimensionalidad, presentando las técnicas con mejores resultados para dicho fin. Se da una descripción de las mismas a fin de conocer en detalle sus fortalezas y debilidades dentro de un análisis automático de la información extraída de las expresiones faciales. Esto brinda un análisis riguroso en una etapa crítica en el seguimiento, ya que el uso inadecuado de una técnica puede conducir a resultados sin sentido. Más aún, los resultados obtenidos por dos técnicas distintas pueden diferir de manera considerable si las técnicas o la selección de sus parámetros no son las adecuadas. Es un capítulo en el que se da prioridad a la rigurosidad y no a la brevedad, siendo su papel crucial en los resultados y posibles aplicaciones de los mismos. Además se da una presentación de una contribución realizada para estimar la dimensión intrínseca de los datos que se asumen recaen sobre una variedad diferencial.

Los resultados generales del esquema de seguimiento dinámico de la expresión son finalmente presentados en el capítulo 6, en el que además se describen los productos entregados en las investigación y se presenta el escenario para su aplicación.

Finalmente en el capítulo 7, se consignan las conclusiones de la investigación. Se presentan además las recomendaciones para trabajos futuros.

2

Detección de Rostro



La localización o detección de rostro en una imagen facial juega un papel esencial en el análisis del comportamiento dinámico de las expresiones. Si bien existen algunas herramientas como los *Active Appearance Models (AAM)* [1], que permiten en principio la segmentación detallada del rostro [8], su buen funcionamiento depende fuertemente de qué tan cercana sea la condición inicial de dichos modelos al rostro ubicado en la imagen objetivo. La mejor forma de lograr dicha condición es localizando lo más aproximado posible el rostro.

En este capítulo se presenta el esquema empleado para realizar la detección de rostro, con el cual se busca mejorar las condiciones para la construcción y aplicación de los *Active Appearance Models (AAM)* en la localización de puntos faciales. Inicialmente se realiza un breve comentario del estado del arte sobre la detección de rostro, mencionando las técnicas que se consideraron más relevantes y/o pertinentes para esta investigación.

En la sección 2.2 se presenta en detalle el esquema seleccionado mencionando las principales propiedades y características de cada técnica, y finalmente se presentarán los resultados de la misma para dar algunas de las conclusiones parciales del empleo de este esquema.

2.1. Estado del Arte

En [1] [2] se evidencia que existen dos grandes enfoques para la detección de rostro. El primero de ellos se basa en el uso de información de textura y *features* obtenidos a partir de la misma para localizar la región facial, mientras que el segundo se basa en el uso de la información a color para identificar regiones con piel y después discriminar con textura u otro criterio.

En tanto la técnicas basadas en textura son atractivas por su robustez [1] [2], las técnicas cromáticas los son por su bajo costo computacional representando un atractivo para implementaciones en dispositivos de bajas características de hardware [9].

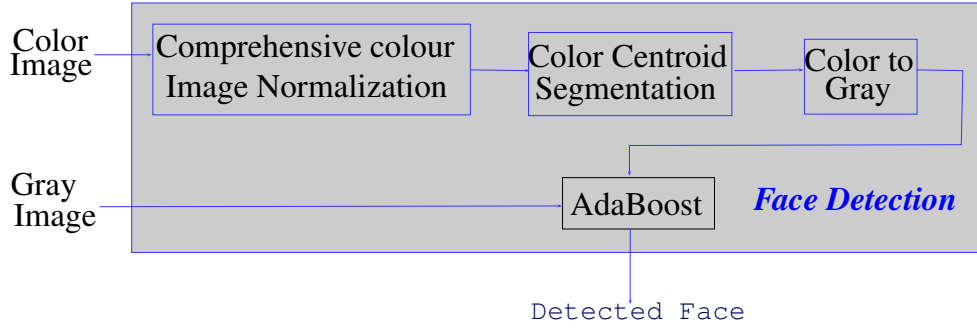


Figura 2.1: Esquema Propuesto para la Detección de Rostro

Teniendo en cuenta estas ideas, en este trabajo se propone realizar una integración de estos dos grandes enfoques, considerando los análisis realizados por Zli [1] [2] según los cuales, la detección de rostro basada en *Adaboost*, desarrollada por Viola y Jones [10], ha mostrado ser la mejor solución para la detección de rostro en condiciones controladas. Sumado a esto se consideran los resultados obtenidos por Ichikawa [11] y Huang [12] en los cuales se muestra un alto grado de robustez de ésta técnica para lidiar con problemas de detección facial incluso con oclusiones severas y grandes ángulos de rotación.

A fin de mejorar el grado de aproximación del detector de Viola y Jones [10], se emplea segmentación cromática siguiendo las ideas plasmadas en [13] [14] [15] en donde se emplea parte del detector de Viola y Jones adicionando información de tipo cromático para mejorar el proceso de detección.

Los trabajos realizados sobre segmentación cromática para detección de rostros son variados, como se evidencia en [1] [2]. La razón de esto recae en gran medida en el hecho de que las imágenes a color pueden representarse en distintos espacios, que pueden ser adecuados o no según la distribución de color de piel empleada para la construcción del *cluster* [1] [16]. Los espacios más empleados para estos fines son el YC_bC_r [17] [18] [19] y el espacio HSV [20] [21], combinados con procesos de discriminación adicionales basados en features extraídos de la imagen en escala de grises. En algunos trabajos como en [22] [23] [24] se emplea simultáneamente la información correspondiente a varios espacios de color, pero obteniendo pocas o ninguna ventaja sobre el empleo de un solo espacio.

La idea común a todos estos desarrollos sobre segmentación cromática de rostros, es la de encontrar una región adecuada en un espacio de color que contenga la distribución de colores correspondiente a las zonas de piel de una determinada imagen facial [1]. El uso del espacio YC_bC_r , aunque busca disminuir la dependencia de la iluminación al considerar para la segmentación los planos C_bC_r , no logra completamente dicho fin pues el plano Y involucra información cromática [25]. Al usar otros espacios este problema sigue existiendo pues el *cluster* empleado para realizar la segmentación dependerá fuertemente de la iluminación de las imágenes usadas para construir la distribución [25].

Como parte de una solución a este problema King [26] desarrolla una técnica que incluye las ideas básicas de segmentación cromática mencionadas anteriormente, pero adicionando una poderosa técnica de corrección de iluminación desarrollada por Finlayson [27]. Los resultados son extremadamente satisfactorios pues por medio de esta técnica de corrección de iluminación se logra *estabilizar* la distribución que representa el color de la piel sea cual sea el espacio que se use para representarla [26].

Por otra parte Zhang [28] desarrolla una técnica de segmentación por color capaz de lidiar por sí sola con algunos defectos de iluminación, y con la propiedad de *separar* satisfactoriamente la distribución de colores de la piel de las distribuciones de color de otros objetos no fácilmente separables por medio de otras técnicas.

En este trabajo se emplea la técnica de iluminación desarrollada por Finlayson [27] junto con la técnica propuesta por Zhang [28], para finalmente usar el detector de rostro de Viola y Jones [10] basado en *Adaboost*. Con esta propuesta se busca seguir las principales tendencias en el estado del arte, teniendo en cuenta la calidad de los desarrollos y su pertinencia para este trabajo. El esquema planteado se puede apreciar en la figura 2.1

2.2. Esquema Seleccionado

2.2.1. *Comprehensive Colour Image Normalization (CCIN)*

Una imagen bajo diferentes condiciones de iluminación puede aparecer con diferentes valores en su representación en el espacio RGB , ya sea debido a la geometría de la iluminación o a el *illuminant color* [25]. Por esto una técnica de normalización del color es necesaria para una segmentación cromática de mayor precisión. Finlayson [27] ha desarrollado una técnica de tipo iterativo para la normalización de imágenes a color. Éste autor prueba que usando sucesivamente las ecuaciones clásicas para compensar la geometría de la iluminación, es posible encontrar una imagen que es idempotente y única, y por tanto el proceso es siempre convergente. Además la rata de convergencia es alta, típicamente es alcanzada entre cinco y seis iteraciones.

Sea $\mathcal{I}_{i,j}$ $j = R, G, B$ una imagen en el espacio de color RGB , i es el pixel. Sea R el operador normalizador de renglones definido como

$$R(\mathcal{I}_{i,j}) = \frac{\mathcal{I}_{i,j}}{\sum_{k=R,G,B} \mathcal{I}_{i,k}} \quad (2.1)$$

y sea C el operador normalizador de columnas, definido como

$$C(\mathcal{I}_{i,j}) = \frac{N\mathcal{I}_{i,j}}{3 \sum_{k=1}^N \mathcal{I}_{k,j}} \quad (2.2)$$

N es el número total de pixeles en la imagen.

El método propuesto por Finlayson [27] puede presentarse como

1. Inicialización: Tome $\mathcal{I}_{i,j}^{(0)} = \mathcal{I}_{i,j}$ como el valor inicial en el proceso iterativo.
2. Paso de Iteración: Haga $\mathcal{I}_{i,j}^{(r+1)} = C \left(R \left(\mathcal{I}_{i,j}^{(r)} \right) \right)$.
3. Paso de Finalización: Si $\mathcal{I}_{i,j}^{(r+1)} = \mathcal{I}_{i,j}^{(r)}$ detenga el proceso y haga $\mathcal{I}_{i,j}^{normalized} = \mathcal{I}_{i,j}^{(r)}$, si no vaya al paso 2.

Una vez la convergencia es alcanzada es necesario hacer un escalamiento de los valores de la imagen resultante. En este trabajo se propone realizar este escalamiento como

$$\mathcal{I}_{i,j}^{final} = \mathcal{I}_{i,j}^{(0)} \mathcal{I}_{i,j}^{normalized} \quad (2.3)$$

Ebner ha mostrado en [25], que este no es el único camino para realizar este escalamiento pero es uno de los más usados y uno de los que permite obtener mejores resultados.

En la figura 2.2 se muestra un ejemplo de los resultados obtenidos para una imagen de la base de datos FERET [29] [30]. La principal diferencia no está en la apariencia de las imágenes sino en la relación que tiene cada pixel con sus vecinos en la imagen. Esta diferencia se remarcará en los espacios de color de representación.

2.2.2. *Centroid Color Segmentation (CCS)*

Zhang [28] ha propuesto una nueva técnica para la segmentación de color basada en una transformación del espacio RGB a un espacio de dos dimensiones usando un nuevo sistema de coordenadas

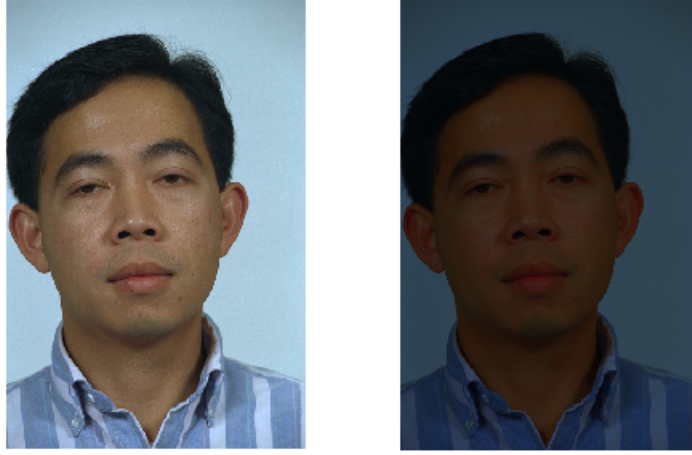


Figura 2.2: Izquierda: Imagen Original $\mathcal{I}_{i,j}$. Derecha: Imagen Normalizada $\mathcal{I}_{i,j}^{final}$

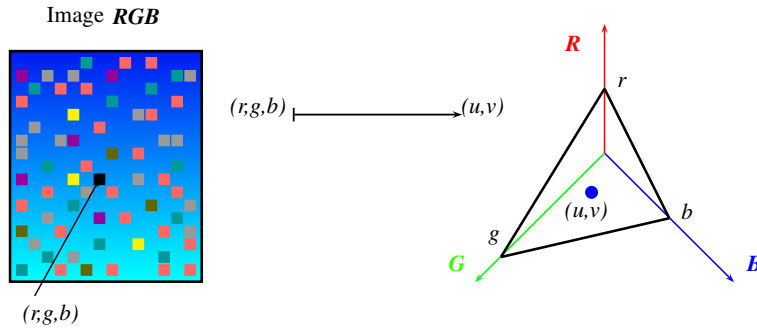


Figura 2.3: Transformación del espacio RGB a \mathbb{R}^2 .

y el centroide de un triángulo. En la figura 2.3 se muestra esta transformación, en donde cada componente del vector $(\mathcal{I}_{i,R}, \mathcal{I}_{i,G}, \mathcal{I}_{i,B}) = (r, g, b)$ es mapeada a un nuevo sistema de ejes, en el cual cada uno de los ejes está separado por 120 grados. El nuevo eje R está a 90 grados de la horizontal. Entonces, se construye un triángulo con los vértices asociados a los valores de los píxeles en este nuevo sistema coordenado, y éste tendrá un centroide (u, v) , donde (u, v) es el punto en el nuevo espacio de representación. Algebraicamente, esto se puede expresar de la siguiente manera:

Sea $\mathcal{T} : \mathbb{R}^3 \mapsto \mathbb{R}^2$ el operador de transformación del espacio RGB a \mathbb{R}^2 . La transformación está definida como

$$\mathcal{T}(\mathcal{I}_{i,j}) = u_i \hat{\mathbf{e}}_{\mathbf{x}} + v_i \hat{\mathbf{e}}_{\mathbf{y}} \quad (2.4)$$

donde

$$u_i = \frac{1}{3} \left[\mathcal{I}_{i,B} \cos\left(\frac{\pi}{6}\right) - \mathcal{I}_{i,G} \cos\left(\frac{\pi}{6}\right) \right] \quad (2.5)$$

$$v_i = \frac{1}{3} \left[\mathcal{I}_{i,R} - \mathcal{I}_{i,B} \sin\left(\frac{\pi}{6}\right) - \mathcal{I}_{i,G} \sin\left(\frac{\pi}{6}\right) \right] \quad (2.6)$$

$\hat{\mathbf{e}}_{\mathbf{x}}, \hat{\mathbf{e}}_{\mathbf{y}}$ son los vectores unitarios en las coordenadas x, y de \mathbb{R}^2 . Acorde a Zhang [28], después de que se hace esta transformación, se representan los datos en coordenadas polares (θ, r) , de manera que la región en la cual se espera que se encuentren los datos asociados al color de la piel es definida como

$$\{(\theta, r) \mid r \in [r_{min}, r_{max}], \quad \theta \in [\theta_{min}, \theta_{max}]\} \quad (2.7)$$

En este trabajo se representará la región de distribución de los píxeles de color de la piel, delimitando aún más la región de interés como el interior de un polígono. En la figura 2.4 se

muestra la distribución de pixeles de piel en el espacio (θ, r) con 517880 pixeles de piel tomadas de las bases de datos: FERET [29] [30], Cohn-Kanade [7] y PIE [31] junto con una miscelánea de 100 imágenes extraídas de Internet. La región de interés se delimita por medio de las ecuaciones:

$$r - 54,84 \leq -1,2758(\theta - 54,24) \quad (2.8)$$

$$r - 54,84 \leq 2,2074(\theta - 54,24) \quad (2.9)$$

$$r \geq 2,5 \quad (2.10)$$

Esta región se denomina $\mathfrak{M}_s \subset \mathbb{R}^2$, de tal manera que la región de distribución de la piel es

$$\{(\theta, r) | (\theta, r) \in \mathfrak{M}_s\} \quad (2.11)$$

Así, se espera obtener un modelo flexible pero que elimine muchas zonas en la imagen cuyo color no es cercano a la distribución de la piel. Se obtiene esta descripción matemática de manera empírica, de acuerdo a lo propuesto por Zhang [28], quién tomó algunos valores empíricos para delimitar θ y r . La máscara resultante de esta umbralización es

$$\mathcal{I}_{i,ccs} = \begin{cases} 1 & \text{Si } \mathcal{I}_{i,j} \in \mathfrak{M}_s \\ 0 & \text{En caso contrario} \end{cases} \quad (2.12)$$

Después de la umbralización, es necesario hacer un proceso de corrección *no lineal*. Zhang [28] realiza este proceso de la siguiente manera:

Sea $\mathcal{I}_{i,gray}$ la versión en escala de grises de $\mathcal{I}_{i,j}$, entonces la corrección no lineal es

$$\mathcal{I}_{i,binary} = \frac{\ln(1 + 255\mathcal{I}_{i,gray})}{k \ln(1 + 255)} \quad (2.13)$$

donde $k = 2$. Con este proceso el ruido es reducido. El valor de k , indica el número de *clusters* en la imagen resultante. Luego, la máscara de segmentación se obtiene como

$$\mathcal{I}_{i,masc} = \mathcal{I}_{i,ccs} \mathcal{I}_{i,binary} \quad (2.14)$$

Después de esto, se llenan los agujeros de $\mathcal{I}_{i,masc}$. Luego la máscara final de segmentación es

$$\mathcal{I}_{i,Fmasc} = \mathbf{F}_{fillHoles}(\mathcal{I}_{i,masc}) \quad (2.15)$$

donde $\mathbf{F}_{fillHoles}$ es el operador de llenado de huecos. En la figura 2.5 se muestran los resultados de usar segmentación CCS sobre la imagen normalizada, obtenida en el proceso de normalización de color.

2.2.3. Adaboost

Viola y Jones han propuesto y desarrollado en [10] un detector de rostro robusto basado en *Adaboost*, logrando con éste grandes tasas de reconocimiento. Acorde a Zli [1] [2] éste es de lejos el mejor detector de rostro para entornos controlados.

El funcionamiento de esta propuesta se basa principalmente en el uso de *features* extraídos con funciones tipo *Haar*, y el uso de una cascada de clasificadores. Usando *Adaboost*, los clasificadores débiles son diseñados de tal manera que se obtiene un clasificador fuerte como una superposición de los primeros, y la cascada de clasificación aumenta por etapas su poder de discriminación.

Para hacer este entrenamiento eficiente Viola y Jones introducen el concepto de *imagen integral*, por medio de la cual es posible obtener eficientemente todos los *features* de la imagen en una nueva computada de la original usando sumas acumulativas. Si se denota con \mathcal{I}_I la imagen integral calculada a partir de la imagen en escala de grises \mathcal{I} , se tiene que

$$\mathcal{I}_I(r, s) = \sum_{m \leq r, n \leq s} \mathcal{I}(m, n) \quad (2.16)$$

Los principales aspectos acerca del algoritmo de *Adaboost*, pueden resumirse de la siguiente manera [10] [1]

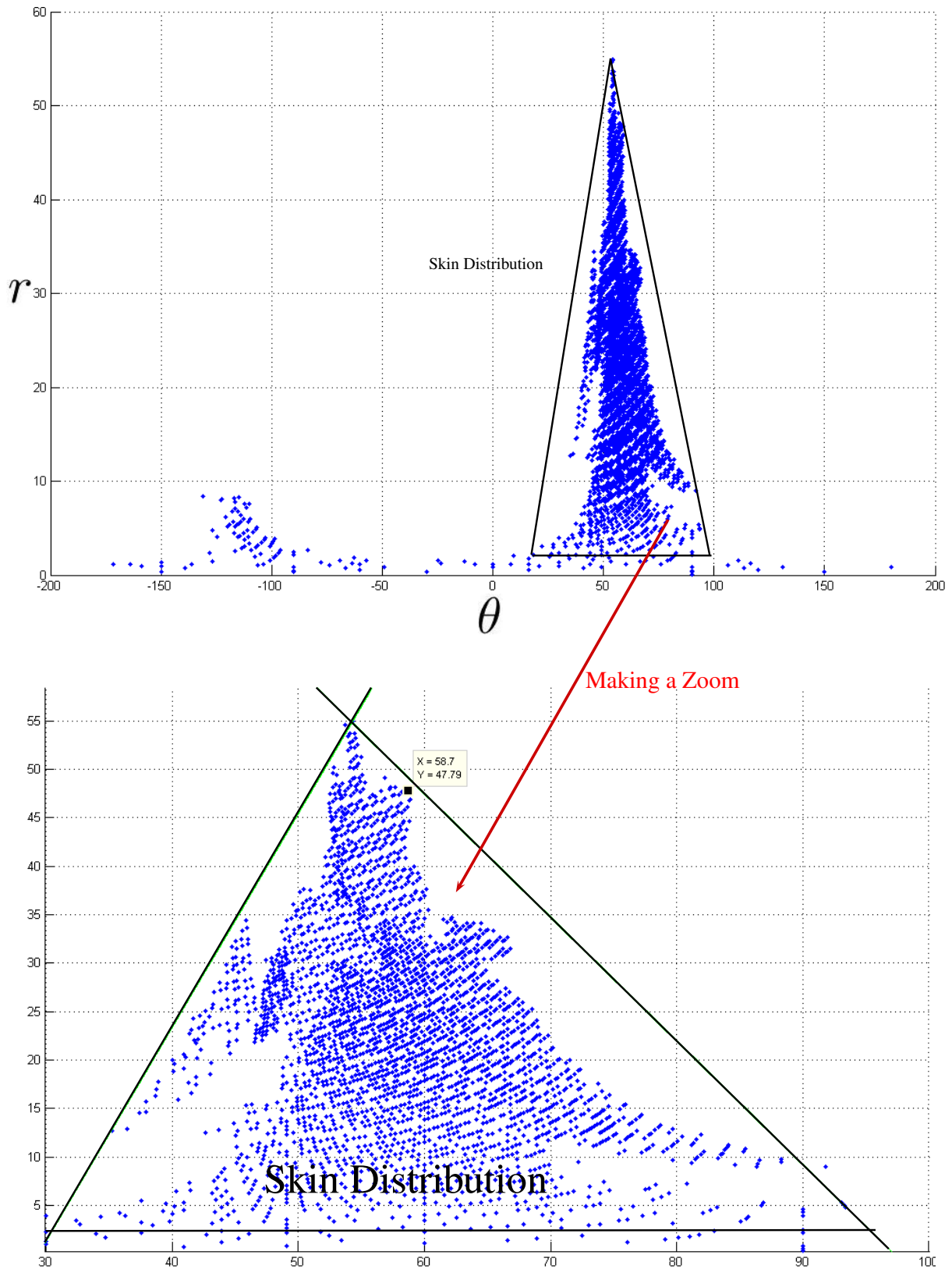


Figura 2.4: Distribución de la Piel en el plano (θ, r) . Hay 517880 pixeles de Piel tomadas de las bases de datos: FERET [29] [30], Cohn-Kanade [7] y PIE [31] junto con una miscelánea de 100 imágenes extraídas de Internet



Figura 2.5: Izquierda: $\mathcal{I}_{i,j}^{normalized}$. Derecha: Segmentación sobre $\mathcal{I}_{i,j}^{normalized}$ usando CCS, con los umbrales especificados en 2.8, 2.9, 2.10

0. (Entrada)

- (1) Sean los ejemplos de entrenamiento $\mathcal{Z} = \{(x_1, y_1), \dots, (x_N, y_N)\}$, donde $N = a + b$, tal que existen a ejemplos con $y_i = 1$ y b ejemplos con $y_i = -1$.
- (2) Sea M el número de clasificadores débiles a ser combinados.

1. (Inicialización)

$$w_i^{(0)} = \frac{1}{2a} \quad \text{Para los ejemplos con } y_i = 1$$

$$w_i^{(0)} = \frac{1}{2b} \quad \text{Para los ejemplos con } y_i = -1.$$

2. (Ciclo Iterativo)

For $m = 1, \dots, M$:

- (1) Seleccione el h_m óptimo que minimice el error residual:

$$\epsilon_m = \sum_i w_i^{(m-1)} 1[\text{sign}(h_m(x_i)) \neq y_i]$$

- (2) Seleccione α_m de acuerdo a

$$\alpha_m = \log \left(\frac{1 - \epsilon_m}{\epsilon_m} \right)$$

- (3) Actualizar $w_i^{(m)} \leftarrow w_i^{(m)} e^{-y_i \alpha_m h_m(x_i)}$ y normalice a $\sum_i w_i^{(m)} = 1$

3. (Salida)

Se construye la Función de Clasificación: H_M como

$$H_M(x) = \frac{\sum_{m=1}^M \alpha_m h_m(x)}{\sum_{m=1}^M \alpha_m}$$

Una vez que todos los clasificadores han sido diseñados. La cascada de clasificadores llamada *attentional cascade* es usada para llevar a cabo una búsqueda eficiente en la imagen. Cada clasificador usa un número creciente de *features*. De tal manera que los primeros juegan un papel importante para descartar regiones que no contengan información facial y las regiones con la información más rica son analizadas con los clasificadores más elaborados y aceptadas como regiones faciales. Para todos los detalles acerca de este procedimiento el autor refiere al lector al trabajo de Viola y Jones [10].

En la figura 2.6 se muestran los resultados de detección de rostro empleando Adaboost.

2.3. Implementación

2.3.1. Comprehensive Color Image Normalization (CCIN)

Comprehensive colour image normalization (CCIN) es implementada en Matlab. Debido al bajo costo computacional, del orden de mili segundos, el lenguaje de programación no juega un papel central.



Figura 2.6: Detección de Rostro usando *Adaboost* sobre una imagen de la base de datos FERET [29] [30]. Algoritmo de Viola y Jones [10]

2.3.2. *Color Centroids Segmentation (CCS)*

Como en el caso anterior, la implementación de CCS es realizada en Matlab. El programa se realiza haciendo uso de operaciones vectoriales, de tal manera que el costo computacional es muy bajo, del orden de mili segundos.

2.3.3. *Adaboost*

Una de las implemantaciones más populares del detector de rostro de Viola y Jones basada en Adaboost se encuentra disponible en la librería openCV para C++, y se ejecuta en el orden de los mili segundos. Masnadi [32] ha realizado una implementación de ésta técnica para Matlab, pero no es la mejor opción debido al enorme costo computacional, que puede llegar a ser de varios minutos para una imagen de baja resolución. Por lo tanto en este trabajo se usará la implementación que se encuentra en la librería openCV.

2.4. Resultados

En la figura 2.7 se muestra un diagrama que ejemplifica el esquema general de detección de rostro discutido, sobre una de las imágenes de la base de datos FERET [29] [30], mientras que en las figuras 2.8, 2.9, 2.10 y 2.11 se muestran los resultados sobre una imagen en escala de grises y todas las imágenes a color de la base de datos Cohn-Kanade [7]. Los resultados de detección son exitosos salvo por dos casos en la figura 2.9. La tasa de localización en imágenes en escala de grises es la misma que en [10] puesto que las imágenes de este tipo son tratadas directamente por medio del detector de Viola y Jones (ver figura 2.1).

2.5. Conclusión

El uso de segmentación cromática es un buen enfoque para obtener resultados más aproximados en el empleo de un detector de rostro basado en Adaboost. La contribución más importante en esta sección del trabajo es la combinación de la técnica propuesta por Finlayson [27] con la técnica de segmentación por centroides desarrollada por Zhang [28] junto con Adaboost. La integración de estas dos técnicas cromáticas ha mostrado ser una excelente alternativa para la segmentación de

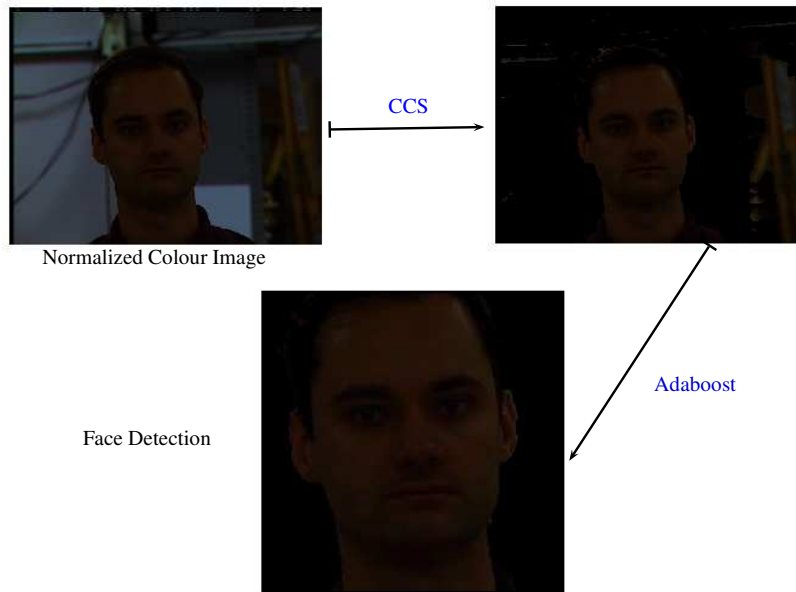


Figura 2.7: Esquema general de Resultados Finales. Imagen de la base de datos FERET [29] [30]

otros objetos en imágenes de color en sí misma y podría usarse con otro tipo de técnica posterior distinta de Adaboost.



Figura 2.8: Resultados Finales. Imágenes Base de Datos Cohn-Kanade [7]



Figura 2.9: Resultados Finales. Imágenes Base de Datos Cohn-Kanade [7]

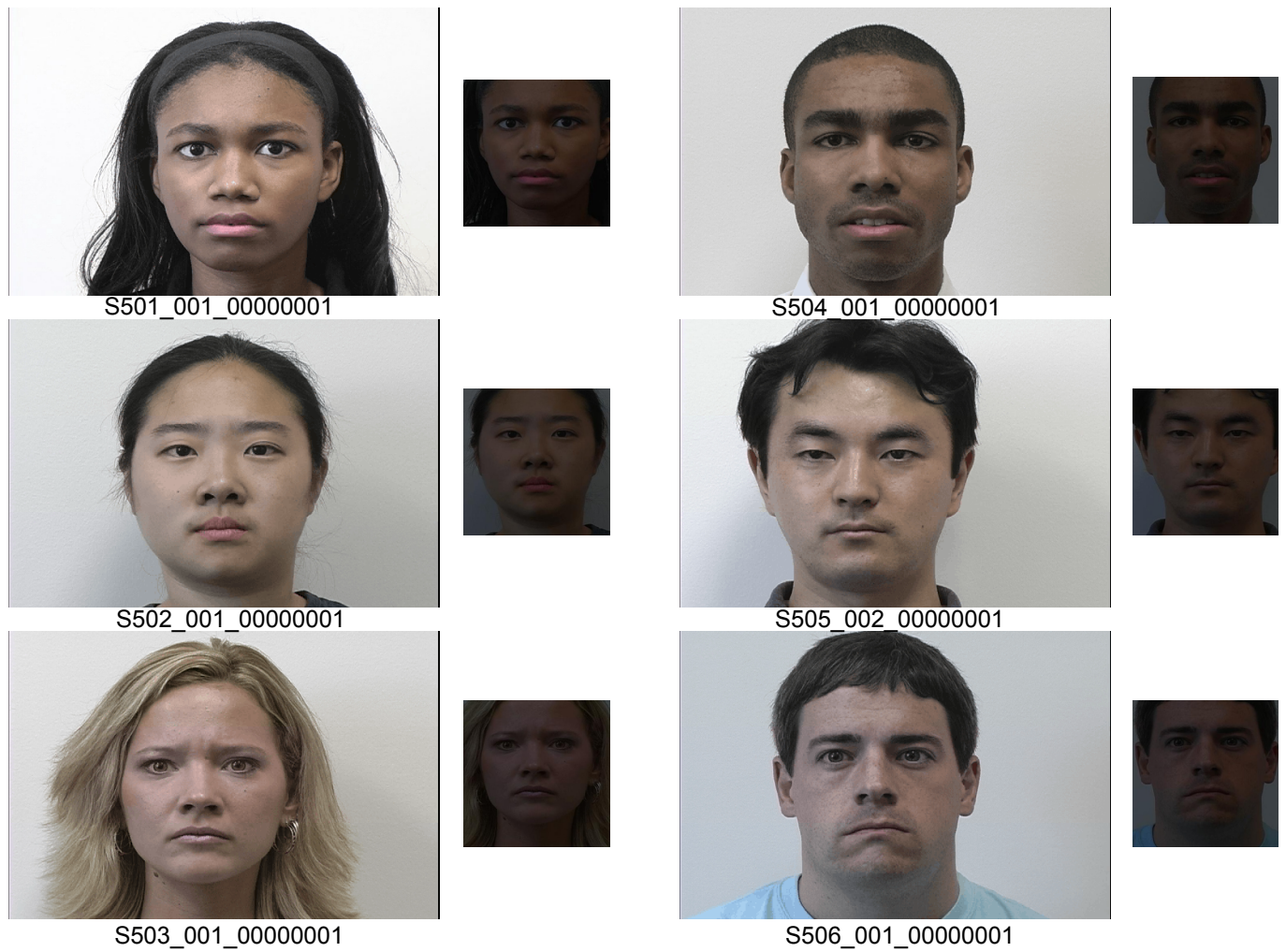


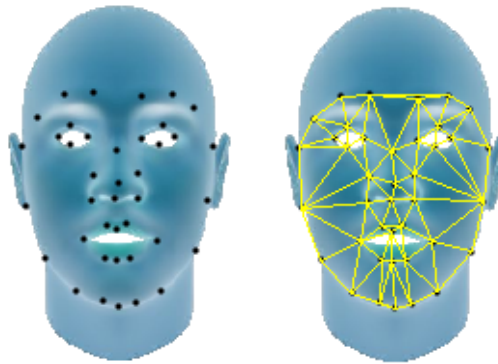
Figura 2.10: Resultados Finales. Imágenes Base de Datos Cohn-Kanade [7]



Figura 2.11: Resultados Finales. Imágenes Base de Datos Cohn-Kanade [7]

3

Localización Automática de Puntos Faciales



La localización automática de puntos faciales representa uno de los mayores desafíos en el análisis automático de expresiones, cualquiera sea su finalidad [1]. Extraer la información a partir de estos puntos brinda la posibilidad de analizar los patrones más relevantes involucrados en la imagen facial, sin necesidad de realizar *encajamientos* en espacios de mayor dimensión.

En este capítulo se presenta el esquema diseñado para la detección de puntos faciales, que se apoya y depende fuertemente del esquema de detección de rostros presentado en el capítulo anterior. Se escogen los *Active Appearance Models (AAM)* para llevar a cabo este propósito teniendo en cuenta sus ventajas y el hecho de que representan información consolidada para este fin.

En primera instancia se presenta una breve descripción del estado del arte en esta temática, mencionando los trabajos más importantes y/o relevantes para los propósitos de esta investigación. Posteriormente se dedicarán las demás secciones a la discusión de los detalles sobre la construcción y proceso de búsqueda con los *AAM*.

Finalmente se dará una presentación de los resultados obtenidos.

3.1. Estado del Arte

El problema de la detección de puntos faciales es aún hoy en día un problema abierto, en donde existen diferentes propuestas que se consolidan con el pasar de los años y las investigaciones [1] [2], pero que no ofrecen soluciones definitivas al problema.

Desde trabajos como los realizados por Chen [33] en donde se aprovechan argumentos de tipo probabilista a fin de conocer las regiones más probables para la ubicación de un punto facial con

base en un conjunto de imágenes de entrenamiento, hasta trabajos como el de Yun [34] en donde se busca emplear herramientas de clasificación de tipo estadístico para localizar los puntos faciales de acuerdo a los *features* asociados a una vecindad con centro en dicho punto, y pasando por trabajos como el de Hu [35] en donde se usan combinaciones lineales de modelos de rostro prototipo para la localización de las marcas faciales; los argumentos y propuestas son diversos. Sin embargo existen dos grandes enfoques que despiertan cada vez más el interés de los investigadores. El primero de éstos basado en *Active Appearance Models (AAM)*, como se evidencia en los trabajos [36] [37] [33], mientras que el segundo, donde destacan los trabajos de Pantic [38], Kozakaya [39] y recientemente Valstar [40], en los cuales la localización de cada punto facial es independiente de la localización de otro dentro de un mismo esquema.

El primer enfoque retoma las fortalezas y debilidades de los *AAM*, pero con el valor agregado de aprovechar el hecho de que es información consolidada, analizada, mejorada y criticada por diferentes investigadores, lo que permite conocer con gran profundidad sus limitaciones y potencialidades.

Los trabajos de Valstar [40] y Pantic [38] en el Imperial College of London, toman como referencia el esquema de detección de rostro desarrollado por Viola y Jones [10], buscando la localización de los puntos por medio de *features*. Valstar [40] propone usar el esquema de detección de rostro a fin de hacer más eficiente la búsqueda de los puntos, estableciendo regiones más probables para la búsqueda de acuerdo a unas imágenes de entrenamiento. Los *features* son extraídos empleando wavelets o máscaras de tipo Haar, para posteriormente usar clasificadores *boosted* o regresión de soporte vectorial.

Estos últimos trabajos dan indicios tener una fuerte fundamentación, además de ser trabajos promisorios. Sin embargo las publicaciones sobre éstos no son lo suficientemente detalladas como para realizar una implementación, y más importante, no permiten conocer en detalle sus limitaciones y fortalezas para su aplicación en un proyecto de investigación.

Por estas razones, en este trabajo se emplearán los *AAM* a fin de lograr la detección de puntos faciales, siguiendo parte de las ideas de Valstar [40] y Pantic [38] en donde se emplea el mismo esquema de detección de rostro, para realizar una búsqueda mucho más aproximada. La localización automática de estos puntos se realiza en cada *frame*, y no se consideran algoritmos de seguimiento ya que las secuencias de imágenes que se encuentran en la base de datos Cohn-Kanade [7], poseen tasas de muestreo arbitrarias.

3.2. Modelos de Apariencia Activa (AAM)

3.2.1. Formulación del Modelo de Forma

Para la construcción de un modelo de forma, se requiere de un conjunto de imágenes de entrenamiento anotadas con su respectivo conjunto de marcas [41]. Como el interés en este trabajo es en imágenes faciales se usan las imágenes de la base de datos Cohn-Kanade [7] después de emplear el esquema de detección presentado en el capítulo 2.

Las marcas representan los principales puntos geométricos del rostro (ver figura 3.1), y no existe un único camino para definirlas [1]. A fin de obtener el modelo de forma es necesario en primera instancia realizar un alineamiento del conjunto de formas empleado para llegar a sus características geométricas esenciales [42] [43] [41]. Dichas características geométricas son invariantes después de haber eliminado la rotación, el desplazamiento respecto al origen y el escalamiento [44] [45]. Una vez hecho esto se lleva a cabo la construcción de un modelo de tipo estadístico. Este procedimiento se detalla a continuación, por medio de varias etapas.

Alineamiento de Forma

Sea

$$\mathbf{x}_i = \left(x_i^{(1)}, x_i^{(2)}, x_i^{(3)}, \dots, x_i^{(n)}, y_i^{(1)}, y_i^{(2)}, y_i^{(3)}, \dots, y_i^{(n)} \right)^T$$

Un vector de $2n$ componentes, cuyas primeras n componentes representan las abscisas y las componentes de $n + 1$ hasta $2n$ son las ordenadas de un conjunto de marcas que representan la forma de la i -ésima imagen facial de un conjunto de entrenamiento. Los pasos del proceso de alineamiento se pueden presentar como [45] [44] [41]

1. Trasladar cada forma, de tal manera que su centroide se ubique en el origen.
2. Tome una forma \mathbf{x}_i como una estimado inicial del *mean shape* $\bar{\mathbf{x}}$ y escálelo de tal manera que $\|\bar{\mathbf{x}}\| = 1$.
3. Se hace $\bar{\mathbf{x}}_0 = \bar{\mathbf{x}}$

Se repite

4. Alinee todas las formas con la estimación actual del *mean shape*
5. Estimar nuevamente el *mean shape* como:

$$\bar{\mathbf{x}} = \frac{1}{n_S} \sum_{j=1}^{n_S} \mathbf{x}_j \quad (3.1)$$

donde n_S es el número de formas que serán alineadas.

6. Alinee $\bar{\mathbf{x}}$ a la referencia inicial $\bar{\mathbf{x}}_0$
7. Normalice el *mean shape* tal que $\|\bar{\mathbf{x}}\| = 1$

Hasta lograr convergencia

En el paso 4 se plantea alinear cada forma con el *mean shape*. Es decir este paso se reduce al alineamiento entre dos formas. Sea $\mathbf{x}_i, \mathbf{x}_j$, $i \neq j$ dos formas diferentes que se desean alinear. El proceso de alineación de la forma \mathbf{x}_i a la forma \mathbf{x}_j se plantea como el de encontrar los parámetros de una transformación de similaridad \mathcal{T} tal que

$$\mathcal{T}(\mathbf{x}_i) = \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \mathbf{x}_i + \begin{bmatrix} t_x \\ t_y \end{bmatrix} = \mathbf{x}_j \quad (3.2)$$

por supuesto no hay garantía de que existan a, b, t_x, t_y de tal manera que se cumpla una igualdad, por ello para llegar a los mejores parámetros se minimiza la función

$$E = \|\mathcal{T}(\mathbf{x}_i) - \mathbf{x}_j\|^2 \quad (3.3)$$

que llega a su valor mínimo seleccionando los parámetros con los siguientes valores

$$a = \frac{\mathbf{x}_i \cdot \mathbf{x}_j}{\|\mathbf{x}_i\|^2}, \quad b = \frac{\sum_{r=1}^n (\mathbf{x}_i(r)\mathbf{x}_j(n+r) - \mathbf{x}_i(r+n)\mathbf{x}_j(r))}{\|\mathbf{x}_i\|^2} \quad (3.4)$$

$$t_x = \frac{1}{n} \sum_{r=1}^n \mathbf{x}_j(r), \quad t_y = \frac{1}{n} \sum_{r=1}^n \mathbf{x}_j(n+r) \quad (3.5)$$

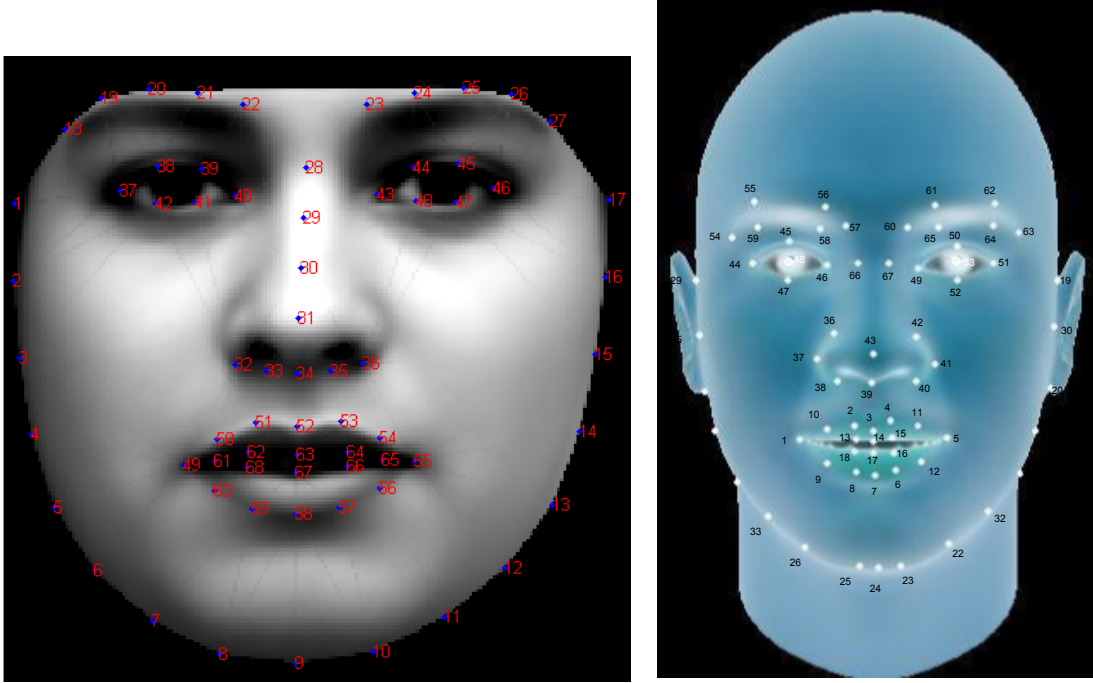


Figura 3.1: Izquierda: Conjunto de marcas asignadas con el estándar de la base de datos Cohn-Kanade [7]. Derecha: Conjunto de marcas asignadas siguiendo el esquema de Cootes [1]

Análisis de Componentes Principales

Para la construcción del modelo estadístico se emplea análisis de componentes principales. Primero se computa la matriz de covarianza \mathbf{S}_s de los datos como [42] [43] [41]

$$\mathbf{S}_s = \frac{1}{n_S - 1} \sum_{i=1}^{n_S} (\mathbf{x}_i - \bar{\mathbf{x}}) (\mathbf{x}_i - \bar{\mathbf{x}})^T \quad (3.6)$$

y entonces se calculan los eigenvectores ϕ_{sk} correspondientes a los eigenvalores λ_{sk} , ordenados tal que $\lambda_{sk} \geq \lambda_{s(k+1)}$. Si Φ_s es la matriz cuyos vectores columna son los eigenvectores ϕ_{sk} correspondientes a los q_s eigenvalores más grandes, el valor de cualquier \mathbf{x} puede ser aproximado por

$$\mathbf{x} \approx \bar{\mathbf{x}} + \Phi_s \mathbf{b}_s \quad (3.7)$$

donde \mathbf{b}_s es un vector que contiene el conjunto de parámetros del modelo deformable. Variando los elementos de \mathbf{b}_s se puede variar la forma \mathbf{x} usando la ecuación 3.7. La varianza del i -ésimo parámetro, b_{si} , en el conjunto de entrenamiento está dado por λ_{si} [46]. El número de eigenvectores a retener, q_s , puede ser escogido de tal manera que se conserve una porción de la varianza total de los datos [46] [42] [43], acorde a Cootes et al [42] [43] un criterio para tomar este porcentaje es

$$\frac{\sum_{k=1}^{q_s} \lambda_k}{\sum_{k=1} \lambda_k} \geq 0,98$$

es decir, tomando el 98 % de la energía total de los eigenvalores.

3.2.2. Formulación del Modelo de Textura

Como en el caso del modelo de forma, para la construcción del modelo de textura se requiere de un conjunto de imágenes de entrenamiento, que es el mismo conjunto empleado para la construcción del

modelo de forma. Una vez se cuenta con estas imágenes, se realiza un procedimiento de normalización o alineación para así construir un modelo estadístico empleando análisis de componentes principales [42] [43].

Es importante dejar claro que las muestras de textura se toman dentro del *convex hull* obtenido de las formas no alineadas [41].

Image Warping

Usando el *image warping* se mapea la textura de las muestras dentro de una forma a otra [41]. Para esto se construye un enmallado triangular cuyos vértices son las marcas que definen la forma del objeto, y la textura del mismo se encontrará en el *convex hull* de dicha forma. En esta construcción de enmallados se usa usualmente el algoritmo *Delaunay* [41].

Una vez realizado esto, se mapean los niveles de gris correspondientes a los pixeles dentro de cada triángulo de la forma original al triángulo asociado de la otra forma [41]. Para ilustrar esto considere el diagrama de la figura 3.2, sea \mathbf{x}_r la forma de un objeto y $\tilde{\mathbf{x}}_r$ otra forma asociada. Sean $\mathbf{v}_{(1)}, \mathbf{v}_{(2)}, \mathbf{v}_{(3)}$ los vectores que representan los vértices de un triángulo $\Delta_{\mathbf{v}_{(1,2,3)}}$ sobre el enmallado del objeto \mathbf{x}_r tomados en sentido contrario a las manecillas del reloj, mientras que $\mathbf{v}'_{(1)}, \mathbf{v}'_{(2)}, \mathbf{v}'_{(3)}$ son los vectores correspondientes al triángulo $\Delta_{\mathbf{v}'_{(1,2,3)}}$ asociado a $\Delta_{\mathbf{v}_{(1,2,3)}}$ sobre la forma $\tilde{\mathbf{x}}_r$. Entonces el valor del pixel localizado en $\mathbf{v} = [x, y]^T$ dentro del triángulo $\Delta_{\mathbf{v}_{(1,2,3)}}$ es mapeado a $\mathbf{v}' = [x', y']^T$ dentro del triángulo $\Delta_{\mathbf{v}'_{(1,2,3)}}$ de la siguiente manera

$$\mathcal{T}_{warp}(\mathbf{v}) = \begin{bmatrix} x' \\ y' \end{bmatrix} = \alpha \mathbf{v}'_{(1)} + \beta \mathbf{v}'_{(2)} + \gamma \mathbf{v}'_{(3)} \quad (3.8)$$

donde

$$\alpha = 1 - (\beta + \gamma) \quad (3.9)$$

$$\beta = \frac{yv_{x3} - v_{x1}y - v_{x3}v_{y1} - v_{y3}x + v_{x1}v_{y3} + xv_{y1}}{-v_{x2}v_{y3} + v_{x2}v_{y1} + v_{x1}v_{y3} + v_{x3}v_{y2} - v_{x3}v_{y1} - v_{x1}v_{y2}} \quad (3.10)$$

$$\gamma = \frac{xv_{y2} - xv_{y1} - v_{x1}v_{y2} - v_{x2}y + v_{x2}v_{y1} + v_{x1}y}{-v_{x2}v_{y3} + v_{x2}v_{y1} + v_{x1}v_{y3} + v_{x3}v_{y2} - v_{x3}v_{y1} - v_{x1}v_{y2}} \quad (3.11)$$

y $\mathbf{v}_{(i)} = [v_{xi}, v_{yi}]^T$, $i = 1, 2, 3$. Una vez hecho este mapeo es necesario llevar a cabo una interpolación para definir los niveles de gris en los cuales no se mapea ningún valor de textura [41].

Aunque este procedimiento se define entre dos formas generales que cuentan con enmallados cuyos números de triángulos y vértices son iguales, se considera que la segunda forma es una versión deformada de la primera por medio de alguna transformación [41]. En la figura 3.3 se evidencia la operación de *image warping* sobre una imagen de la base de datos FERET [29] [30].

Alineación de Textura

Para minimizar los efectos de variaciones globales de iluminación, se normaliza el conjunto de imágenes empleadas para construir el modelo buscando dejar varianza unitaria y media nula [42] [43] [1].

Sea \mathbf{g}_i el vector columna de las muestras de textura (los niveles de gris en la imagen en escala de grises) de la imagen i en el interior del *convex hull* de la forma no alineada \mathbf{x}_i . \mathbf{g}_i es la representación vectorial de la imagen por lo cual allí habrá una cantidad de ceros asociada a los pixeles que estén fuera del *convex hull* de \mathbf{x}_i . El proceso de normalización se puede presentar como

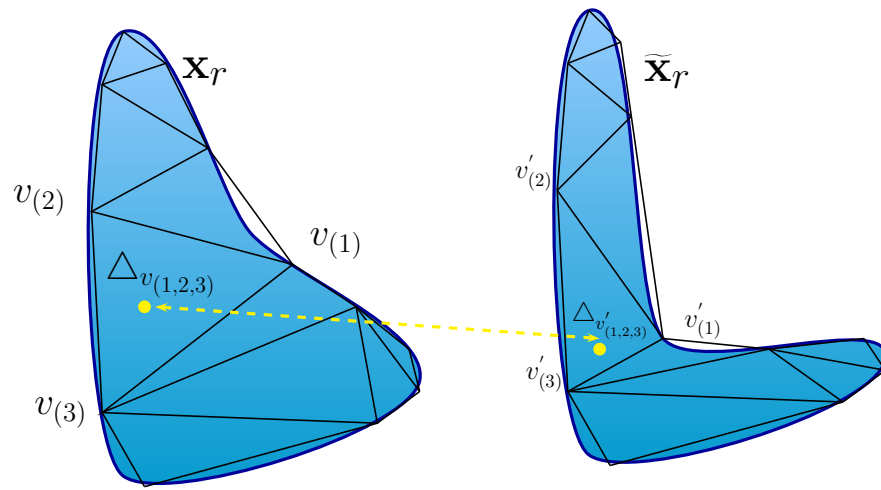


Figura 3.2: Diagrama que ilustra el *Image Warping*, en donde \mathbf{x}_r es una versión deformada de $\tilde{\mathbf{x}}_r$

1. Haga el *warping* de las muestras de textura dentro del *convex hull* de \mathbf{x}_i para la imagen original i al *mean shape* ubicado en el centro de la imagen y con un escalamiento que es el promedio de todos los tamaños de los rostros en el conjunto de imágenes de entrenamiento. Esto se representa como $\mathbf{g}_i \leftarrow \mathcal{T}_{warp}(\mathbf{g}_i)$.

2. Calcule para todos los \mathbf{g}_i

$$\mathbf{g}_i = \frac{\mathbf{g}_i - \beta \mathbf{1}}{\sqrt{\alpha}} \quad (3.12)$$

donde

$$\beta = \frac{\mathbf{g}_i \cdot \mathbf{1}}{n_S} \quad (3.13)$$

$$\alpha = \frac{\|\mathbf{g}_i\|^2}{n_S} - \beta^2 \quad (3.14)$$

aquí $\mathbf{1}$ es el vector cuyas componentes son todas iguales a 1. Este no es el único camino para realizar esta normalización, Cootes en [42] propone otra forma para realizar esta normalización con resultados semejantes a los presentados en este trabajo y en el trabajo de Stegmann [41].

Analisis de Componentes Principales

Una vez las muestras de textura se han normalizado, se calcula la matriz de covarianza¹ [42] [43] [41] como

$$\mathbf{S}_g = \frac{1}{n_S - 1} \sum_{i=1}^{n_S} (\mathbf{g}_i - \bar{\mathbf{g}}) (\mathbf{g}_i - \bar{\mathbf{g}})^T \quad (3.15)$$

y entonces se computan los eigenvectores $\phi_{\mathbf{g}k}$ correspondientes a los eigenvalores $\lambda_{\mathbf{g}k}$, ordenados como $\lambda_{\mathbf{g}k} \geq \lambda_{\mathbf{g}(k+1)}$. Si $\Phi_{\mathbf{g}}$ es la matriz cuyos vectores columna son los eigenvectores $\phi_{\mathbf{g}k}$ correspondientes a los $q_{\mathbf{g}}$ eigenvalores más grandes, el valor de cualquier \mathbf{g}_i puede ser aproximado como

$$\mathbf{g} \approx \bar{\mathbf{g}} + \Phi_{\mathbf{g}} \mathbf{b}_{\mathbf{g}} \quad (3.16)$$

donde $\mathbf{b}_{\mathbf{g}}$ es un vector que representa un conjunto de parámetros del modelo. Variando los elementos de $\mathbf{b}_{\mathbf{g}}$ se puede modificar la textura \mathbf{g} usando la ecuación 3.16. La varianza del i -ésimo parámetro $b_{\mathbf{g}i}$ para el conjunto de entrenamiento está dado por $\lambda_{\mathbf{g}i}$ [46]. El número de eigenvectores a retener $q_{\mathbf{g}}$, puede ser escogido tal que el modelo represente una porción de la varianza total de los datos [46] [42] [43]. Acorde a Cootes et al, un criterio que puede tomarse es, al igual que en el modelo de forma

$$\frac{\sum_{k=1}^{q_{\mathbf{g}}} \lambda_k}{\sum_{k=1} \lambda_k} \geq 0,98$$

3.2.3. Formulación del Modelo Combinado

Se puede obtener una formulación combinada del modelo de apariencia aplicando análisis de componentes principales a los vectores

$$\mathbf{b} = \begin{bmatrix} \mathbf{W}_s \mathbf{b}_s \\ \mathbf{b}_g \end{bmatrix} = \begin{bmatrix} \mathbf{W}_s \Phi_s^T (\mathbf{x} - \bar{\mathbf{x}}) \\ \Phi_g^T (\mathbf{g} - \bar{\mathbf{g}}) \end{bmatrix} \quad (3.17)$$

¹Generalmente el tamaño de las imágenes hace inaceptable el costo computacional asociado a esta matriz de covarianza, por lo cual es necesario realizar una proyección. Con ésta se podrá encontrar los eigenvalores de una matriz más pequeña, y cuyos eigenvectores permitirán por medio de dicha proyección obtener los eigenvectores de la matriz de covarianza original [47] [48]

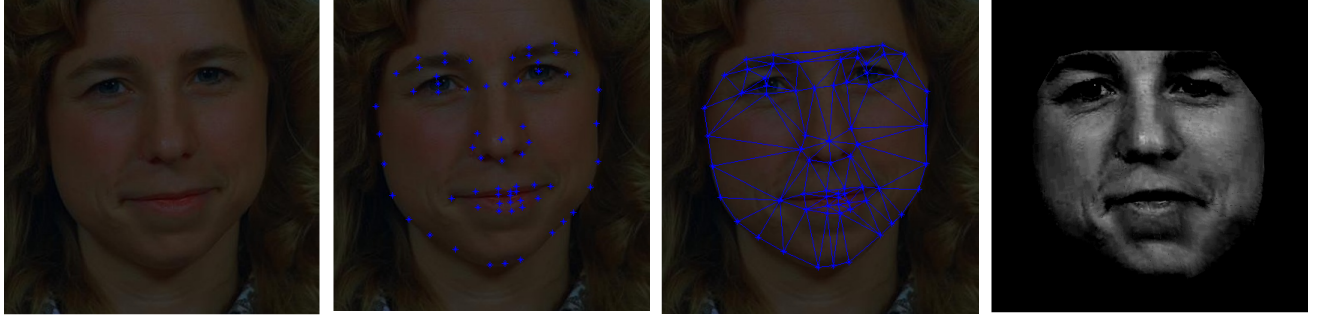


Figura 3.3: De izquierda a derecha: a) Imagen Original, b) Imagen con Marcas asignadas, c) Enmallado construido usando las marcas, d) Textura en el interior del *convex hull* del enmallado mapeada sobre la forma media usando *image warping*

en donde \mathbf{W}_s es una matriz de escalamiento para adecuar unidades. Si se denota la matriz de estos nuevos eigenvectores como $\hat{\Phi}$, entonces el nuevo modelo tiene la siguiente forma

$$\mathbf{b} = \hat{\Phi} \mathbf{c} \quad (3.18)$$

Los vectores \mathbf{c} son llamados parámetros de apariencia [49] [50] [51]. Con esto, se puede controlar la forma y el modelo de textura. Puesto que el modelo de forma y textura tienen media cero, \mathbf{c} también tiene media cero [42] [43]. Entonces el modelo combinado puede ser escrito como

$$\mathbf{x} = \bar{\mathbf{x}} + \Phi_s \mathbf{W}_s^{-1} \hat{\Phi}_{cs} \mathbf{c} \quad (3.19)$$

$$\mathbf{g} = \bar{\mathbf{g}} + \Phi_g \hat{\Phi}_{cg} \mathbf{c} \quad (3.20)$$

donde

$$\hat{\Phi} = \begin{bmatrix} \hat{\Phi}_{cs} \\ \hat{\Phi}_{cg} \end{bmatrix} \quad (3.21)$$

En la figura 3.4 se puede apreciar el comportamiento del modelo combinado variando los parámetros del vector \mathbf{c} .

Selección de parámetros de ponderación de forma \mathbf{W}_s

En este trabajo se usa la propuesta realizada por Cootes [42], en la que \mathbf{W}_s es seleccionado como:

$$\mathbf{W}_s = r \mathbf{I} \quad (3.22)$$

donde \mathbf{I} es la matriz identidad, y $r > 0$ es tal que

$$r^2 = \frac{\sum_{k=1}^{q_g} \lambda_{gk}}{\sum_{k=1}^{q_s} \lambda_{sk}} \quad (3.23)$$

3.2.4. Búsqueda en *AAM*

La búsqueda en los *Active Appearance Models* requiere de una etapa de entrenamiento, en la cual se construyen cantidades requeridas para la siguiente etapa, que es finalmente el proceso de minimización de una función de error para acoplar el modelo a la imagen objetivo.



Figura 3.4: De izquierda a Derecha: a) Valor del modelo haciendo $c_1 = -\sqrt{\lambda_1}$, $c_i = 0 \quad \forall i > 1$, b) Textura media, sobre la forma media (*Mean Shape*) c) Valor del modelo haciendo $c_1 = \sqrt{\lambda_1}$, $c_i = 0 \quad \forall i > 1$, d) Representación de los cambios de forma por la variación de c_1 (Azul: *Mean Shape*, Verde: $c_1 = \sqrt{\lambda_1}$, Rojo: $c_1 = -\sqrt{\lambda_1}$)

Para realizar este proceso es necesario aplicar una transformación de similaridad sobre el modelo, con la que se busca representar la variación del modelo combinado en el frame de la imagen. Esta transformación se representa para cualquier punto $\mathbf{v} = [x, y]^T$ como

$$\mathcal{T}_{\mathbf{t}}(\mathbf{v}) = \begin{bmatrix} 1 + s_x & -s_y \\ s_y & 1 + s_x \end{bmatrix} \mathbf{v} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \quad (3.24)$$

$\mathbf{t} = (t_1, t_2, t_3, t_4) = (s_x, s_y, t_x, t_y)$ donde $s_x = s \cos(\theta) - 1$, $s_y = s \sin(\theta)$. θ es la rotación, y s el escalado, mientras que (t_x, t_y) es la traslación.

A continuación se detallan las dos etapas que describen la búsqueda con los *AAM*

Entrenamiento: Corrección de los Parámetros del Modelo

La principal idea para el entrenamiento en la corrección de los valores de los parámetros, es usar las condiciones de imágenes conocidas en el modelo. Se hace una perturbación controlada sobre éstas y se almacena el error. Entonces, todos los errores son almacenados en una matriz que es usada para realizar la búsqueda en el frame de la imagen.

Sean $\mathbf{c}_{i,0}$ y $\mathbf{t}_{i,0}$ los valores de los parámetros del modelo y de la transformación de similaridad con los cuales el modelo genera el elemento \mathbf{g}_i de la imagen i . Se realiza una perturbación conocida $\delta\mathbf{c}, \delta\mathbf{t}$ para cada parámetro *individualmente* como

$$\mathbf{c} = \mathbf{c}_{i,0} + \delta\mathbf{c} \quad (3.25)$$

$$\mathbf{t} = \mathbf{t}_{i,0} + \delta\mathbf{t}, \quad (3.26)$$

de manera que

$$\mathcal{T}_{\mathbf{t}_{i,0} + \delta\mathbf{t}}(\mathbf{x}_m) = \mathcal{T}_{\mathbf{t}_{i,0}}(\mathcal{T}_{\delta\mathbf{t}}(\mathbf{x}_m)) = \hat{\mathbf{x}}_m \quad (3.27)$$

$$\mathbf{g}_m = \bar{\mathbf{g}} + \Phi_g \hat{\Phi}_{cg}(\mathbf{c}_{i,0} + \delta\mathbf{c}) \quad (3.28)$$

$$\mathbf{x}_m = \bar{\mathbf{x}} + \Phi_s \mathbf{W}_s^{-1} \hat{\Phi}_{cs}(\mathbf{c}_{i,0} + \delta\mathbf{c}) \quad (3.29)$$

$\hat{\mathbf{g}}_m$ es la textura generada en el marco de la imagen i haciendo *warping* de \mathbf{g}_m a $\hat{\mathbf{x}}_m$, entonces el error se puede computar como

$$\Xi_{\mathbf{g}} = \mathbf{g}_s - \hat{\mathbf{g}}_m \quad (3.30)$$

donde \mathbf{g}_s es la muestra de textura dentro del *convex hull* de $\hat{\mathbf{x}}_m$ que es el modelo de forma perturbado en el marco de la imagen i .

Sea $\mathbf{g}_{\partial^k p_i}$ el error computado por perturbación del parámetro p_i por un valor de $\zeta(k)$, $k = 1, 2, 3, \dots, d$ y sea \mathbb{A} la matriz cuyos vectores columna \mathbf{a}_i son de la forma

$$\mathbf{a}_i = \frac{1}{d} \sum_{k=1}^d w_k \mathbf{g}_{\partial^k p_i} \quad (3.31)$$

donde

$$w_k = \frac{1}{p_k} e^{-\frac{(\delta p_k)^2}{2\sigma_k^2}} \quad (3.32)$$

σ_k es la desviación estándar, si $p_k = c_k$ entonces $\sigma_k = \sqrt{\lambda_k}$ y si $p_k = t_k$ σ_k tiene otro valor dependiendo de las perturbaciones realizadas en los t_i .

Ahora, con esta notación y acorde a [43] [52] se construye

$$\hat{\mathbb{R}} = (\mathbb{A}\mathbb{A}^T)^{-1} \mathbb{A} \quad (3.33)$$

Refinamiento Iterativo del modelo

Para el refinamiento iterativo del modelo, se siguen los siguientes pasos [42] [43] [52]:

1. Se inicia con $\mathbf{c} = \mathbf{0}, t = t_0$ siendo t_0 la condición inicial de la transformación que ubica \mathbf{x}_m en el marco de referencia de la imagen con un escalamiento que es el promedio de los tamaños de las imágenes sin alinear, junto con un desplazamiento al centro de la imagen. Entonces

$$\mathbf{g}_m = \bar{\mathbf{g}} \quad (3.34)$$

$$\mathbf{x}_m = \bar{\mathbf{x}} \quad (3.35)$$

y $\mathcal{T}_{t_0}(\mathbf{x}_m) = \hat{\hat{\mathbf{x}}}_m$ es el marco de referencia asociado al modelo de textura.

2. Se proyecta la muestra de textura \mathbf{g}_s dentro de $\hat{\hat{\mathbf{x}}}_m$ haciendo normalización para que tenga media cero y varianza unitaria, obteniendo un nuevo \mathbf{g}_s .
3. Evalúe el error $\mathbf{\Xi}_g = \mathbf{g}_s - \hat{\mathbf{g}}_m$, y el error actual, $E = \|\mathbf{\Xi}_g\|^2$.
4. Calcule los desplazamientos $\delta \mathbf{p} = -\hat{\mathbb{R}} \mathbf{\Xi}_g$, donde $\mathbf{p}^T = (\mathbf{c}^T | \mathbf{t}^T)$
5. Actualice los parámetros del modelo $\mathbf{p} \rightarrow \mathbf{p} + \zeta(k) \delta \mathbf{p}$ donde inicialmente $k = 1$, y

$$\zeta(k) \delta \mathbf{p} = \begin{bmatrix} \zeta_{\mathbf{c}}(k) \delta c_1 \\ \zeta_{\mathbf{c}}(k) \delta c_2 \\ \vdots \\ \zeta_{\mathbf{c}}(k) \delta c_n \\ \zeta_{\mathbf{t}}(k) \delta t_1 \\ \vdots \\ \zeta_{\mathbf{t}}(k) \delta t_4 \end{bmatrix} \quad (3.36)$$

6. Calcule los nuevos puntos, $\hat{\hat{\mathbf{x}}}'_m$ y el modelo de textura en el marco de la imagen $\hat{\hat{\mathbf{g}}}'_m$.
7. Tome las muestras de textura de la imagen en el interior del *convex hull* de $\hat{\hat{\mathbf{x}}}'_m$ y normalice para obtener \mathbf{g}'_s .

8. Calcule el nuevo vector de error, $\Xi'_g = g'_s - \hat{g}'_m$.
9. Si $\|\Xi'_g\|^2 < E$, entonces acepte el nuevo estimado; de lo contrario se prueba con $k = 0,5; k = 0,25; \text{ etc.}$

Cootes et al [42] [43] y Stegmann en [52] han demostrado que la selección óptima de los desplazamientos se realiza como

$$\delta c_i = k \sqrt{\lambda_i} \quad k \in \mathbb{Q} \quad (3.37)$$

teniendo en cuenta esto, en este trabajo se selecciona

$$\zeta_c(k) = \frac{1}{5}(k - 3,5), \quad \forall k \in [1, 6], k \in \mathbb{Z} \quad \text{y} \quad \delta c_i = \sqrt{\lambda_i} \quad (3.38)$$

para los desplazamientos en los parámetros de la transformación de similaridad se hace

$$\delta t_i = 1, \quad \forall i \quad (3.39)$$

y se selecciona

$$\zeta_{t|t_1}(k) = \zeta_{t|t_2}(k) = \frac{1}{5}(k - 3,5), \quad \forall k \in [1, 6], k \in \mathbb{Z} \quad (3.40)$$

$$\zeta_{t|t_3}(k) = \zeta_{t|t_4}(k) = \frac{2(k - 3,5)}{s_{x0} + 1}, \quad \forall k \in [1, 6], k \in \mathbb{Z} \quad (3.41)$$

donde $s_{x0} + 1 = s_{x0} \cos(\theta_0)$ define los parámetros de la transformación de similaridad sin perturbación, tal que g_i es generado en el frame de la imagen como $\mathcal{T}_{t_i,0}(g_i) = \hat{g}_i$.

3.3. Implementación

Para realizar la implementación de los AAM, se empleó la herramienta de simulación Matlab. Los tiempos de búsqueda por imagen oscilan entre los 3 y 5 minutos aproximadamente en un computador con características promedio. En el entrenamiento se requiere de un tiempo del orden de horas.

3.4. Resultados

3.4.1. Evaluación Cuantitativa

El interés principal de este trabajo en el uso de los modelos de apariencia activa es la localización de puntos faciales, en donde una medida de la calidad de esta localización puede estar asociada al comportamiento final en textura del modelo. Sin embargo este aspecto no es definitivo, teniendo en cuenta que en muchos casos el acople del enmallado final no está siempre relacionado con errores finales pequeños, por ello se establece un umbral de error satisfactorio valorando los resultados experimentales.

En las figuras 3.5, 3.6, 3.7, 3.8, 3.9, 3.10, 3.11, 3.12, 3.13, 3.14, 3.15, 3.16, 3.17, 3.18, 3.19, se muestran los resultados finales de detección de puntos faciales sobre un conjunto de 100 imágenes de la base de datos Cohn-kanade [7] no usadas ni en el entranamiento ni en la construcción del modelo. Los resultados por imagen individual se encuentran organizados en renglones. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla

3.1.

Para la construcción del modelo fueron empleadas 244 imágenes de la base de datos Cohn-Kanade [7] usando el estándar de marcas asociado a la misma (ver figura 3.1). El entrenamiento fue realizado sobre un conjunto de 193 imágenes. Teniendo en cuenta todos estos resultados el umbral establecido para considerar como satisfactoria la búsqueda es de 95. Así, 84 de las pruebas consignadas en la tabla 3.1 tienen un error final inferior a este umbral.

3.5. Discusión y Conclusiones

El uso del esquema de detección de rostro planteado en el capítulo 2, permitió mejorar los resultados obtenidos en el uso de *AAM*, ya que se redujo en gran medida el grado de subjetividad para indicar la condición inicial del modelo respecto de la imagen. Esto representa una opción para los fines de extracción de información.

La forma de entrenamiento requerida para realizar el proceso de búsqueda con los *AAM* implica que para un buen funcionamiento y comportamiento de los mismos debe haber una gran correlación entre los patrones faciales del sujeto en la imagen objetivo y las imágenes de entrenamiento. Por ello el escenario más pertinente para la aplicación de este desarrollo es uno en el cual es posible con unas cuantas imágenes de un individuo, identificar en una secuencia arbitraria y no conocida de imágenes del mismo, los puntos faciales de interés.

Imagen	Error Final	Iteraciones		Imagen	Error Final	Iteraciones
1	48.1108	44		51	83.6237	6
2	98.5801	21		52	74.0367	6
3	79.7742	16		53	71.9703	14
4	48.4909	43		54	34.9860	57
5	59.5254	40		55	43.7373	37
6	56.9495	35		56	50.7269	32
7	58.0280	29		57	50.7269	49
8	66.8353	16		58	94.7319	5
9	71.0264	28		59	11.30386	5
10	40.9431	52		60	97.8205	25
11	67.6079	14		61	75.9192	35
12	67.1339	17		62	77.3782	26
13	62.3094	39		63	71.2283	17
14	100.9103	15		64	52.7941	55
15	101.4590	17		65	67.9733	17
16	69.0196	16		66	86.0298	17
17	68.7041	25		67	54.4744	58
18	75.1369	20		68	108.9230	6
19	73.2374	9		69	57.9465	51
20	48.3397	25		70	56.0894	81
21	51.1266	36		71	114.6948	5
22	73.9588	14		72	49.6707	78
23	71.9316	8		73	94.0578	5
24	69.2547	20		74	54.1309	54
25	78.1378	12		75	108.6562	8
26	86.3782	8		76	104.2382	7
27	85.2431	9		77	87.5434	11
28	74.6489	10		78	44.4923	59
29	53.1672	27		79	95.8944	18
30	60.4367	33		80	72.1129	16
31	39.7533	55		81	62.2984	23
32	79.2244	18		82	66.0403	20
33	71.1190	18		83	64.6535	58
34	40.4023	47		84	49.0298	91
35	81.1578	7		85	68.8150	17
36	92.4814	7		86	55.9712	59
37	92.4814	36		87	100.4090	6
38	48.0576	43		88	58.0705	55
39	41.5301	42		89	95.1503	4
40	97.3068	7		90	81.6702	11
41	53.8275	46		91	43.2614	60
42	54.6813	33		92	59.9984	18
43	34.7397	52		93	60.6217	23
44	99.3486	5		94	67.4194	22
45	99.4995	4		95	76.5257	16
46	85.1043	6		96	69.8225	23
47	79.6058	5		97	52.9968	47
48	43.0793	45		98	40.8333	56
49	77.3700	25		99	50.7203	46
50	75.8155	31		100	104.9347	5

Cuadro 3.1: Resultados AAM

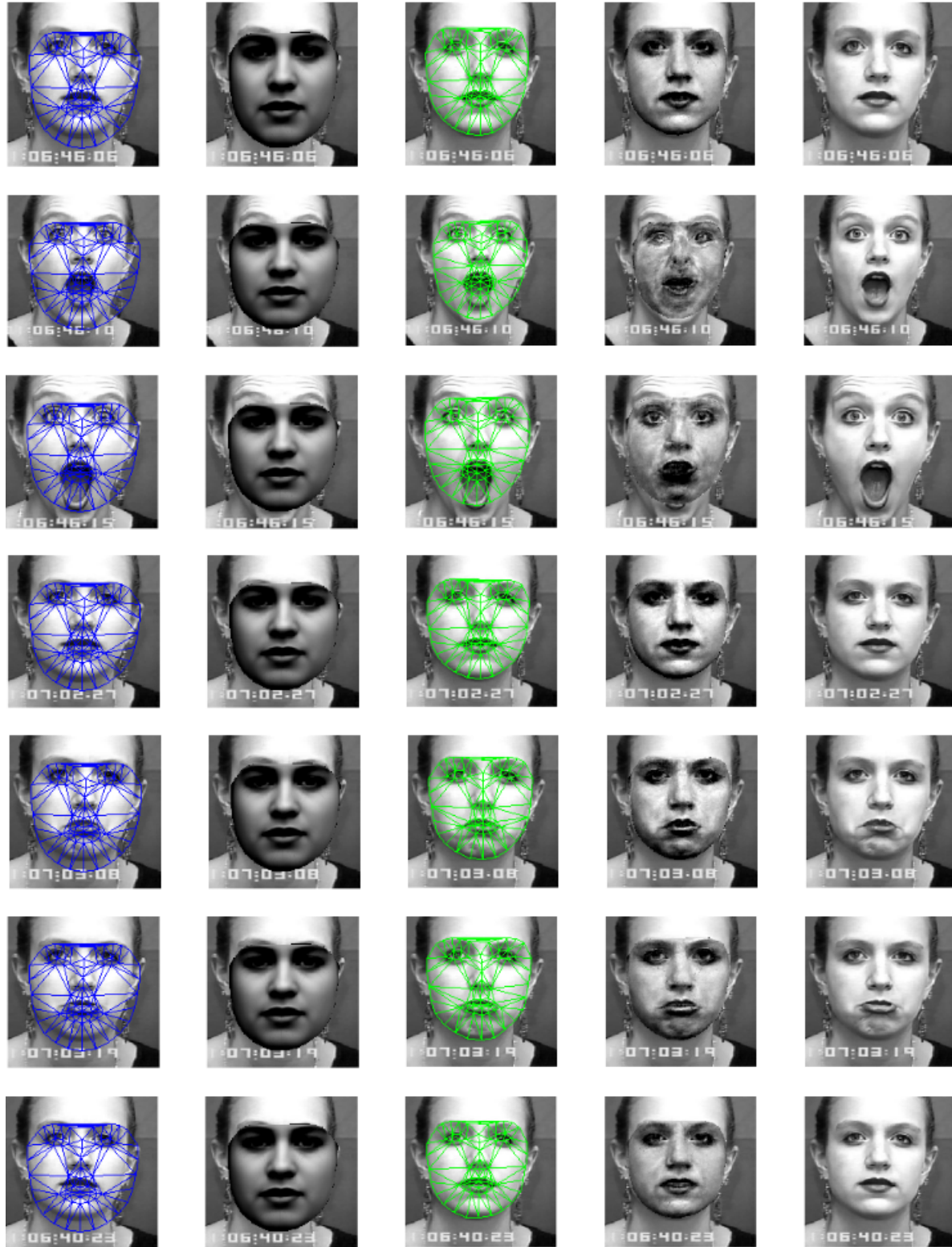


Figura 3.5: Resultados Imágenes 1-7. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1

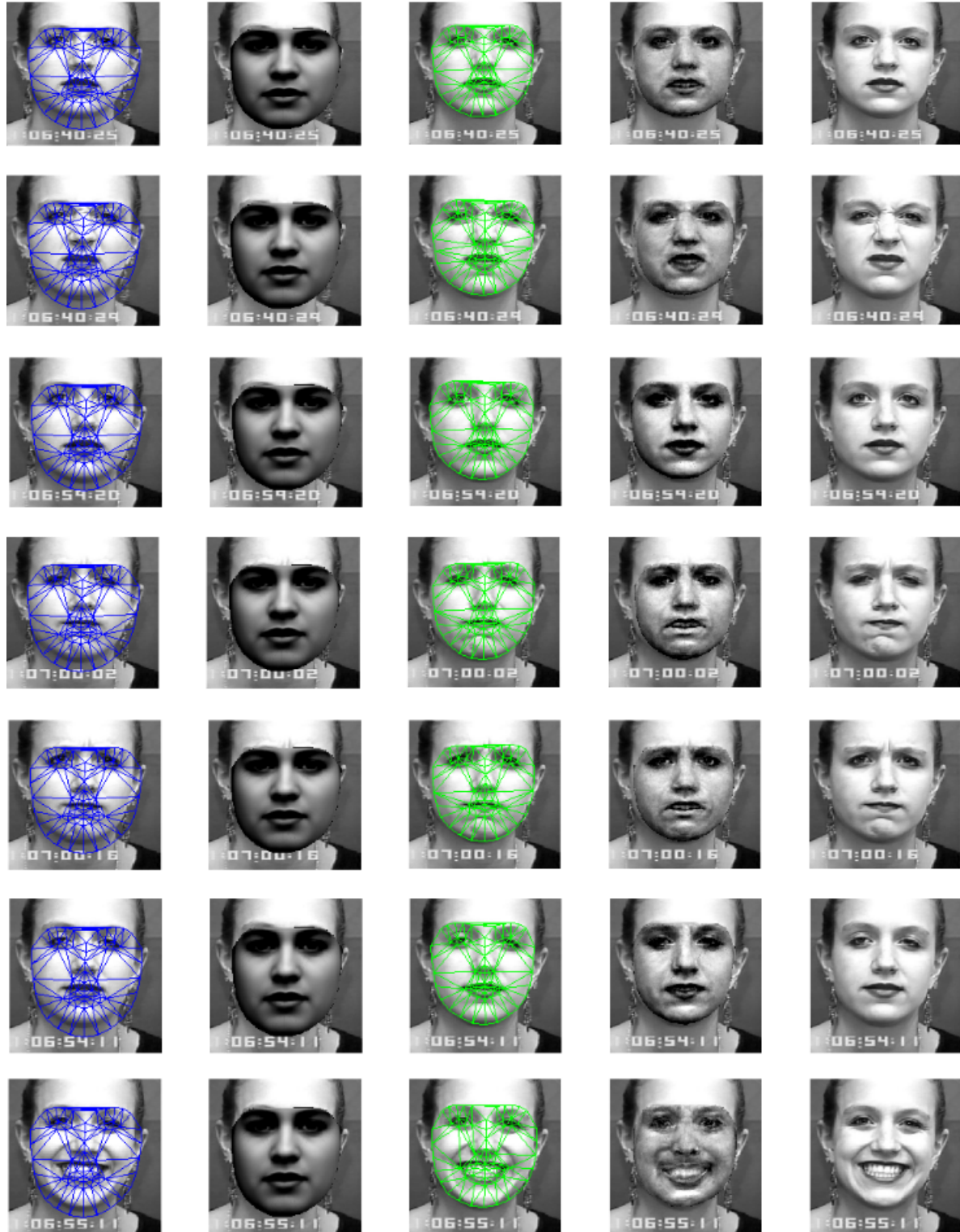


Figura 3.6: Resultados Imágenes 8-14. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1

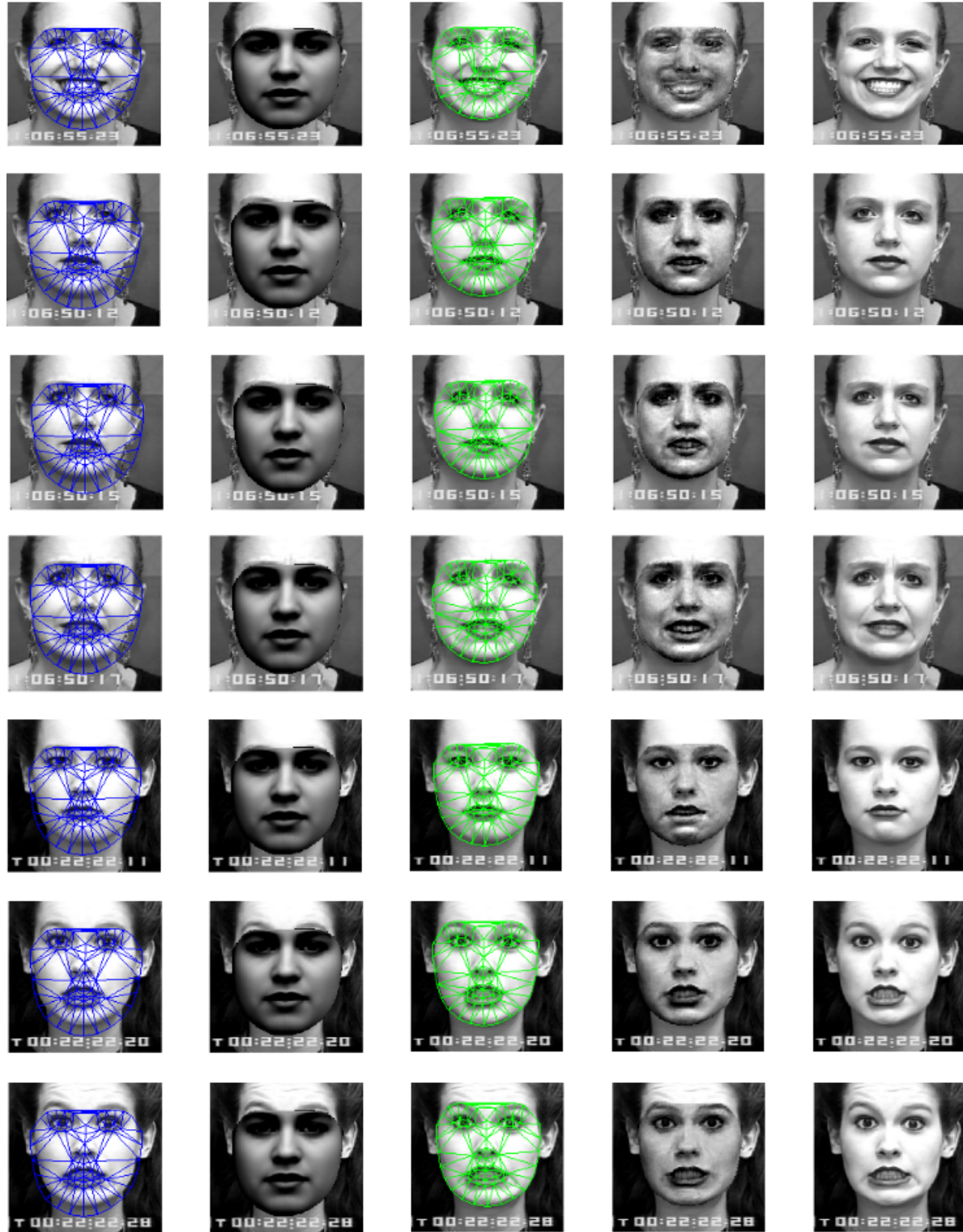


Figura 3.7: Resultados Imágenes 15-21. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1



Figura 3.8: Resultados Imágenes 22-28. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1



Figura 3.9: Resultados Imágenes 29-35. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1



Figura 3.10: Resultados Imágenes 36-42. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1

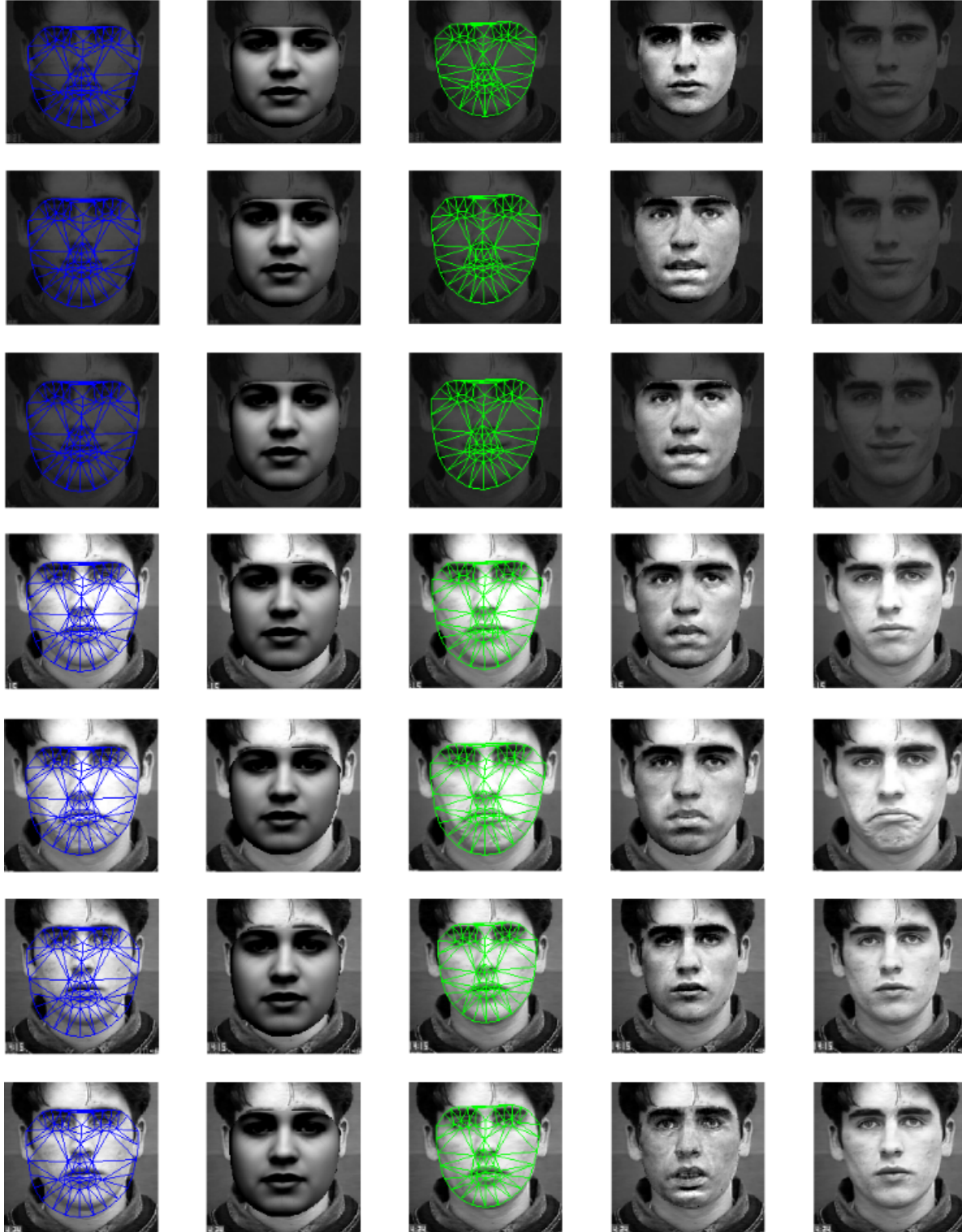


Figura 3.11: Resultados Imágenes 43-49. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1



Figura 3.12: Resultados Imágenes 50-56. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1

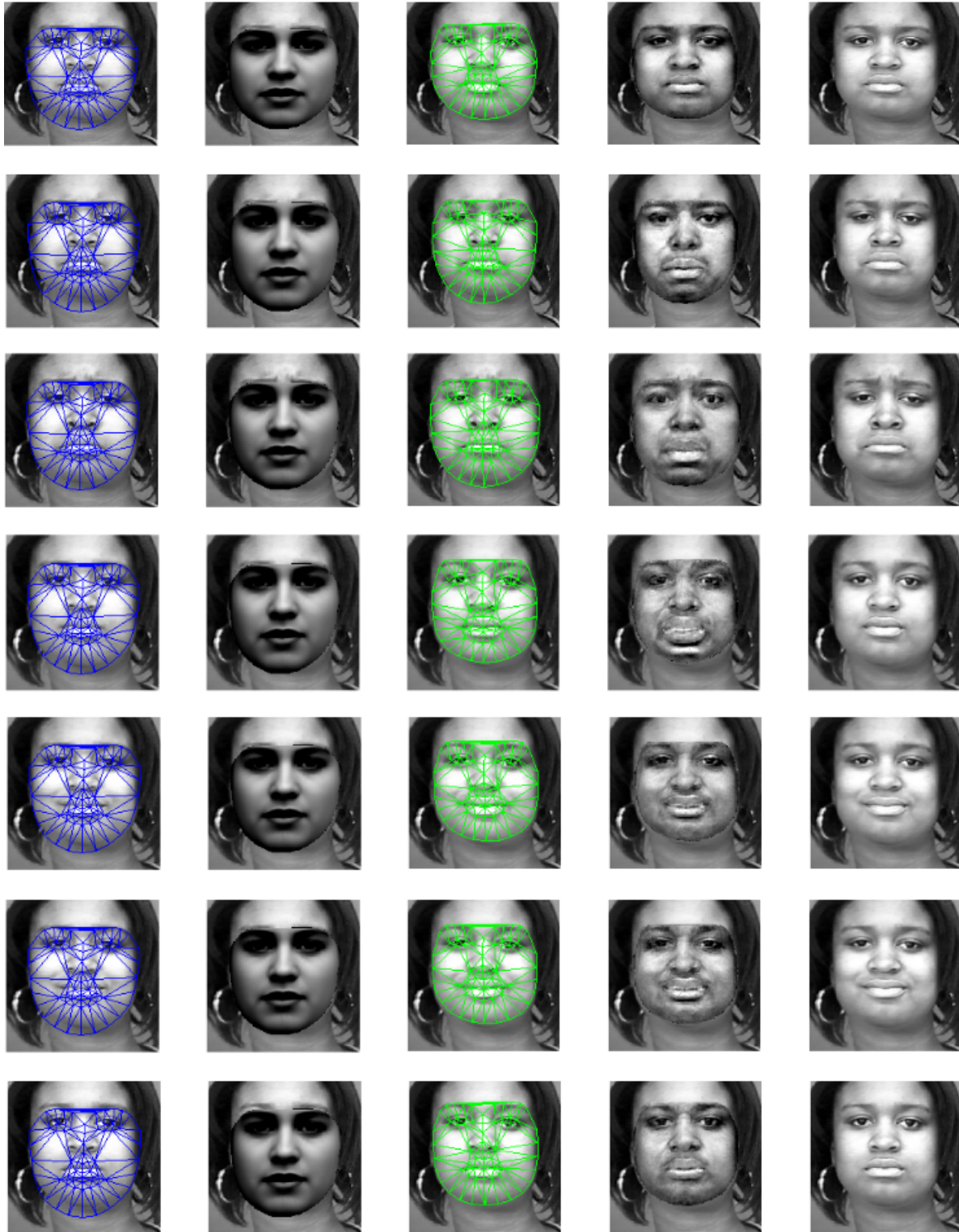


Figura 3.13: Resultados Imágenes 57-63. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1

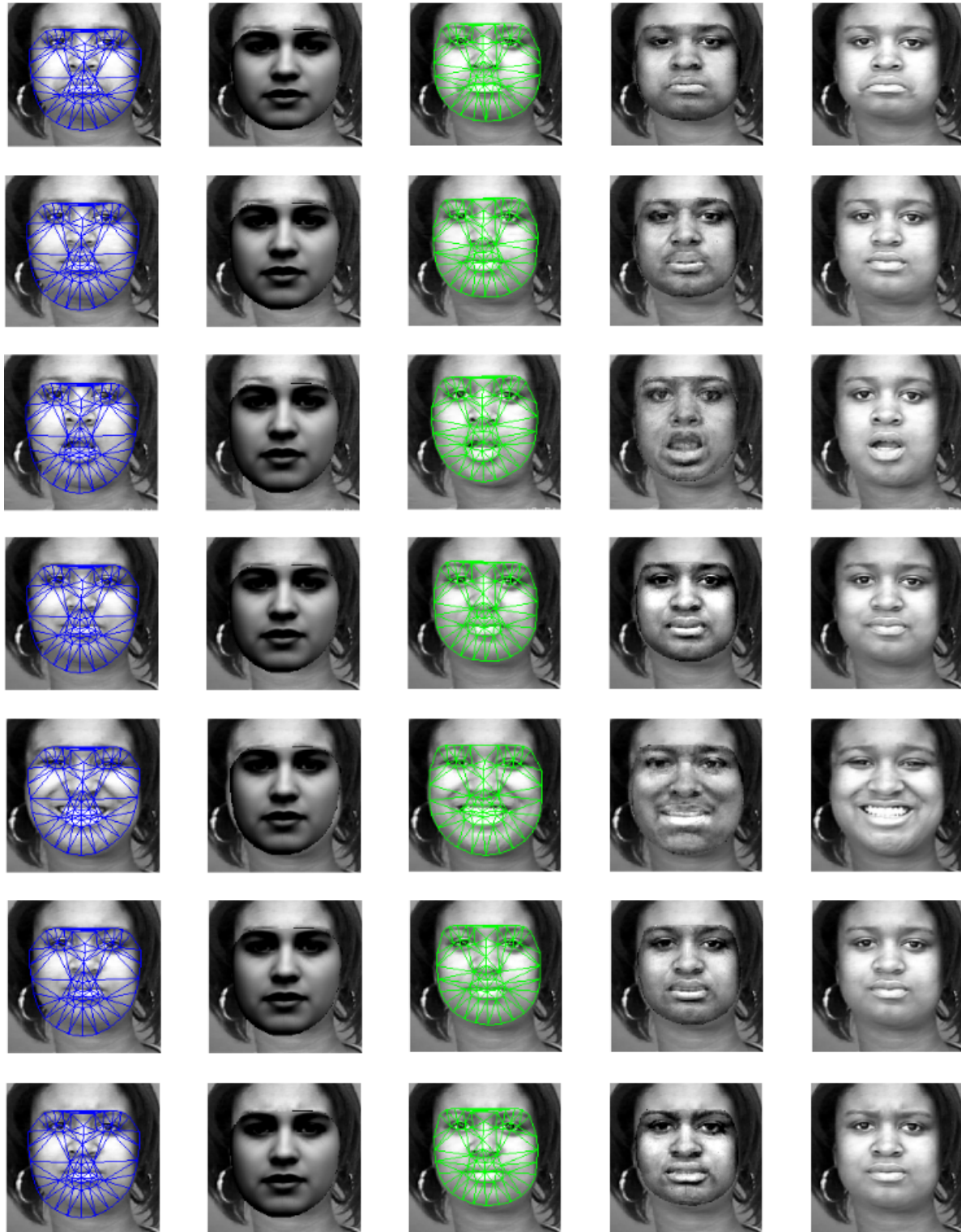


Figura 3.14: Resultados Imágenes 64-70. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1

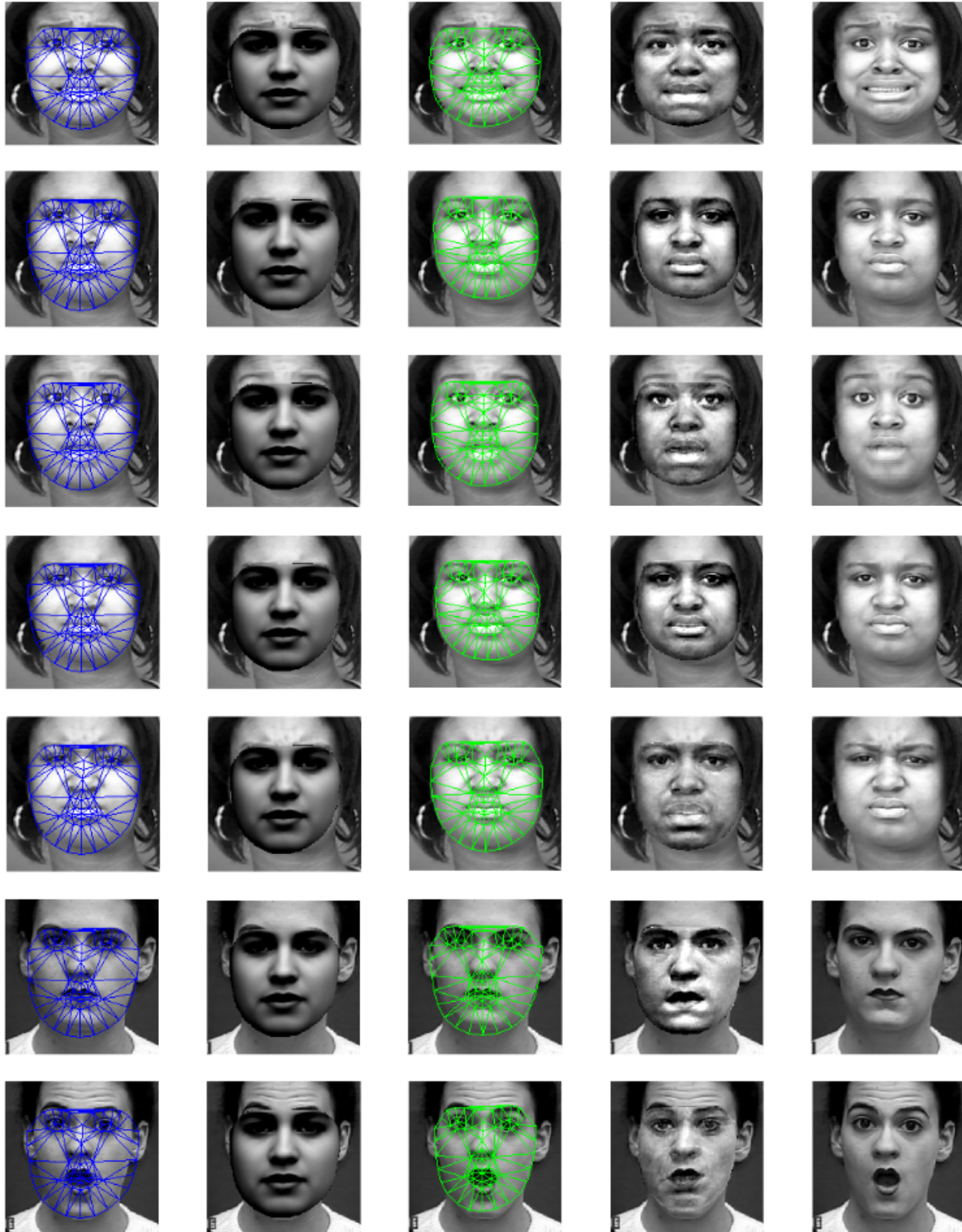


Figura 3.15: Resultados Imágenes 71-77. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1



Figura 3.16: Resultados Imágenes 78-84. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1



Figura 3.17: Resultados Imágenes 85-91. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1



Figura 3.18: Resultados Imágenes 92-98. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1

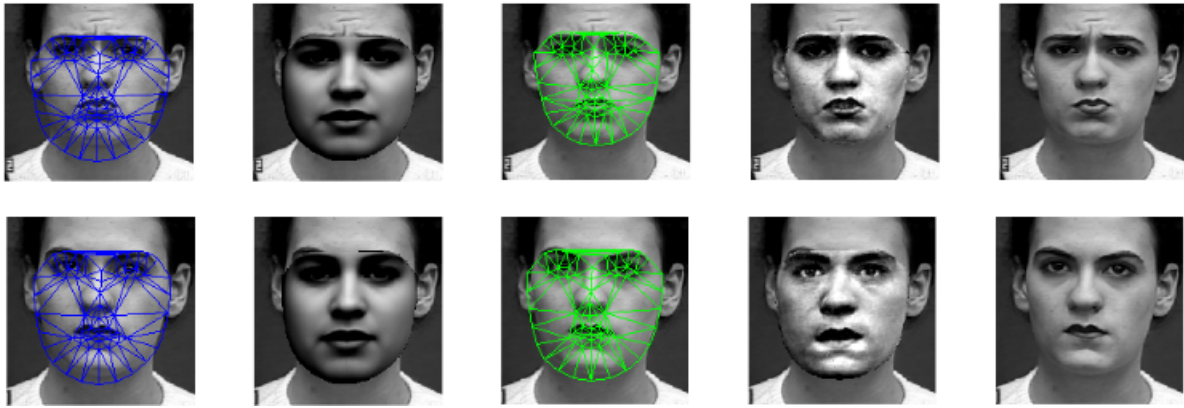
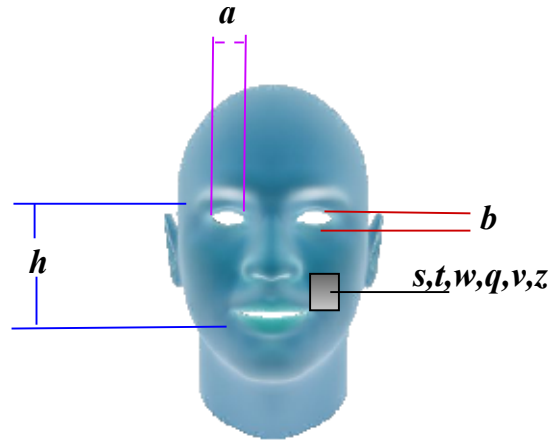


Figura 3.19: Resultados Imágenes 99-100. Para cada renglón, la primera columna muestra el sujeto con la ubicación inicial de la forma media, la segunda columna muestra el *mean shape* en la posición inicial, la tercera columna muestra el enmallado final después del proceso de minimización sobre la imagen objetivo, la cuarta columna muestra la textura final del modelo sobre la imagen y por último la quinta columna muestra la imagen original. Los datos de error final y número de iteraciones asociadas a estas figuras se encuentran consignados en la tabla 3.1

4

Extracción de Descriptores



La representación de patrones en señales e imágenes, busca la descripción de un objeto particular en un entorno donde existe más información que no es de interés [53]. Dicha información adicional no siempre es fácilmente separable del mismo, y por ello la forma en como se representa este objeto debe obedecer a criterios tanto teóricos como prácticos, buscando que cada descriptor posea una gran riqueza de información [54].

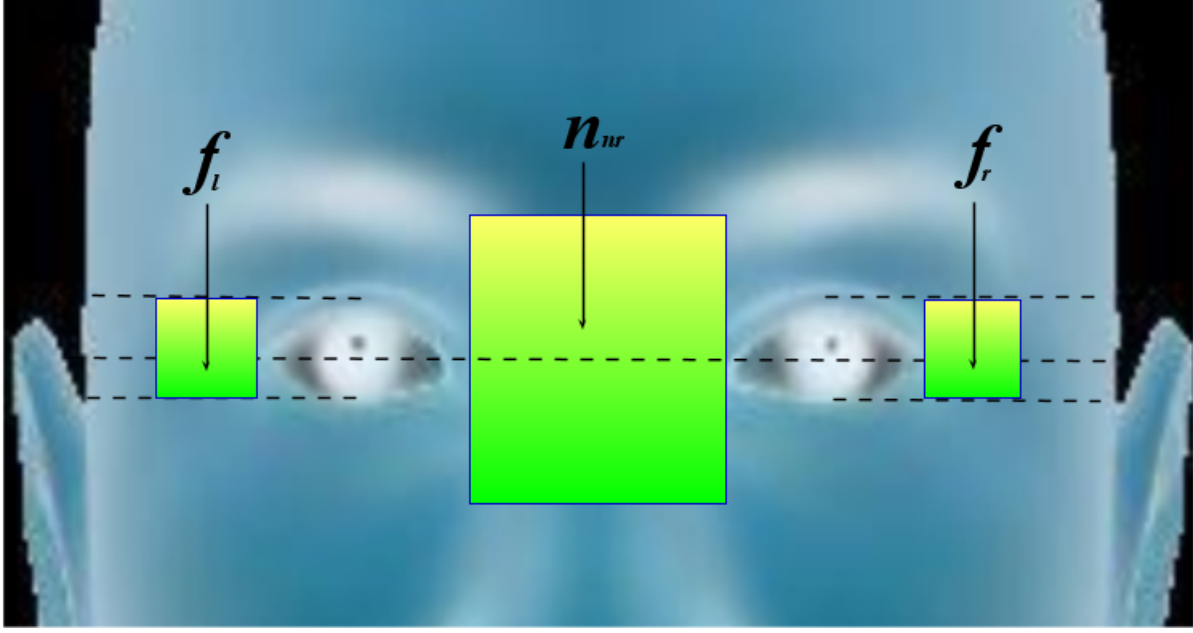
El problema de representación de las expresiones faciales, ha mostrado ser fuente de distintas propuestas en la definición de los descriptores [1]. Sin embargo, existen propuestas que destacan entre las demás por su fundamentación y efectividad en fines de clasificación y síntesis, convirtiendo el problema de selección de los descriptores en una tarea donde se deben valorar la fundamentación y la efectividad.

En este capítulo se realiza la definición de los *features* a emplear para representar las expresiones faciales siguiendo la propuesta realizada por Tian [55], la cual posee sólidos argumentos teóricos y está acompañada por resultados satisfactorios en su utilización. Se definirá además su forma de separación en *variedades* para su posterior procesamiento.

Adicionalmente, se realizará un análisis de los efectos de la simetría en los conjuntos resultantes que describen la expresión, para finalmente dar algunas conclusiones generales al respecto.

4.1. Estado del Arte

En la descripción de comportamientos faciales existe una distinción importante entre el problema de identificación de rostros y el de reconocimiento de expresiones faciales [1] [2], quedando claro

Figura 4.2: Features \mathbf{m}_{FST}

$$\mathbf{m}_{FSF} = \{ \mathbf{v} \in \mathbb{R}^9 \mid \mathbf{v}^T = [D, b_{ri}, b_{li}, h_{r1}, h_{r2}, h_{l1}, h_{l2}, b_{ro}, b_{lo}] \} \quad (4.1)$$

El conjunto que representa los *features* de textura de la región facial superior propuestos por Tian en [55] se representarán como \mathbf{m}_{FST} , las regiones en las cuales se calculan éstos se indican en la figura 4.2. Éste conjunto puede definirse como

$$\mathbf{m}_{FST} = \{ \mathbf{v} \in \mathbb{R}^6 \mid \mathbf{v}^T = [f_{r1}, f_{r2}, f_{l1}, f_{l2}, n_{rs1}, n_{rs2}] \} \quad (4.2)$$

para definir cada uno de los componentes de este vector sean \mathbf{fl} , \mathbf{fr} y \mathbf{n}_{rs} las regiones indicadas en la figura 4.2. Sea ζ_b el operador de detección de bordes cuya salida es una imagen binaria, η_{cc} el operador que indica el número de componentes conexas y σ el operador que indica el valor de la desviación estándar. De esta manera las componentes de los vectores que pertenecen a \mathbf{m}_{FST} se definen como

$$f_{r1} = \eta_{cc}(\zeta_b(\mathbf{fr})), \quad f_{l1} = \eta_{cc}(\zeta_b(\mathbf{fl})), \quad n_{rs1} = \eta_{cc}(\zeta_b(\mathbf{n}_{rs})) \quad (4.3)$$

$$f_{r2} = \sigma(\mathbf{fr}), \quad f_{l2} = \sigma(\mathbf{fl}), \quad n_{rs2} = \sigma(\mathbf{n}_{rs}) \quad (4.4)$$

El conjunto que representa los *features* de forma de la región facial inferior se puede apreciar en las figuras 4.3 y 4.4. Este conjunto de *features* puede definirse como $\mathbf{m}_{FIF} \cup \mathbf{m}_{FIS}$ en donde

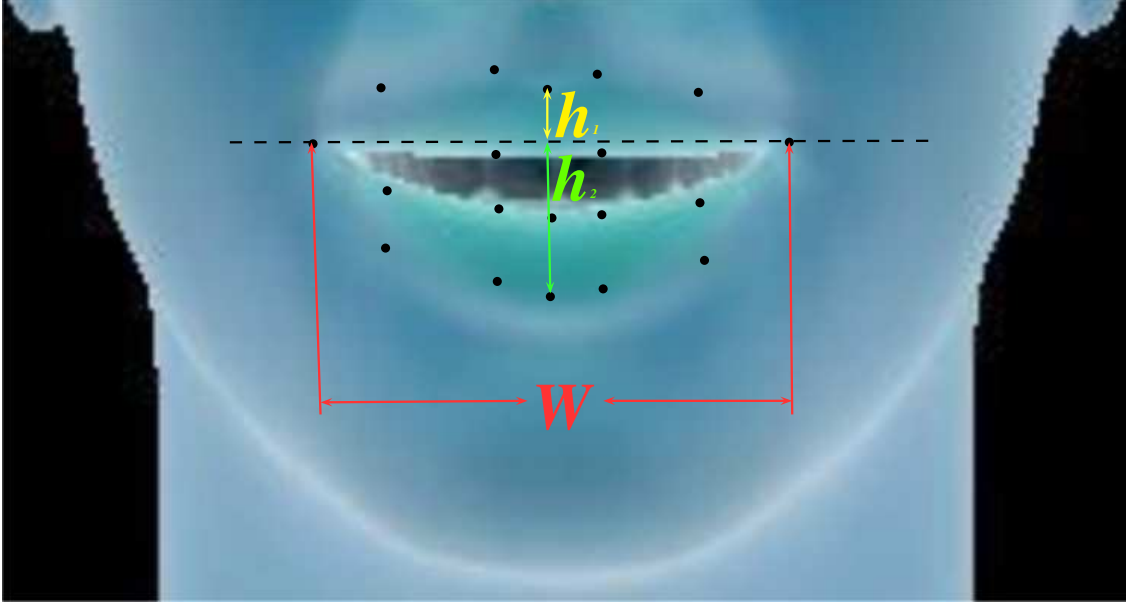
$$\mathbf{m}_{FIF} = \{ \mathbf{v} \in \mathbb{R}^3 \mid \mathbf{v}^T = [h_1, h_2, W] \} \quad (4.5)$$

$$\mathbf{m}_{FIS} = \{ \mathbf{v} \in \mathbb{R}^4 \mid \mathbf{v}^T = [D_r, D_l, d_{sr}, d_{sl}] \} \quad (4.6)$$

en donde \mathbf{m}_{FIS} da una medida de las movimientos simétricos de la región facial inferior.

Por último, para el conjunto de *features* de textura de la región facial inferior se considera en la figura 4.5. Éste se puede definir como

$$\mathbf{m}_{FIT} = \{ \mathbf{v} \in \mathbb{R}^6 \mid \mathbf{v}^T = [T_{ch1}, T_{ch2}, T_{r1}, T_{r2}, T_{l1}, T_{l2}] \} \quad (4.7)$$

Figura 4.3: Features \mathbf{m}_{FIF}

donde $\mathbf{T}_{ch}, \mathbf{T}_r, \mathbf{T}_l$ son las regiones indicadas en la figura 4.5. Empleando los operadores ζ_b, η_{cc} y σ los componentes de los vectores que pertenecen a \mathbf{m}_{FIT} se describen como

$$T_{ch1} = \eta_{cc}(\zeta_b(\mathbf{T}_{ch})), \quad T_{r1} = \eta_{cc}(\zeta_b(\mathbf{T}_r)), \quad T_{l1} = \eta_{cc}(\zeta_b(\mathbf{T}_l)) \quad (4.8)$$

$$T_{ch2} = \sigma(\mathbf{T}_{ch}), \quad T_{r2} = \sigma(\mathbf{T}_r), \quad T_{l2} = \sigma(\mathbf{T}_l) \quad (4.9)$$

4.2.2. Comportamiento Dinámico: Variedades de Datos

Si bien los conjuntos de *features* representan características esenciales de las expresiones faciales en cada frame, su representación debe realizarse respecto de un punto inicial si se desea conocer el comportamiento dinámico. Sea $\mathbf{v}_{(t)}$ el vector de *features* asociado al frame t de una secuencia de vídeo de imágenes faciales de un determinado sujeto. Los conjuntos de datos que describen el comportamiento dinámico, para un vídeo con un total de $q + 1$ frames se representan como

$$\widehat{\mathcal{M}}_{FSF} = \{\mathbf{u} \in \mathbb{R}^9 \mid \mathbf{u} = \mathbf{v}_{(t)} - \mathbf{v}_{(0)}, \quad \mathbf{v}_{(t)}, \mathbf{v}_{(0)} \in \mathbf{m}_{FSF}, \quad \wedge \quad 0 \leq t \leq q\} \quad (4.10)$$

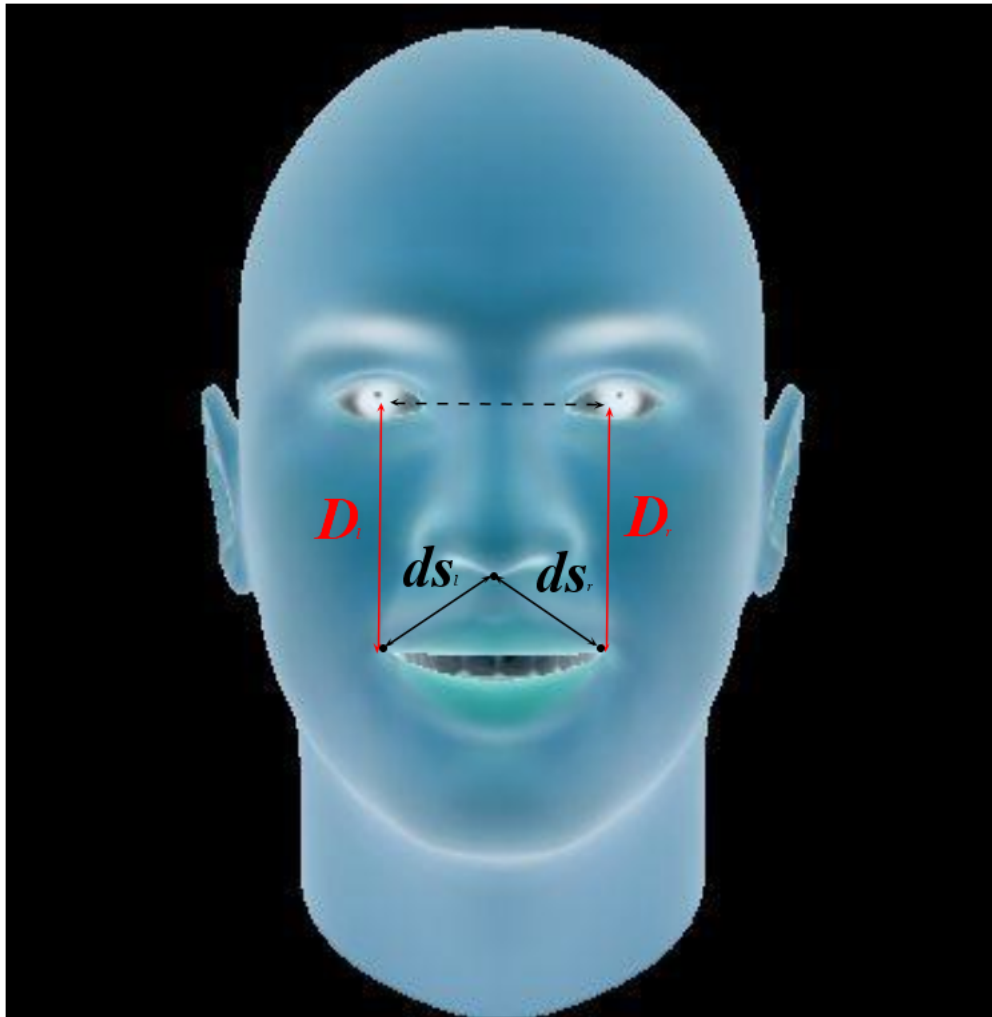
$$\widehat{\mathcal{M}}_{FST} = \{\mathbf{u} \in \mathbb{R}^6 \mid \mathbf{u} = \mathbf{v}_{(t)} - \mathbf{v}_{(0)}, \quad \mathbf{v}_{(t)}, \mathbf{v}_{(0)} \in \mathbf{m}_{FST}, \quad \wedge \quad 0 \leq t \leq q\} \quad (4.11)$$

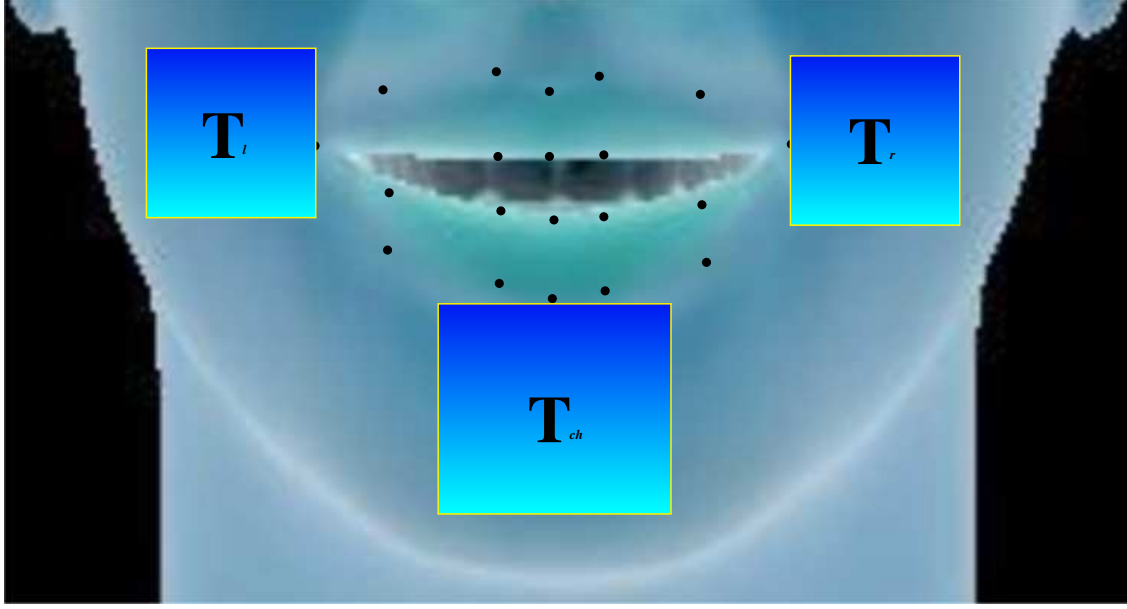
$$\widehat{\mathcal{M}}_{FIF} = \{\mathbf{u} \in \mathbb{R}^3 \mid \mathbf{u} = \mathbf{v}_{(t)} - \mathbf{v}_{(0)}, \quad \mathbf{v}_{(t)}, \mathbf{v}_{(0)} \in \mathbf{m}_{FIF}, \quad \wedge \quad 0 \leq t \leq q\} \quad (4.12)$$

$$\widehat{\mathcal{M}}_{FIS} = \{\mathbf{u} \in \mathbb{R}^4 \mid \mathbf{u} = \mathbf{v}_{(t)} - \mathbf{v}_{(0)}, \quad \mathbf{v}_{(t)}, \mathbf{v}_{(0)} \in \mathbf{m}_{FIS}, \quad \wedge \quad 0 \leq t \leq q\} \quad (4.13)$$

$$\widehat{\mathcal{M}}_{FIT} = \{\mathbf{u} \in \mathbb{R}^6 \mid \mathbf{u} = \mathbf{v}_{(t)} - \mathbf{v}_{(0)}, \quad \mathbf{v}_{(t)}, \mathbf{v}_{(0)} \in \mathbf{m}_{FIT}, \quad \wedge \quad 0 \leq t \leq q\} \quad (4.14)$$

de esta manera se podrá medir la evolución de la expresión respecto de las características encontradas en el primer *frame*.

Figura 4.4: Features \mathbf{m}_{FIS}

Figura 4.5: Features \mathbf{m}_{FIT}

Comportamiento Dinámico: Comparaciones de Magnitud y Separación en Variedades

En el trabajo de Tian [55] se propuso realizar la clasificación de expresiones faciales empleando dos clasificadores diferentes para las regiones faciales inferior y superior respectivamente. Esta misma idea, se retoma en este trabajo, y se propone realizar el análisis de la información para su reducción de dimensionalidad individualmente en $\widehat{\mathcal{M}}_{FSF}$, $\widehat{\mathcal{M}}_{FST}$, $\widehat{\mathcal{M}}_{FIF}$, $\widehat{\mathcal{M}}_{FIS}$, $\widehat{\mathcal{M}}_{FIT}$. Esto permitirá analizar datos que son *comparables* entre sí en órdenes de magnitud, además de eliminar el problema de hallar escalamientos para lidiar con los cambios de unidades.

En el capítulo 5 se adicionarán argumentos relacionados con efectos de las grandes dimensiones sobre la representación de los datos. Ésto reforzará aún más la idea de analizar los datos de acuerdo a la propuesta realizada en este capítulo.

Comportamiento Dinámico: Acerca de la Simetría

La mayor parte de los *features* extraídos representan información redundante en caso de que exista simetría en los movimientos faciales. En muchas situaciones, este es el caso. Por ello es importante resaltar, que para fines de procesamiento de la información es necesario tener esto presente, pues la dimensión estimada para la variedad de los movimientos no simétricos podría ser mayor que la dimensión correspondiente a una variedad de movimientos simétricos. Para ver esto considere un elemento \mathbf{v} de \mathbf{m}_{FSF}

$$\mathbf{v} = [D, b_{ri}, b_{li}, h_{r1}, h_{r2}, h_{l1}, h_{l2}, b_{ro}, b_{lo}] \quad (4.15)$$

en caso de que existiesen movimientos simétricos, la siguiente aproximación sería totalmente válida

$$[D, b_{ri}, b_{li}, h_{r1}, h_{r2}, h_{l1}, h_{l2}, b_{ro}, b_{lo}] \approx [D, b_{ri}, b_{ri}, h_{r1}, h_{r2}, h_{r1}, h_{r2}, b_{ro}, b_{ro}] \quad (4.16)$$

es decir ya no serían nueve las variables que describen el vector de representación sino cinco. Esto bien podría tener una implicación fuerte en la estimación de la dimensión intrínseca asociada a cada conjunto. Más aún cuando las simetrías, en caso de que existan, solo se registren para algunos *features*.

Este argumento, brinda un soporte para el esquema de reducción de dimensionalidad propuesto

en el capítulo 5, en la cual no se generaliza la dimensión estimada de la variedad. Por el contrario se considera como una característica particular asociada a cada variedad de *features* correspondientes a un determinado sujeto.

4.3. Implementación

La implementación del código para la extracción de características se realizó en la herramienta de simulación Matlab. La detección de bordes se realizó empleando los operadores de Sobel.

4.4. Discusión y Conclusiones

El esquema seleccionado basado en la propuesta de Tian [55] busca recoger los beneficios obtenidos con esta selección, pero además de ello se adiciona la extracción de los *features* \mathbf{m}_{FIS} que brindan una medida de los movimientos asimétricos de la región facial inferior. Con esto se espera añadir información que brinde la posibilidad de realizar discriminación entre expresiones faciales.

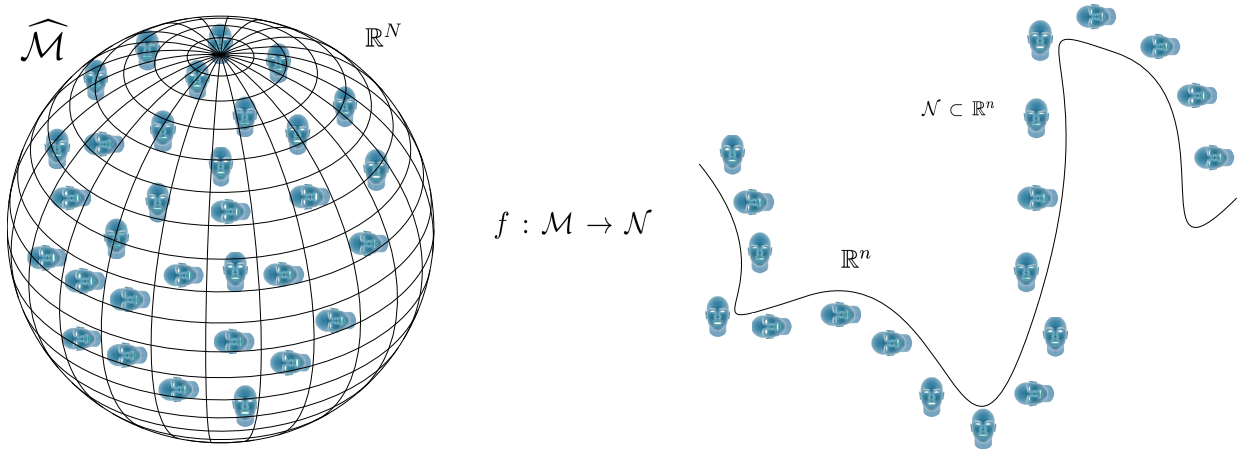
Si se optara por representar la variedad descriptora por medio de los niveles de gris de la zona facial extraída, se tendría dicha variedad encajada o incrustada en \mathbb{R}^{RS} siendo $R \times S$ el tamaño de la imagen. El esquema propuesto para la extracción y representación de los *features*, supera enormemente esta primera opción de representación pues las variedades están encajadas en espacios euclidianos de dimensiones fijas, y no se ven afectadas por los efectos nocivos del aumento de la dimensionalidad a medida que $RS \rightarrow \infty$.

Es importante considerar que los conjuntos de datos $\widehat{\mathcal{M}}_{FSF}, \widehat{\mathcal{M}}_{FST}, \widehat{\mathcal{M}}_{FIF}, \widehat{\mathcal{M}}_{FIS}, \widehat{\mathcal{M}}_{FIT}$ pueden ser combinados de manera arbitraria, sin embargo la estimación de su dimensión intrínseca puede variar considerablemente, y más aún debido a las condiciones de simetría que pueden no ser fijas dentro de un conjunto arbitrario de secuencias de expresiones. Por esto en capítulos posteriores se realiza un procesamiento individual de cada uno de estos conjuntos sin dar una generalización de la dimensión de la variedad sobre la cual recaen los datos.

La separación de los datos en variedades diferentes brinda un análisis más amplio y exhaustivo de las regiones facial superior e inferior, pues Lee [58] prueba que los métodos de reducción de dimensionalidad, que se discutirán en detalle en el capítulo 5, pueden usarse con total confianza para datos que se encuentren encajados en espacios de dimensión N tal que $N \leq 5$. En el esquema de extracción de *features* que se planteó en este capítulo, se garantiza en lo que se refiere a la dimensión del espacio de representación original que se tiene un grado total de confianza para el procesamiento de las variedades $\widehat{\mathcal{M}}_{FIF}, \widehat{\mathcal{M}}_{FIS}$ que están encajadas en espacios de dimensiones 3 y 4 respectivamente, mientras que para las variedades $\widehat{\mathcal{M}}_{FST}, \widehat{\mathcal{M}}_{FIT}$ el grado de confianza es alto pues están en espacios de dimensión 6 que es apenas una unidad mayor al grado de confidencialidad. Para el procesamiento de la variedad $\widehat{\mathcal{M}}_{FSF}$ el grado de confianza aún puede considerarse cercano al valor óptimo. Así, se confirma aún más la ventaja de esta separación de la información en comparación con el uso directo de la información de textura, pues en este último enfoque la dimensión RS del espacio de representación sería $RS \gg 5$ y por lo tanto la confianza sería muy baja.

5

Reducción de Dimensionalidad



El tratamiento de la información extraída de las expresiones faciales representa la última etapa previa a su uso en clasificación y/o síntesis. Ésta es de esencial cuidado, no solo porque es necesario develar en lo posible la estructura que forman los datos que representan los patrones, sino porque además los resultados obtenidos por varias de las distintas técnicas existentes pueden ser muy diferentes entre sí.

En este capítulo se hace una presentación y un análisis de las técnicas más consolidadas y eficaces en la reducción de dimensionalidad, teniendo en cuenta el valioso estudio realizado por Lee en [58], las demás referencias analizadas de acuerdo al estado del arte y por supuesto las características de la información extraída junto con el efecto de estas técnicas sobre la misma.

En primera instancia se presentarán las motivaciones tanto teóricas como prácticas para realizar la reducción de dimensionalidad. Posteriormente se hará énfasis en los aspectos a explorar en la aplicación de las distintas técnicas de reducción de dimensionalidad para los propósitos del tratamiento de la información en este trabajo, y en seguida se hará una breve discusión del estado del arte sobre reducción de dimensionalidad.

En la sección 5.6 se hará una presentación breve de las diferentes definiciones formales de dimensión y los esquemas existentes para realizar su estimación práctica, en donde adicionalmente se hará la inclusión de una nueva propuesta para realizar dicha estimación basada en algunos resultados sobre variedades diferenciales. En la sección 5.7 se presentarán las técnicas de reducción de dimensionalidad basadas en preservación de la distancia realizando un análisis detallado de cada una y destacando sus respectivos pros y contras para su aplicación en este trabajo. En la sección 5.8 se realizará este mismo procedimiento con las técnicas basadas en preservación de la topología.

Finalmente se presentará una sección en la cual se discuten algunos aspectos de comparación entre las técnicas que no son tratados en la presentación teórica de cada una, y que siguen princi-

palmente los experimentos realizados por Lee en [58].

Se presentará además una sección en donde se remarcarán las conclusiones principales sobre estos métodos, y así concluir con el planteamiento de un esquema para la reducción dimensionalidad.

La discusión de cada una de las técnicas tanto de estimación de la dimensión como de reducción, se acompañará de su aplicación al conjunto de datos $\widehat{\mathcal{M}}_{FSF}$ empleando la información para cada expresión y la información total correspondiente al sujeto S055 de la base de datos Cohn-Kanade [7]. Al usar $\widehat{\mathcal{M}}_{FSF}$ se verifica el comportamiento de los métodos empleados con el conjunto de datos con mayores dimensiones y es posible tener en cuenta esto para tomar una decisión en la selección de los mismos.

5.1. Hipótesis Central

En general, y en la mayor parte de las aplicaciones cuando se extraen vectores de descripción de un determinado patrón, se usan valores reales, de manera que los vectores de descripción pertenecen al espacio euclidiano \mathbb{R}^N siendo N la dimensión de los vectores que representan dicho patrón [58] [53].

Todas las técnicas de reducción de dimensionalidad no lineal (NLDR) se basan en la hipótesis de que los datos a los cuales se les quiere realizar reducción de dimensiones caen o hacen parte de una *subvariedad topológica* de \mathbb{R}^N [58] [59] [60]. En particular para algunos métodos es conveniente restringir esta hipótesis a *subvariedades diferenciales* y/o *subvariedades Riemannianas* de \mathbb{R}^N [61] [62]. En este trabajo se consideraran vectores de descripción con componentes reales, de manera que se asume esta misma hipótesis analizando la diferenciabilidad y la existencia de un producto interno asociado a la variedad cuando se considera apropiado, dependiendo del método usado para reducir la dimensionalidad.

Se representará con $\widehat{\mathcal{M}}$ al conjunto de datos patrón que caen sobre la variedad \mathcal{M} , es decir los datos se representarán con un gorro sobre la letra o símbolo que representa la variedad analítica de la que según la hipótesis general, hacen parte los datos.

5.2. Motivaciones Prácticas

Lee en [58] plantea como una de las principales motivaciones prácticas para la reducción de la dimensionalidad, la eliminación de la redundancia en la información, de tal manera que se tenga el mínimo número de variables que verdaderamente describa el problema. Pero además de ello, es inmediata la utilidad en el entendimiento, clasificación y procesamiento de la información así como también las inferencias y generalizaciones que se puedan lograr por medio de este entendimiento [58] [63] [64].

5.3. Motivaciones Teóricas

Parte de las motivaciones teóricas recaen en la visualización de los datos que representan un patrón objetivo. En el caso de representaciones espaciales es bien conocido el problema de visualización de datos con más de tres dimensiones, y así mismo la visualización de datos que además de las dimensiones espaciales cuentan con variaciones temporales [58]. Sin embargo los efectos más crudos y significativos de usar datos en espacios de grandes dimensiones son conocidos como *The Curse of Dimensionality* y hacen referencia al fenómeno de espacio vacío y a la concentración de normas y distancias [58] [63] [65].

El fenómeno de espacio vacío está asociado en parte a la pérdida de propiedades geométricas y topológicas de un objeto a medida que el espacio de representación incrementa sus dimensiones. Lee [58], Francois [63] y Demartines [65] muestran independientemente esto analizando inicialmente el volumen de objetos geométricos simples como una esfera en N dimensiones así: sea V_s el volumen

de una esfera de radio r en el espacio euclidiano \mathbb{R}^N

$$V_s = \frac{\pi^{\frac{N}{2}} r^N}{\Gamma(1 + \frac{N}{2})} \quad (5.1)$$

si se considera $r = 1$, es decir una hiper-esfera de radio 1, se tiene que

$$V_s = \frac{\pi^{\frac{N}{2}}}{\Gamma(1 + \frac{N}{2})} \implies \lim_{N \rightarrow \infty} V_s = 0 \quad (5.2)$$

es decir a medida que aumentan las dimensiones del espacio de representación el volumen de la esfera se reduce, y un aspecto importante para resaltar es que si se tratase de una representación numérica de dicho volumen, eventualmente habría un número finito de dimensiones a partir del cual el volumen calculado sería nulo.

Usando estos mismos argumentos Lee [58], Francois [63] y Demartines [65] muestran que los contornos equiprobables de una distribución gaussiana en N dimensiones crecen en términos de la desviación estándar a medida que las dimensiones se aumentan. Este mismo análisis puede realizarse para otras representaciones geométricas.

Estos fenómenos permiten hacer alusión a un problema directamente involucrado en los procesos de clasificación y de identificación de patrones: *la pérdida de discriminación de las normas en grandes dimensiones*, es decir la distribución de normas en una distribución de puntos dada tiende a concentrarse, esto es conocido como el *fenómeno de concentración* [58] [63] [65]. Éste puede describirse por medio del siguiente teorema, presentado en [58] [65]

Teorema 1 Sea \mathbf{y} un vector N -dimensional $[y_1, \dots, y_d, \dots, y_N]^T$; todas las componentes y_d del vector son independientes e idénticamente distribuidas (iid), con momento de orden ocho finito. Entonces la media $\mu_{\|\mathbf{y}\|}$ y la varianza $\sigma_{\|\mathbf{y}\|}^2$ de la norma euclidiana son

$$\mu_{\|\mathbf{y}\|} = E(\|\mathbf{y}\|) = \sqrt{aN - b} + \mathcal{O}(N^{-1}) \quad (5.3)$$

$$\sigma_{\|\mathbf{y}\|}^2 = \text{Var}(\|\mathbf{y}\|) = b + \mathcal{O}(N^{-\frac{1}{2}}) \quad (5.4)$$

donde a y b son parámetros dependientes únicamente de los momentos centrales de orden 1, 2, 3 y 4 de los y_i :

$$a = \mu^2 + \sigma^2 \quad (5.5)$$

$$b = \frac{4\mu^2\sigma^2 - \sigma^4 + 4\mu\mu_3 + \mu_4}{4(\mu^2 + \sigma^2)} \quad (5.6)$$

donde μ_r es el momento central de orden r : $\mu_r = E[(y_k - \mu)^r]$. μ es la media $E(y_k)$, y σ^2 la varianza $\text{Var}(y_k)$

con este Teorema se puede garantizar que mientras la media aumenta proporcionalmente a \sqrt{N} la varianza permanece relativamente constante, numéricamente esto indicaría que a partir de un valor N que crece se llega a un punto donde no es posible distinguir entre $\mu_{\|\mathbf{y}\|}$ y $\|\mathbf{y}\|$. Por otra parte usando la desigualdad de chebyshev se tiene que

$$P(\|\mathbf{y}\| - \mu_{\|\mathbf{y}\|} \geq \varepsilon) \leq \frac{\sigma_{\|\mathbf{y}\|}^2}{\varepsilon^2} \quad (5.7)$$

es decir la probabilidad de que la norma de los vectores esté por fuera de un intervalo de longitud fija es constante. Lo cual confirma este hecho.

5.4. Aspectos a Explorar

En la selección de un método para la reducción de la dimensionalidad, es necesario tener en cuenta características como, la densidad de los datos, la cantidad de los mismos y las características de algunas de las matrices resultantes en los procesos de almacenamiento y procesamiento. Otro aspecto de igual importancia es el concerniente a la selección de los parámetros empelados por una determinada técnica de acuerdo a las características de los datos, ya que si bien algunas técnicas de reducción y estimación de la dimensión pueden dar resultados satisfactorios para ciertos parámetros, la selección de los mismos puede ser subjetiva y/o por sintonización, aspecto esencial a considerar si se desea un procesamiento automático de la información extraída de las expresiones faciales.

Paralelo a estos argumentos se considerará la fundamentación teórica empleada en el desarrollo de los métodos ya que ésta puede jugar un papel esencial para establecer conexiones y mejoras entre éstos, además de brindar un panorama de exploración para posibles trabajos futuros asociados a estas temáticas y al análisis de expresiones faciales.

5.5. Estado del Arte

El problema de reducción de dimensionalidad y estimación de la dimensión intrínseca de un conjunto de datos ha despertado interés en diversas áreas incluso décadas atrás del comienzo del *boom* del manejo de la información [58]. David Donoho de Stanford University presentó en [64] el problema de manejar y desentrañar la naturaleza de datos representados en espacios de grandes dimensiones como uno de los desafíos icónicos de este siglo.

Los desarrollos en estas temáticas son variados, con diferentes grados de fundamentación y motivados por aplicaciones en distintos campos de la ciencia. Lee desarrolla en [58] una presentación muy completa y detallada del estado del arte de las principales técnicas asociadas al problema de la reducción de la dimensionalidad y así mismo al de estimación de la dimensión intrínseca. En esta investigación se sigue la presentación de este estado del arte complementándolo con técnicas que no son abordadas por Lee [58] pero que han mostrado su efectividad en el análisis de expresiones faciales. De esta manera la presentación de cada una de las técnicas en secciones subsecuentes representa para esta investigación el estado del arte de técnicas de reducción de dimensionalidad y estimación de la dimensión intrínseca más pertinentes, relevantes y efectivas.

5.6. Dimensión Intrínseca

La hipótesis de que los datos que representan determinado conjunto de patrones caen o hacen parte de una variedad topológica, diferenciable y/o de Riemann, subyacen bajo la idea intuitiva de que estos conjuntos de datos hacen parte de una estructura que es localmente similar a \mathbb{R}^n [66] [67]. Por medio de la estimación de la dimensión intrínseca de dicha estructura se pretende determinar este n .

Matemáticamente hay definiciones formales y precisas de dimensión para conjuntos que forman distintas estructuras, pero difíciles de calcular o aproximar en situaciones prácticas, como es el caso de la dimensión de recubrimiento de Lebesgue o la dimensión de Hausdorff [68]. Sin embargo en un intento por estimar los valores de estas dimensiones se han desarrollado otras medidas de la dimensión como la dimensión de correlación, que si bien su definición estricta es aún teórica, su forma ha permitido adaptarla para realizar cálculos en situaciones prácticas con mucha frecuencia [58] [68] [69].

A continuación se presentan las distintas definiciones de las diferentes dimensiones, se establecerán algunas relaciones importantes entre las mismas y finalmente se planteará un esquema para la estimación práctica de dicha dimensión teniendo en cuenta la información presentada.

Considerando los argumentos teóricos y resultados experimentales, se planteará una nueva alternativa para la estimación de la dimensión que se usará en conjunto con otras formulaciones, para estimar la dimensión de las distintas variedades que forman los conjuntos de expresiones faciales a partir de los *features* descritos en el capítulo 4.

5.6.1. Dimensión de Recubrimiento

Los resultados generales sobre la dimensión de recubrimiento están dados sobre espacios topológicos [70]. Siendo una variedad topológica un espacio topológico, se plantea en primera instancia una definición usando estos conceptos.

El recubrimiento de un espacio topológico \mathcal{M} es una familia $\mathcal{A} = \{A_\lambda\}_{\lambda \in \Lambda}$ de subconjuntos tales que

$$\bigcup_{\lambda \in \Lambda} A_\lambda = \mathcal{M} \quad (5.8)$$

si cada elemento de \mathcal{A} es abierto entonces \mathcal{A} es llamada cubierta abierta de \mathcal{M} , en caso de que todos los elementos de \mathcal{A} sean cerrados entonces será llamada una cubierta cerrada de \mathcal{M} . Una cubierta $\{B_\gamma\}_{\gamma \in \Gamma}$ se dice que es un refinamiento de la cubierta $\{A_\lambda\}_{\lambda \in \Lambda}$ si para cada $\gamma \in \Gamma$ existe algún $\lambda \in \Lambda$ tal que $B_\gamma \subset A_\lambda$ [70].

El orden de una familia de subconjuntos $\{A_\lambda\}_{\lambda \in \Lambda}$, no todos vacíos, de algún conjunto es el entero más grande para el cual existe un subconjunto P de Λ con $n + 1$ elementos tales que $\bigcap_{\lambda \in P} A_\lambda$ es no vacío, o es ∞ si no existe tal entero [70].

Con las anteriores definiciones se puede dar la definición de dimensión de recubrimiento en términos de espacios topológicos de la siguiente manera, siguiendo a Pears [70]

Definición 1 *La dimensión de recubrimiento $\dim(\mathcal{M})$ de un espacio topológico \mathcal{M} es el mínimo entero n tal que cada cubierta de \mathcal{M} tiene un refinamiento abierto de un orden que no excede a n o es ∞ si no existe tal entero.*

Para los propósitos de este trabajo, se considerará la hipótesis adicional sobre las variedades topológicas a las cuales pertenecen los patrones extraídos, de ser *metrizables*; es decir que van a ser variedades homeomorfas a un espacio métrico, y la métrica allí definida va a inducir una topología equivalente a la asignada a esta variedad topológica [71], o dicho de otra manera su topología es inducida por una métrica [70]. Lee demuestra en [71] que toda variedad diferenciable es metrizable y en particular si la variedad es Riemanniana es posible definir esta métrica en términos de *geodésicas*, así que la consideración adicional se enfoca en particular en el tratamiento de variedades topológicas no diferenciales sobre las cuales pueden recaer ciertos conjuntos de patrones. Cuando se discutan los métodos existentes para la reducción de dimensionalidad se verá que esta hipótesis es muy adecuada puesto que varias técnicas se basan en aproximaciones de distancias medidas sobre la variedad incrustada en el espacio original de grandes dimensiones.

La definición dada por Pears [70] se puede particularizar a espacios métricos de la siguiente manera acorde a Robinson [72]. Sea (X, ρ) un espacio métrico, y sea \mathcal{M} un subconjunto de X . Una cubierta de $\mathcal{M} \subseteq X$ es una colección finita $\{U_j\}_{j=1}^r$ de subconjuntos abiertos de X tales que

$$\mathcal{M} \subseteq \bigcup_{j=1}^r U_j \quad (5.9)$$

el orden de una cubierta es el entero más grande n tal que existen $n + 1$ miembros de la cubierta que tienen intersección no vacía. Un a cubierta β es un refinamiento de una cubierta α si cada miembro de β está contenido en algún miembro de α [72]. Con esto la definición se particulariza de la siguiente manera

Definición 2 *Un subconjunto $\mathcal{M} \subseteq X$ tiene dimensión $\dim(\mathcal{M}) \leq n$ si cada cubierta tiene un refinamiento de orden $\leq n$. Un conjunto \mathcal{M} tiene dimensión $\dim(\mathcal{M}) = n$ si $\dim(\mathcal{M}) \leq n$ pero no es cierto que $\dim(\mathcal{M}) \leq n - 1$.*

La dimensión de recubrimiento de un conjunto \mathcal{M} en adelante se denotará como $\dim(\mathcal{M})$, ésta es un invariante topológico, es decir es invariante a los homeomorfismos.

5.6.2. Dimensión de Hausdorff

La presentación de la dimensión de Hausdorff se hará siguiendo a Falconer [73] [68] y Robinson [72], quien presenta varios desarrollos teóricos sobre esta dimensión trabajando en \mathbb{R}^n . Esto es bastante adecuado para este trabajo pues las variedades asociadas a los distintos conjuntos de patrones están incrustadas (*embedding*) en \mathbb{R}^N siendo N el número de componentes de los vectores descriptores del patrón. Al igual que en las secciones anteriores esta definición se asocia directamente a espacios métricos.

Si U es un subconjunto no vacío de \mathbb{R}^n se define el diámetro de U como $|U| = \sup \{|x - y| : x, y \in U\}$. Si $\mathcal{M} \subset \bigcup_i U_i$ y $0 < |U_i| \leq \delta$ para cada i , se dice que $\{U_i\}$ es una δ -cubierta de \mathcal{M} . Así se define

$$\mathcal{H}_\delta^s(\mathcal{M}) = \inf \left\{ \sum_{i=1}^{\infty} |U_i|^s \right\} \quad (5.10)$$

donde el ínfimo es tomado sobre todas las δ -cubiertas contables $\{U_i\}$ de \mathcal{M} . De esta manera la medida externa de Hausdorff s -dimensional de \mathcal{M} se obtiene haciendo $\delta \rightarrow 0$ como

$$\mathcal{H}^s(\mathcal{M}) = \lim_{\delta \rightarrow 0} \mathcal{H}_\delta^s(\mathcal{M}) \quad (5.11)$$

este límite existe, pero puede ser infinito [72]. La dimensión de Hausdorff $\dim_H(\mathcal{M})$ de un conjunto \mathcal{M} es el valor único tal que

$$\mathcal{H}^s(\mathcal{M}) = \infty \quad \text{si} \quad 0 \leq s < \dim_H(\mathcal{M}), \quad \mathcal{H}^s(\mathcal{M}) = 0 \quad \text{si} \quad \dim_H(\mathcal{M}) < s < \infty \quad (5.12)$$

en particular si $\mathcal{M} \subseteq (X, \rho)$, la dimensión de Hausdorff de \mathcal{M} es

$$\dim_H(\mathcal{M}) = \inf \left\{ d \geq 0 : \mathcal{H}^d(\mathcal{M}) > 0 \right\} \quad (5.13)$$

siendo (X, ρ) el espacio métrico en el cual está incrustada la variedad.

5.6.3. Dimensión de Capacidad (*Box-Counting Dimension*)

Sea $N(\mathcal{M}, \epsilon)$ que representa el mínimo número de bolas de radio ϵ con centros en \mathcal{M} requerida para cubrir \mathcal{M} , la *box-counting dimension* de \mathcal{M} se define como [72] [73] [68]

$$\dim_{box}(\mathcal{M}) = \lim_{\epsilon \rightarrow 0} \frac{\log N(\mathcal{M}, \epsilon)}{-\log \epsilon} \quad (5.14)$$

5.6.4. Dimensión de Información

Lee presenta en [58] la dimensión de información como un caso particular de una dimensión más general conocida como *q-dimension*, que se expresa en términos de

$$C_q(\mu, \epsilon) = \int (\mu(B_\epsilon(\mathbf{y})))^{q-1} d\mu(\mathbf{y}) \quad (5.15)$$

siendo μ una medida de probabilidad de Borel sobre un espacio métrico \mathcal{M} tal que $\mathbf{y} \in \mathcal{M}$. $q \geq 0$, $\epsilon > 0$ y $B_\epsilon(\mathbf{y})$ es una bola cerrada de radio ϵ . De manera que para todo $q \geq 0$, $q \neq 1$ se definen las siguientes cantidades

$$D_q^-(\mu) = \lim_{\epsilon \rightarrow 0} \inf \frac{\log C_q(\mu, \epsilon)}{(q-1) \log \epsilon} \quad (5.16)$$

$$D_q^+(\mu) = \limsup_{\epsilon \rightarrow 0} \frac{\log C_q(\mu, \epsilon)}{(1-q) \log \epsilon} \quad (5.17)$$

Si $D_q^+ = D_q^-$, se dice que este valor común, representado como D_q es la q -dimensión de \mathcal{M} . Así la dimensión de información se define como

$$d_{inf}(\mathcal{M}) = \lim_{q \rightarrow 1} D_q(\mu) \quad (5.18)$$

Lee muestra además como esta dimensión puede ser igual a la dimensión de capacidad (box-counting dimension) cuando la definición de la q -dimensión se plantea desde una perspectiva física y se realizan ciertas supuestos.

5.6.5. Dimensión de Correlación

En [58] se presenta la dimensión de correlación como un caso particular de la dimensión de información cuando $q = 2$. Si $\widehat{\mathcal{M}} = \{\mathbf{y}_1, \dots, \mathbf{y}_k, \dots, \mathbf{y}_P\}$ es el conjunto de puntos representativos del patrón que caen sobre la variedad \mathcal{M} , se define

$$C_2(\epsilon) = \lim_{P \rightarrow \infty} \frac{1}{P(P-1)} \sum_{\substack{i,j=1 \\ i \neq j}} H(\epsilon - \|\mathbf{y}_i - \mathbf{y}_j\|_2) \quad (5.19)$$

siendo $H(u)$ la función de Heaviside. Con esto la dimensión de correlación se define como

$$\dim_{corr}(\mathcal{M}) = \lim_{\epsilon \rightarrow 0} \frac{\log C_2(\epsilon)}{\log \epsilon} \quad (5.20)$$

5.6.6. Relación entre algunas dimensiones

Robinson demuestra en [72] que para cualquier espacio métrico compacto (X, ρ) se verifica que

$$\dim(X) \leq \dim_H(X) \leq \dim_{box}(X) \quad (5.21)$$

y Lee presenta en [58] que se cumple además la siguiente desigualdad

$$\dim_{corr}(\mathcal{M}) \leq \dim_{info}(\mathcal{M}) \leq \dim_{box}(\mathcal{M}) \quad (5.22)$$

Un aspecto importante para resaltar de acuerdo a lo expuesto en [58] es que la dimensión de correlación puede subestimar la dimensión real en determinados casos.

5.6.7. Estimación Práctica de la Dimensión Intrínseca

Las dimensiones presentadas anteriormente aunque bien definidas, son extremadamente difíciles de calcular si no se cuenta con una expresión analítica que describa la variedad \mathcal{M} sobre la cual recaen los datos patrón [58], esto es principalmente cierto si se habla de las dimensiones de recubrimiento o de Hausdorff. En el caso de la *box-counting dimension* aunque es posible plantear métodos aproximados para tratar numéricamente la operación de límite, el costo computacional termina convirtiéndola en inapropiada para problemas en donde existan grandes dimensiones que es donde precisamente se busca estimar la dimensión [58]. La dimensión de correlación aunque inicialmente es un planteamiento que requiere también de la operación de límite, es atractiva desde el punto de vista computacional [69] [58] y por lo tanto esto la convierte en una herramienta para la estimación de la dimensión en situaciones prácticas.

Adicionalmente, existen otras técnicas consideradas aceptables en muchas aplicaciones, que no parten de las definiciones formales de dimensión dadas anteriormente, pero que buscan examinar comportamientos locales como en el caso de PCA local (**PCA**– ℓ) [58], o que buscan adecuar computacionalmente una aproximación de una dimensión teórica como en el caso de *Packing Numbers* [74]. A continuación se discuten varias de estas estimaciones prácticas.

Dimensión de Correlación

En [58] se propone aproximar la dimensión de correlación de un conjunto finito de datos $\widehat{\mathcal{M}}$ que caen sobre la variedad \mathcal{M} como

$$d_{corr}(\widehat{\mathcal{M}}) = \frac{\log \hat{C}_2(\epsilon_2) - \log \hat{C}_2(\epsilon_1)}{\log \epsilon_2 - \log \epsilon_1} \quad (5.23)$$

siendo $\hat{C}_2(\epsilon)$ una aproximación de $C_2(\epsilon)$ dada por el número finito de puntos con los que se cuenta para estimar la dimensión de la variedad. Este es un estimado de la pendiente media de una gráfica de $\log C(\epsilon)$ vs $\log \epsilon$. Los valores de ϵ_1 y ϵ_2 se seleccionan entre el mínimo y el máximo de las distancias entre puntos medidos en $\widehat{\mathcal{M}}$. De acuerdo a los análisis realizados por Lee [58] los valores de $\epsilon_{1,2}$ no deben ser demasiado pequeños ya que se obtendría una dimensión cercana a cero que es la dimensión de un conjunto aislado de puntos, e igualmente valores grandes de estos parámetros no concuerdan con el tipo de aproximación que se quiere lograr, ya que se trata de aproximar un límite que tiende a cero.

Der Maaten [75] desarrolló un código en matlab en donde se usa el esquema propuesto por Lee [58], usando los valores de ϵ_1, ϵ_2 como la mediana de las distancias y el valor máximo respectivamente, obteniéndose buenos resultados dentro de las pruebas realizadas por ese autor. Sin embargo la selección de estos valores es arbitraria y no parece haber una razón de peso para haber seleccionado estos parámetros de esa manera.

Por otra parte Theiler en [76] encuentra de manera rigurosa y exhaustiva un criterio para la selección de estos dos valores conocido como *la regla de cinco*. En este trabajo, se seguirá este criterio que se describe de la siguiente manera: sea n_2 el número de distancias inferior a ϵ_2 y sea n_1 el número de distancias inferior a ϵ_1 . Una vez se ha seleccionado ϵ_2 el valor de ϵ_1 se selecciona de tal manera que $n_2/n_1 \simeq 5$; con estos valores es posible usar parte del esquema desarrollado por Der Maaten [75], seleccionando ϵ_2 como el valor máximo de las distancias. Este algoritmo se describe como

1. Se computa la matriz de distancias entre pares de puntos \mathbf{D} , para el conjunto $\widehat{\mathcal{M}}$.
2. Se selecciona $\epsilon_2 = \max(\mathbf{D})$.
3. Se encuentra n_2 el número de distancias menores a ϵ_2 .
4. Se calcula $n_1 = \text{ceil}(\frac{n_2}{5})$ como el número de distancias menores a ϵ_1 , de manera que el valor de este término queda determinado.
5. Calcular

$$\hat{C}_2(\epsilon_\alpha) = \frac{1}{P(P-1)} \sum_{\substack{i=1 \\ i \neq j}}^P H(\epsilon_\alpha - \|\mathbf{y}_i - \mathbf{y}_j\|_2) \quad (5.24)$$

para $\alpha = 1, 2$

6. Finalmente se calcula el estimado de la dimensión de correlación como

$$d_{corr}(\widehat{\mathcal{M}}) = \frac{\log \hat{C}_2(\epsilon_2) - \log \hat{C}_2(\epsilon_1)}{\log \epsilon_2 - \log \epsilon_1} \quad (5.25)$$

En la tabla 5.1 se pueden apreciar los valores estimados de la dimensión de correlación para la información de las expresiones faciales $\widehat{\mathcal{M}}_{FSF}$ del sujeto S055 de la base de datos Cohn-Kanade [7] extraídos con el esquema propuesto en el capítulo anterior, allí se consigna además la cantidad de puntos empleados para realizar dicho cálculo.

	$d_{corr}(\widehat{\mathcal{M}}_{FSF})$	P
S055_01	3	12
S0055_02	3	25
S0055_03	2	9
S0055_04	2	28
S0055_05	3	45
S0055_06	2	8
S0055_all	3	127

Cuadro 5.1: Estimación de la Dimensión de Correlación de $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7].

PCA Local (PCA- ℓ)

Lee presenta en [58] una técnica que se basa principalmente en determinar la dimensión usando **PCA** sobre subconjuntos disyuntos del total de la información. Es decir, si $\widehat{\mathcal{M}} = \{\mathbf{y}_1, \dots, \mathbf{y}_k, \dots, \mathbf{y}_P\}$ es el conjunto de puntos que recaen sobre la variedad \mathcal{M} . La estimación de la dimensión usando **PCA- ℓ** consiste en seleccionar una familia de subconjuntos $\widetilde{\mathcal{A}} = \{\widetilde{U}_1, \dots, \widetilde{U}_j\}$ de $\widehat{\mathcal{M}}$ tales que $\widetilde{U}_i \cap \widetilde{U}_j = \emptyset$ y $\bigcup_i \widetilde{U}_i = \widehat{\mathcal{M}}$, siendo el tamaño de los \widetilde{U}_i uniforme. Se aplica entonces **PCA** sobre cada uno de estos conjuntos para estimar su dimensión intrínseca, así se obtiene un estimado de la dimensión de la variedad $\widehat{\mathcal{M}}$ graficando dicho valor como la mediana o la media de los valores obtenidos para cada subconjunto vs el número de subconjuntos empleados para realizar cada estimación.

De esta manera, la estimación estará asociada a regiones planas de la gráfica donde se *estabiliza* el valor de la dimensión. A medida que aumenta el número de ventanas la cantidad de datos asociados a cada U_j es menor y por ello se asemeja con más propiedad al comportamiento local de la variedad \mathcal{M} . El valor estimado será la ordenada de la región plana de la gráfica correspondiente a los valores más altos en cuanto al número de ventanas empleadas.

Acorde a Lee [58] esta técnica muestra ser bastante atractiva por su practicidad y además resulta ser acertada en los estudios experimentales realizados por este autor. No obstante esta misma técnica muestra una gran dependencia de la cantidad de datos de $\widehat{\mathcal{M}}$, razón por la cual puede ser limitada si el número de datos es reducido. Teniendo en cuenta que en la base de datos Cohn-Kanade [7] existen conjuntos de imágenes muy reducidos que describen el comportamiento dinámico de una determinada expresión, una aplicación directa de este procedimiento para este tipo de información sería fuertemente cuestionable.

Por otra parte, a pesar de los grandes atractivos de esta técnica por su simplicidad y aprovechamiento del comportamiento local, subyace un problema que aun hoy es considerado por expertos como abierto: la selección del número de componentes. Jolliffe realiza en [46] un análisis comparativo de varias técnicas y criterios para realizar esta selección, sin embargo una de las principales conclusiones obtenidas por este autor, es que aún este tema es la causa de numerosos desarrollos estadísticos y matemáticos con fundamentaciones cada vez más elaboradas, pero con pocas ventajas o ninguna sobre los procedimientos simples en la mayoría de los casos prácticos, en donde se desconoce qué estructura forman los datos que representan un determinado patrón.

Dentro de estas reglas simples Jolliffe [46] describe el método basado en la famosa gráfica de los eigenvalores ordenados de mayor a menor y el cuál consiste en la ubicación del *codo* de la curva trazada en esta gráfica, que va a corresponder a un punto cuya vecindad izquierda es una región de gran pendiente (en valor absoluto) mientras que la de su derecha es casi plana. Lee menciona esta misma técnica en [58] pero aplicando logaritmo sobre los eigenvalores. A pesar de que innegablemente puede existir en algunos casos, un grado de subjetividad en la selección del número de componentes por medio de este procedimiento, ésta es igual o menor a la subjetividad de otras técnicas más sofisticadas en casos donde no se conoce a profundidad la naturaleza de la información [46]. Por otra parte Lee en [58] sugiere el uso del criterio AIC (Akaike's Information Criterion)

y/o el criterio MDL presentados por Cichocki [77] como alternativas rigurosas para este fin. En este trabajo se propone integrar en una sola metodología las ideas de Jolliffe [46] y Lee [58], buscando un intermedio entre el *codo* de la curva en una gráfica construida empleando el criterio AIC y el criterio AIC mismo, la presentación de esta estrategia se realizará en la sección 5.6.8 cuando se plantee la versión modificada de **PCA**– ℓ .

5.6.8. Algunos Resultados Sobre Variedades Diferenciales: Acerca de la Dimensión, *PCA local Modificado* (PCA – ℓm)

De acuerdo a la definición formal de variedad topológica dada por Lee en [71]:

Definición 3 Si \mathcal{M} es un espacio topológico, se dice que \mathcal{M} es una variedad topológica de dimensión n si se verifican las siguientes propiedades:

1. \mathcal{M} es un espacio de Hausdorff: Para cada par de puntos $p, q \in \mathcal{M}$ existen subconjuntos abiertos disjuntos $U, V \subset \mathcal{M}$ tales que $p \in U$ y $q \in V$.
2. \mathcal{M} es segundo contable: Existe una base contable para la topología de \mathcal{M} .
3. \mathcal{M} es localmente Euclidiano de dimensión n : Cada punto de \mathcal{M} tiene una vecindad que es homeomorfa a un abierto de \mathbb{R}^n .

y a la idea intuitiva de variedad brindada por Massey en [66]:

... El análogo n -dimensional de una superficie es una variedad de dimensión n , la cual es un espacio topológico con las mismas propiedades locales de un espacio euclidiano n -dimensional ...

se puede afirmar que para todo punto de la variedad \mathcal{M} de dimensión n existe una vecindad homeomorfa a un subconjunto abierto de \mathbb{R}^n , por lo tanto *cualquier* punto de la variedad debe brindar información sobre la dimensión de la misma. Esto otorga un argumento sólido para la aplicación de PCA local en la estimación de la dimensión, pues una estimación a diferentes escalas de la variedad debe corresponder a un mismo valor.

En las situaciones prácticas será necesario tener en cuenta que solo se cuenta con un subconjunto de datos de esta variedad, por lo cual solo habrá un rango de escalas adecuado para la estimación de esta dimensión. En este trabajo se propone teniendo en cuenta estas ideas una variante de PCA local, que se denominará **PCA**– ℓm (PCA local modificado), en la cual se usan familias de subconjuntos de $\widehat{\mathcal{M}}$ no disjuntos y así estimar la dimensión aplicando sobre éstos PCA para estimar la dimensión. Ésta se plantea de la siguiente manera.

```
Sea  $\widehat{\mathcal{M}} = \{y_1, \dots, y_P\}$  el conjunto de datos que caen sobre la variedad  $\mathcal{M}$ 
for k=1:1:P-1
    for i=1:1:P-k
        d(i)= Estimación Dimensión por PCA de  $[y_i, \dots, y_{i+k}]$ ;
    end
    df(k)=median(d);
end
```

La salida de este algoritmo entregará un vector **df** que graficado vs el tamaño de la ventana se espera tenga distintas regiones planas. Se considerará como el estimado de la dimensión al valor de ordenada correspondiente a la primera de estas regiones planas, ya que este representa el mínimo valor al cual se *estabiliza* la dimensión calculada localmente (ver figura 5.1).

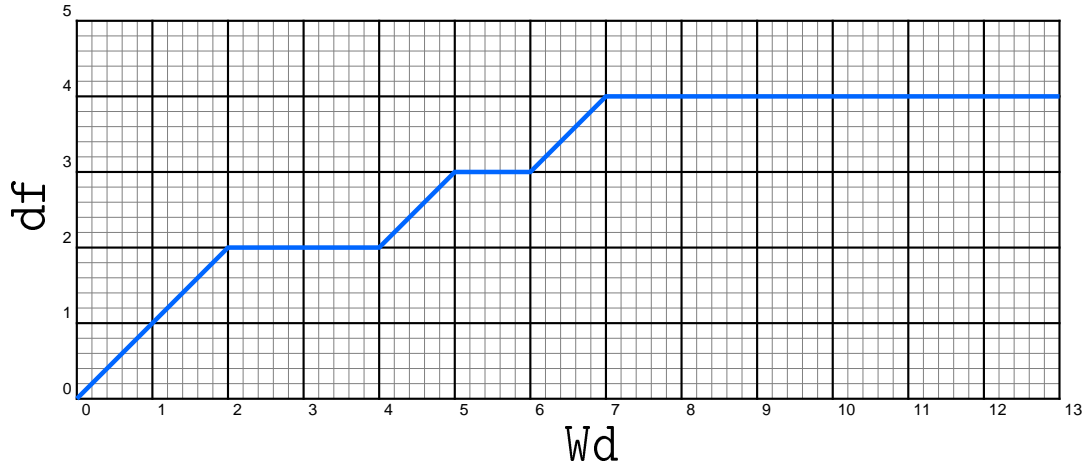


Figura 5.1: Gráfica típica de df vs el tamaño de la ventana Wd en la aplicación de **PCA**– ℓm . La primera región plana asociada a la ordenada de valor 2 indica el valor estimado de la dimensión.

Sobre el Número de Componentes usadas (¿Cuántas Componentes?)

En la presentación de PCA local se mencionó el inconveniente que aun hoy en día existe para determinar el número de eigenvalores de tal manera que se realice una distinción adecuada entre ruido e información esencial. En este trabajo se propone realizar esta selección integrando las ideas de Jolliffe [46], Lee [58] y Cichoky [77].

Mientras Jolliffe propone por una parte el uso del método gráfico para buscar el *codo* de la curva de los eigenvalores, Lee y Cichoki presentan el criterio AIC para realizar la distinción entre ruido e información después de la aplicación de componentes principales. De acuerdo a este criterio el número de componentes que representan la información viene dado por el entero n que minimiza la función

$$AIC(n) = -2P(m - n) \log(\rho(n)) + 2n(2m - n) \quad (5.26)$$

siendo m el número de eigenvalores obtenidos mediante la aplicación de componentes principales, P el número de datos u observaciones empleadas para construir la matriz de covarianza y $\rho(n)$ la función definida como

$$\rho(n) = \frac{(\lambda_{n+1}\lambda_{n+2}\dots\lambda_m)^{\frac{1}{m-n}}}{\frac{1}{m-n}(\lambda_{n+1} + \lambda_{n+2} + \dots + \lambda_m)} \quad (5.27)$$

y λ_i el eigenvalor correspondiente al i -ésimo eigenvector.

La aplicación de este criterio asume que los vectores de representación son de media cero y las variables son independientes e idénticamente distribuidas, que son de acuerdo a Jolliffe [46] y Lee [58] condiciones asumidas cuando se realiza la aplicación de **PCA** sobre un conjunto de datos.

Por otra parte la búsqueda del *codo* de la curva mencionado por Jolliffe [46] y Lee [58] se basa principalmente en una observación gráfica de cambios drásticos de pendiente. En este trabajo se cuantifica esta idea teniendo en cuenta que dicho *codo* posee una vecindad a su izquierda cuya pendiente es elevada (en valor absoluto) y una vecindad a su derecha cuya pendiente es casi nula.

Así, la propuesta de este trabajo se basa en buscar el *codo* de la curva $AIC(n)$, en caso de que esta última función se minimice solo con el número total de eigenvalores. En la figura 5.2 se muestran dos gráficas típicas de la función $AIC(n)$, en donde se evidencia que no siempre se halla un mínimo local en un número menor a la cantidad total de eigenvalores, pero si existen marcados cambios de comportamiento que son cercanos al de un mínimo local. Esta propuesta se describe como

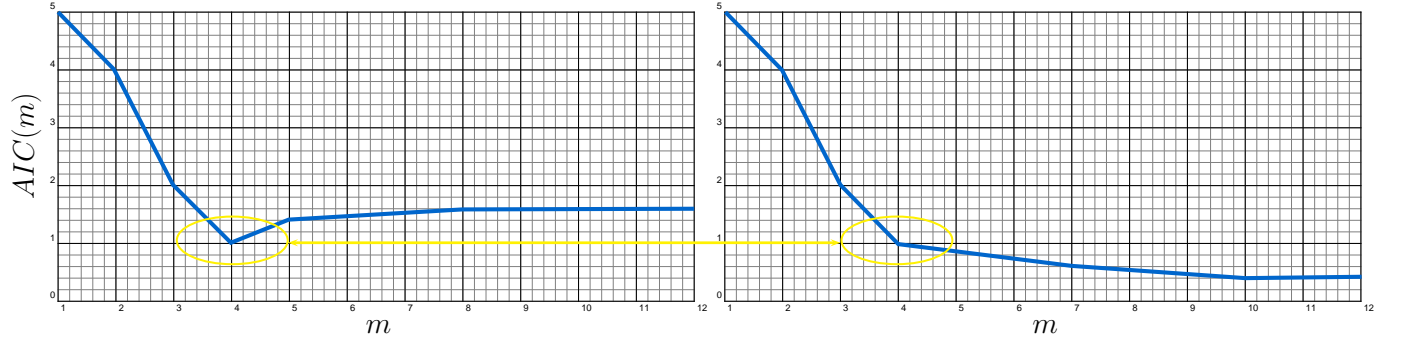


Figura 5.2: Dos comportamientos típicos en la curva AIC . Izquierda: Caso en el cual existe un n que minimiza la función $AIC(n)$. Derecha: Caso en el cual el valor mínimo de AIC se logra con el número total de eigenvalores, pero existe un marcado cambio de comportamiento con un número menor de los mismos

1. Sea $\lambda = [\lambda_1, \lambda_2, \dots, \lambda_m]$ el conjunto de los eigenvalores diferentes de cero asociados a los eigenvectores calculados empleando PCA, ordenados de mayor a menor.
2.

```
for ii=2:1:m-2
    If  $|AIC(ii+1) - AIC(ii)| - |AIC(ii) - AIC(ii-1)| < 0$ 
        
$$\hat{\partial}_{AIC}(ii) = ||AIC(ii+1) - AIC(ii)| - |AIC(ii) - AIC(ii-1)|| \quad (5.28)$$

    else
        
$$\hat{\partial}_{AIC}(ii) = 0 \quad (5.29)$$

    end
end
```

 adicionalmente $\hat{\partial}_{AIC}(1) = 0, \hat{\partial}_{AIC}(m) = 0, \hat{\partial}_{AIC}(m-1) = |AIC(m-1) - AIC(m-2)|$.
3. Encuentre el valor mínimo de $AIC(i)$, $i = 1, \dots, m$. Si dicho mínimo se ubica se ubica en $i = n$, $n < m$, entonces el número de componentes seleccionado es n y el proceso finaliza, de lo contrario vaya al paso 4.
4. Encuentre el valor máximo de $\hat{\partial}_{AIC}$. Si este valor ocurre en $i = n_{\partial}$, el número de componentes seleccionadas \hat{n} se calcula como $\hat{n} = n_{\partial}$.

El condicional presentado en el paso 2 establece como debe ser el cambio de pendiente en el posible punto donde se encuentra el codo de la curva, y sencillamente establece que se debe pasar de un valor de pendiente grande (en valor absoluto) a uno pequeño (en valor absoluto).

Los datos de los resultados obtenidos sobre el conjunto $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7] con esta técnica son presentados en la tabla 5.2, en donde además se encuentran también los estimados por medio de la dimensión de correlación.

5.6.9. Algunos Resultados Sobre Variedades: Sobre el Teorema de Whitney, *un Nuevo Método para la Estimación de la Dimensión* (PCA – ω)

El teorema de Whitney [71], descrito en forma clásica puede expresarse como:

Teorema 2 *Toda variedad diferenciable admite un embedding(encajamiento, incrustamiento) pro-*

	$\mathbf{PCA}-\ell m \left(\widehat{M}_{FSF} \right)$	$d_{corr} \left(\widehat{M}_{FSF} \right)$
S055_01	3	3
S0055_02	2	3
S0055_03	2	2
S0055_04	3	2
S0055_05	2	3
S0055_06	2	2
S0055_all	2	3

Cuadro 5.2: Estimación de la Dimensión de \widehat{M}_{FSF} usando $\mathbf{PCA}-\ell m \left(\widehat{M}_{FSF} \right)$ y $d_{corr} \left(\widehat{M}_{FSF} \right)$, para el sujeto S055 de la base de datos Cohn-Kanade [7]

pio suave en \mathbb{R}^{2n+1}

Y posterior al planteamiento del anterior teorema Whitney demostró un teorema mucho más fuerte

Teorema 3 Si $n > 0$ toda variedad suave admite un embedding(encajamiento, incrustamiento) suave en \mathbb{R}^{2n}

En estos resultados se establece que es posible encajar o incrustar una variedad diferenciable de dimensión n en el espacio \mathbb{R}^{2n+1} o \mathbb{R}^{2n} según corresponda. Como se ha comentado anteriormente, una técnica como **PCA** no tiene plenas capacidades para *desenvolver* las variedades formadas por algunos conjuntos de patrones y realizar un adecuado homeomorfismo del espacio de dimensiones originales a un espacio de la dimensión intrínseca [58]. Sin embargo, se propone estimar la dimensión intrínseca de los datos aplicando **PCA**, de manera que aunque no se asegura la obtención de la dimensión intrínseca n de la variedad sobre la cual caen los datos, se puede llegar al estimado de la dimensión del conjunto incrustado en \mathbb{R}^{2n+1} o \mathbb{R}^{2n} según corresponda.

Acorde a Lee [71], el primero de estos teoremas brinda una respuesta al problema matemático de cuándo una variedad abstracta es difeomorfa a una subvariedad de un espacio euclidiano y es suficiente en muchos contextos para la solución de algunos problemas y la demostración de otros resultados teóricos. Mientras que el segundo teorema demostrado ocho años después, y con el uso de conceptos y herramientas matemáticas más sofisticadas es un resultado más fuerte con implicaciones prácticas.

En este trabajo se propone una nueva técnica para la estimación de la dimensión aprovechando estos resultados teóricos y las dos hipótesis siguientes:

1. Los vectores descriptores del patrón, expresados como vectores de N componentes caen o hacen parte de una variedad diferenciable de dimensión $n \leq N$.
2. Esta variedad *sin ruido* y sin la redundancia menos intrincada está incrustada en el espacio \mathbb{R}^{2n} .

La primera hipótesis es común al uso de técnicas de reducción no lineal [59] [58], y la segunda asegura que si los datos extraídos representan un determinado patrón que está asociado a una variedad de dimensión n , ésta debe estar incrustada en el espacio \mathbb{R}^{2n} si el número de componentes N de los vectores descriptores es $N \gg n$.

La consideración de esta segunda hipótesis sigue la interpretación de Lee en [58] del teorema de Whitney. Éste asocia el Teorema al hecho de que se requieren *a lo más* $2n$ dimensiones para representar la variedad de dimensión n . Adicionalmente este mismo autor en [58] sugiere como un esquema alternativo para la reducción de dimensionalidad aplicar primero **PCA** a los datos y luego

cualquier técnica de reducción de dimensionalidad que resulte adecuada dependiendo del tipo de variedad (en caso de que sea posible extraer algunas conclusiones sobre esta estructura). Esto debido a que **PCA** puede ayudar a atenuar ruido estadístico en los datos y a eliminar la redundancia menos intrincada sin una gran pérdida de información, siempre y cuando el número de componentes haya sido seleccionado correctamente [58].

Así, una vez aplicado **PCA** se asume que se obtiene un conjunto de datos encajado o incrustado en \mathbb{R}^{2n} . Si este no es estrictamente el caso y los datos obtenidos tienen dimensión de $2n - 1$ se puede adicionar un vector de ceros de tal manera que los datos puedan considerarse incrustados en \mathbb{R}^{2n} .

En resumen esta segunda hipótesis puede resumirse como

Sea $\widehat{\mathcal{M}}$ el conjunto de P vectores de N componentes extraídos para representar un determinado patrón. Si estos datos caen sobre una variedad \mathcal{M} de dimensión n y $N \gg n$, al estimar la dimensión de $\widehat{\mathcal{M}}$ usando PCA, se puede considerar que la dimensión estimada corresponde a una incrustación o encajamiento de la variedad \mathcal{M} en \mathbb{R}^{2n} . Es decir la variedad sobre la que recaen los datos es representada respecto de la dimensión n empleando un número de variables mayor al número máximo de variables que se requieren para ello, y el exceso de este número de variables se puede eliminar usando PCA

En esta hipótesis se argumenta que $N \gg n$, pero en verdad esta idea es mucho más específica y es posible establecer una relación entre estas dos cantidades.

Teniendo en cuenta que en la representación de la variedad se presume que se usan más variables que las dadas por el número máximo necesario para dicha representación, y que de acuerdo a la interpretación de Lee [58] del teorema de Whitney [71] este número es $2n$, la anterior desigualdad se representa ahora como $N \geq 2n$. Por esto es importante contar con una estimación de la dimensión, que no se requiere sea exacta, pero que brinde una idea de qué tan lejana es la dimensión intrínseca de la dimensión original de los datos.

Así, el algoritmo planteado para realizar dicha estimación es el siguiente

1. Sea $\widehat{\mathcal{M}}$ el conjunto de datos que caen sobre la variedad \mathcal{M} de dimensión n .
2. Se realiza una estimación de la dimensión usando **PCA- ℓm** y dimensión de correlación sobre $\widehat{\mathcal{M}}$ y se selecciona este valor como

$$\widehat{n} = \left\lceil \frac{\text{PCA} - \ell m(\widehat{\mathcal{M}}) + d_{\text{corr}}(\widehat{\mathcal{M}})}{2} \right\rceil \quad (5.30)$$

3. Si $2\widehat{n} \leq N$ entonces vaya al paso 4, de lo contrario se aborta la estimación.
4. Usar PCA sobre el conjunto $\widehat{\mathcal{M}}$ para estimar su dimensión $k \in \mathbb{N}$.
5. Si $k = 2n$ con $n \in \mathbb{N}$ entonces la dimensión estimada es **PCA- ω** $(\widehat{\mathcal{M}}) = \frac{k}{2}$, de lo contrario haga **PCA- ω** $(\widehat{\mathcal{M}}) = \frac{k+1}{2}$.

En el paso 3 se pretende establecer si las dimensiones originales de los vectores patrón poseen un número mayor al número de $2n$ de dimensiones que a lo más se requieren para incrustar la variedad de dimensión n . Si esto se verifica se procede a aplicar **PCA** para eliminar el ruido y la redundancia menos intrincada, de manera que se llega a una representación usando el número máximo de variables que se requieren para ello. Para estimar la dimensión en el paso 4 se emplea el criterio basado en *AIC* presentado en la sección anterior y usado en **PCA- ℓm** .

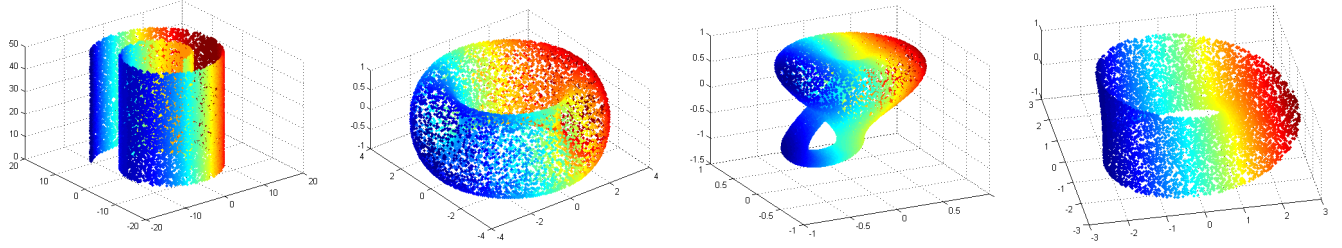


Figura 5.3: De izquierda a derecha: El rollo suizo, El Toro, La botella de Klein, La banda de Möbius.

	$\mathbf{PCA}-\ell m \left(\widehat{\mathcal{M}}_{FSF} \right)$	$d_{corr} \left(\widehat{\mathcal{M}}_{FSF} \right)$	$\mathbf{PCA}-\omega \left(\widehat{\mathcal{M}}_{FSF} \right)$
S055_01	3	3	2
S0055_02	2	3	4
S0055_03	2	2	2
S0055_04	3	2	2
S0055_05	2	2	3
S0055_06	2	2	2
S0055_all	2	3	3
<i>Swiss Roll</i>	2	3	2
<i>Klein Bottle</i>	2	4	2
<i>Torus</i>	2	3	2
<i>Mobius Strip</i>	2	3	2

Cuadro 5.3: Estimación de la Dimensión de $\widehat{\mathcal{M}}_{FSF}$ usando $\mathbf{PCA}-\ell m \left(\widehat{\mathcal{M}}_{FSF} \right)$, $d_{corr} \left(\widehat{\mathcal{M}}_{FSF} \right)$ y $\mathbf{PCA}-\omega \left(\widehat{\mathcal{M}}_{FSF} \right)$ para el sujeto S055 de la base de datos Cohn-Kanade [7]

Comparación con Otros Métodos

Se realiza la contrastación de la dimensión estimada por esta nueva técnica, con la dimensión de correlación y con la dimensión obtenida por $\mathbf{PCA} - \ell m$. Las pruebas se muestran sobre los datos $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7] y sobre cuatro variedades *benchmark* (ver figura 5.3), las cuales están descritas por una distribución de 2000 puntos ubicados aleatoriamente sobre la variedad por medio de una distribución uniforme. Cada componente de la descripción original se contamina con ruido generado por una distribución uniforme cuya magnitud varía entre $\pm 10\%$ de los valores máximos asociados a cada coordenada. Se agregan tres componentes adicionales de ruido con valores alrededor de $\pm 0,01$, de manera que la variedad resultante está incrustada en \mathbb{R}^6 .

Los resultados obtenidos con esta propuesta para estimar la dimensión, pueden observarse en la tabla 5.3, en donde además se presentan los valores estimados empleando la dimensión de correlación. Éstos son cercanos a los obtenidos por los otros estimadores de la dimensión, y aprovecha las ventajas asociadas a \mathbf{PCA} , junto con el criterio basado en *AIC* para la selección del número de componentes. Por esto se pondrá a consideración el uso de esta técnica a fin de contar con un estimado adicional de la dimensión intrínseca.

Otras Estimaciones prácticas de la Dimensión

Se han desarrollado diversas propuestas para realizar la estimación práctica de la dimensión, sin embargo algunas de ellas aun con muchos aspectos por detallar y mejorar. Una de las más prometedoras, fué desarrollada por Kegl [74] denominada *Packing Numbers*, la cual se basa en una aproximación de la dimensión de capacidad, similar a la que se realiza para estimar la dimensión de correlación. Según los experimentos realizados por Kegl [74] esta técnica muestra una mayor robustez frente a ciertos niveles de ruido en comparación con la dimensión de correlación, sin embargo su correcto funcionamiento depende de la selección de dos parámetros, para los cuales no existe un criterio óptimo de selección. Esto es particularmente inconveniente si se desea abordar el problema de una estimación automática. Por esta razón no se considerará para este trabajo.

Por otra parte Levina [78] propone el uso del principio de máxima similitud, para analizar las distancias entre vecinos cercanos, a fin de dar un estimado de la dimensión intrínseca. Esta propuesta es usada en algunas implementaciones [75], sin embargo los resultados obtenidos dependen fuertemente de algunos parámetros para los cuales no existen criterios de selección óptima, en el análisis de conjuntos arbitrarios de datos. Por esta razón no se considerará para este trabajo.

Relación entre Estimaciones Prácticas

La primera de las desigualdades presentadas en la sección 5.6.6, demostrada por Robinson en [72] relaciona de manera rigurosa la dimensión topológica, la dimensión de Hausdorff y la *box-counting dimension*. Sin embargo éstas son de carácter teórico y no se pueden aplicar directamente para analizar conjuntos finitos de datos que recaen sobre variedades diferenciales.

La desigualdad propuesta por Lee [58] es un tanto más cercana a la práctica ya que está en términos de la dimensión de correlación, para la cual pueden obtenerse valores aproximado. De acuerdo a esta desigualdad es posible que el valor obtenido para esta dimensión sea menor a la dimensión topológica de la variedad sobre la cual recaen los datos.

Por otra parte, teniendo en cuenta la fundamentación teórica asociada a **PCA**– ℓ_m que se basa en análisis local, se cuenta con un valor que no sobreestimaré la dimensión de la variedad, de allí que los resultados experimentales mostrados en la tabla 5.2 sean semejantes entre sí.

De acuerdo a los resultados experimentales es adecuado considerar estas dos técnicas y **PCA**– ω cuando se requiera un análisis automático de la dimensión para un conjunto dado de puntos.

5.7. Técnicas Basadas en Preservación de la Distancia

5.7.1. Distancia Euclidiana

A continuación se presentan las distintas técnicas cuyo principio es la conservación de la distancia euclidiana entre los datos en el espacio original de representación.

Multidimensional Scaling (MDS)

Existe una gran cantidad de variantes de ésta técnica, pero en este trabajo sólo se discutirá la versión clásica del método [58]. Éste puede describirse por medio del siguiente algoritmo [79] [58]

1. Si $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_P]$ es la matriz de $N \times P$ dimensiones formada por los datos $\widehat{\mathcal{M}}$ que recaen sobre la variedad \mathcal{M} . Se computa la matriz de productos internos

$$\mathbf{S} = \mathbf{Y}^T \mathbf{Y} \quad (5.31)$$

2. Se calcula la descomposición espectral

$$\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \quad (5.32)$$

siendo \mathbf{U} la matriz cuyos vectores columna están formados por los primeros n eigenvectores de la matriz \mathbf{S} con mayor valor de sus respectivos eigenvalores, y $\mathbf{\Lambda}$ la matriz diagonal cuyos elementos son estos eigenvalores ordenados de mayor a menor valor.

3. La representación en el espacio n -dimensional se obtiene como

$$\mathbf{X} = \mathbf{I}_{n \times P} \mathbf{\Lambda}^{\frac{1}{2}} \mathbf{U}^T \quad (5.33)$$

En este trabajo de acuerdo a los esquemas planteados se cuenta con un conjunto de datos $\widehat{\mathcal{M}}$ que recae sobre una variedad; sin embargo en muchas de las aplicaciones de **MDS** se cuenta inicialmente con una matriz de distancias $\mathbf{D}_{P \times P}$ entre los diferentes puntos, en ese caso el primer paso se reemplazaría por el siguiente

1. Calcular

$$\mathbf{S} = -\frac{1}{2} \left(\mathbf{D} - \frac{1}{P} \mathbf{1}_P \mathbf{1}_P^T - \frac{1}{P} \mathbf{1}_P \mathbf{1}_P^T \mathbf{D} + \frac{1}{P^2} \mathbf{1}_P \mathbf{1}_P^T \mathbf{D} \mathbf{1}_P \mathbf{1}_P^T \right) \quad (5.34)$$

en donde $\mathbf{1}_P$ representa un vector columna cuyas todas componentes son iguales a 1, este proceso es conocido como de doble centrado [79] [58].

Así, **MDS** opera preservando los productos internos euclidianos de los vectores de datos representativos del patrón, y es equivalente a usar **PCA**. Por lo tanto se pueden asociar las mismas ventajas y desventajas que posee esta última técnica, como por ejemplo el hecho de que se asume una relación lineal entre la información original y la información resultante [58], es decir una relación lineal entre las variables originales y las variables latentes de la variedad, lo cual resulta a nivel general altamente insatisfactorio.

Por otra parte esta misma semejanza con **PCA** posee dos enormes ventajas, es simple y robusto; además es el concepto del uso de distancias el que permitió lograr una generalización de **MDS** conocida como **Isomap** (se discutirá más adelante) que emplea aproximación de distancias geodésicas y no distancias Euclidianas.

Lee [58] realiza pruebas de **MDS** para las *benchmark manifolds: rollo suizo y la caja abierta*. Como se esperaba de una técnica que asume una relación lineal, los resultados obtenidos no son satisfactorios pues no se logra una representación adecuada de estas variedades en dos dimensiones.

Los resultados de emplear esta técnica sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7] se pueden apreciar en la tabla 5.4, allí se muestra la dimensión en la cual se incrustan los datos conservando el 98 % de la energía de los eigenvalores y se compara con el estimado de la dimensión de correlación (que también puede verse en la tabla 5.1). Teniendo en cuenta esto se puede asegurar que un encajamiento en un espacio de menores dimensiones es insatisfactorio debido a la lejanía entre la dimensión de correlación y la dimensión necesario para preservar las cualidades esenciales de la información con esta técnica (e.g 98 %), ya que si bien la aproximación de la dimensión de correlación puede ser un subestimado de la dimensión real de la variedad sobre la cual caen los datos, la diferencia es demasiado grande.

Por estas razones **MDS** no es atractivo para realizar la reducción dimensionalidad de los datos representativos de las expresiones faciales. Sin embargo, acorde a Lee [58] es posible e incluso conveniente usar **PCA-MDS** en caso de que la cantidad de datos sea demasiado escasa.

Sammon's Nonlinear Mapping (NLM)

En el mapeo no lineal de Sammon, se plantea mapear los puntos patrón extraídos del espacio euclidiano N -dimensional a un espacio euclidiano de dimensión n preservando las distancias por

	dimensión ($\geq 98\%$)	$dim_{corr}(\widehat{\mathcal{M}}_{FSF})$	P
S055_01	5	3	12
S0055_02	7	3	25
S0055_03	6	2	9
S0055_04	4	2	28
S0055_05	10	2	45
S0055_06	5	2	8
S0055_all	11	3	127

Cuadro 5.4: Estimación de la dimensión usando **MDS** sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7]. La dimensión estimada se basa en conservar el 98 % de la energía de los eigenvalores

medio de la minimización de una función de estrés definida como [80] [58]

$$E_{NLM} = \frac{1}{c} \sum_{j=2}^P \left(\sum_{i=1}^{j-1} \frac{(d_{\mathbf{y}}(i, j) - d_{\mathbf{x}}(i, j))^2}{d_{\mathbf{y}}(i, j)} \right) \quad (5.35)$$

donde $d_{\mathbf{y}}(i, j)$ es la distancia euclidiana entre \mathbf{y}_i y \mathbf{y}_j en el espacio N -dimensional, y $d_{\mathbf{x}}(i, j)$ es la distancia euclidiana entre los mapeos de \mathbf{y}_i y \mathbf{y}_j en el espacio n -dimensional. La constante c está definida como [80] [58]

$$c = \sum_{j=2}^P \left(\sum_{i=1}^{j-1} d_{\mathbf{y}}(i, j) \right) \quad (5.36)$$

de acuerdo a Lee [58] y Sammon [80] este algoritmo se puede plantear como

1. Se calculan los pares de distancias $d_{\mathbf{y}}(i, j)$ en el espacio euclidiano N -dimensional.
2. Se inicializan los puntos \mathbf{x}_i en el espacio n -dimensional, esta inicialización puede ser aleatoria, o aplicando PCA y tomando los primeros n eigenvectores de acuerdo a la energía asociada a sus autovalores.
3. Se actualiza

$$\mathbf{x}_i(k) \leftarrow \mathbf{x}_i(k) - \alpha \frac{\frac{\partial E_{NLM}}{\partial \mathbf{x}_i(k)}}{\left| \frac{\partial^2 E_{NLM}}{\partial \mathbf{x}_i(k)^2} \right|} \quad (5.37)$$

donde de acuerdo con Sammon [80] el valor de α se puede asignar cercano a 0,5 [81].

4. Se computa

$$E_{NLM} = \frac{1}{c} \sum_{j=2}^P \left(\sum_{i=1}^{j-1} \frac{(d_{\mathbf{y}}(i, j) - d_{\mathbf{x}}(i, j))^2}{d_{\mathbf{y}}(i, j)} \right) \quad (5.38)$$

5. Si $E_{NLM} < \Delta_{err}$ se termina el proceso, de lo contrario se regresa al paso 3. Donde Δ_{err} es el límite del error deseado.

el procedimiento de minimización de la función de estrés, expresando en el paso 3 es conocido como *Cuasi-Newton* [80] [58], y los términos de primera y segunda derivada se calculan como

$$\frac{\partial E_{NLM}}{\partial \mathbf{x}_i(k)} = \frac{\partial E_{NLM}}{\partial d_{\mathbf{x}}(i, j)} \frac{\partial d_{\mathbf{x}}(i, j)}{\partial \mathbf{x}_i(k)} \quad (5.39)$$

donde

$$\frac{\partial d_{\mathbf{x}}(i, j)}{\partial \mathbf{x}_i(k)} = \frac{\mathbf{x}_i(k) - \mathbf{x}_j(k)}{d_{\mathbf{x}}(i, j)} \quad (5.40)$$

de manera que

$$\frac{\partial E_{NLM}}{\partial \mathbf{x}_k(i)} = -\frac{2}{c} \sum_{\substack{j=1 \\ j \neq i}}^P \frac{d_{\mathbf{y}}(i, j) - d_{\mathbf{x}}(i, j)}{d_{\mathbf{y}}(i, j)} \frac{(\mathbf{x}_i(k) - \mathbf{x}_j(k))}{d_{\mathbf{x}}(i, j)} \quad (5.41)$$

la segunda derivada puede expresarse como

$$\frac{\partial^2 E_{NLM}}{\partial \mathbf{x}_k^2(i)} = -\frac{2}{c} \sum_{\substack{j=1 \\ j \neq i}}^P \left(\frac{d_{\mathbf{y}}(i, j) - d_{\mathbf{x}}(i, j)}{d_{\mathbf{y}}(i, j) d_{\mathbf{x}}(i, j)} - \frac{(\mathbf{x}_i(k) - \mathbf{x}_j(k))^2}{d_{\mathbf{x}}^3(i, j)} \right) \quad (5.42)$$

los resultados de aplicar esta técnica a los datos $\widehat{\mathcal{M}}_{FSF}$ del sujeto S055 de la base de datos Cohn-Kanade [7] se pueden ver en la figura 5.4. Cada conjunto de *features* es representado en un espacio de dimensión igual a la dimensión estimada $\hat{n} = \dim_{corr}(\widehat{\mathcal{M}}_{FSF})$.

En la figura 5.4 se evidencia que en algunos casos de secuencias individuales de datos de expresiones, hay solapamiento de la información respecto del espacio dado por la estimación de la dimensión de correlación. Sin embargo cuando se analiza la variedad que incluye todos los comportamientos no hay intersecciones entre las distintas trayectorias que representan los distintos tipos de expresiones, ya que el espacio en cuestión tiene la dimensión más alta de las estimadas para cada variedad analizada por expresión. Esto indica que dicho solapamiento se podría atribuir a una subestimación de la dimensión.

De acuerdo a la presentación de estos resultados para los conjuntos de features obtenidos, podría considerarse que los mismos son satisfactorios, sin embargo Lee analiza en [58] el comportamiento de esta técnica sobre las variedades *rollo suizo* y *caja abierta* obteniendo resultados no satisfactorios respecto de la dimensión real de estas variedades. Pues si bien *Sammon's nonlinear mapping* es más capaz de lidiar con la no linealidad en comparación de técnicas como **PCA** y **MDS**, no es recomendable si la variedad posee grandes radios de curvatura [58].

Lee menciona en [58] además el inconveniente de que a función de estrés en la que se basa este método no se puede garantizar que sea siempre cóncava, por lo cual se podría caer en un mínimo local. Esta técnica se comparará con las analizadas en secciones subsiguientes.

Curvilinear Component Analysis (CCA)

De manera similar a **NLM**, *Curvilinear Component Analysis (CCA)* se basa en la preservación de las distancias por medio de la minimización de la función de estrés [82] [65] [58]:

$$E_{CCA} = \frac{1}{2} \sum_{\substack{i=1 \\ j=1}}^N (d_{\mathbf{y}}(i, j) - d_{\mathbf{x}}(i, j))^2 F_{\lambda}(d_{\mathbf{x}}(i, j)) \quad (5.43)$$

donde $d_{\mathbf{y}}(i, j)$ es la distancia euclidiana entre los puntos \mathbf{y}_i y \mathbf{y}_j en el espacio \mathbb{R}^N y $d_{\mathbf{x}}(i, j)$ es la distancia euclidiana de entre los puntos mapeados en el espacio \mathbb{R}^n . La función F_{λ} es escogida para preservar prioritariamente las distancias cortas sobre las largas. Consecuentemente F_{λ} es seleccionada usualmente como una función monótona decreciente de su argumento [82] [65]. Además dado

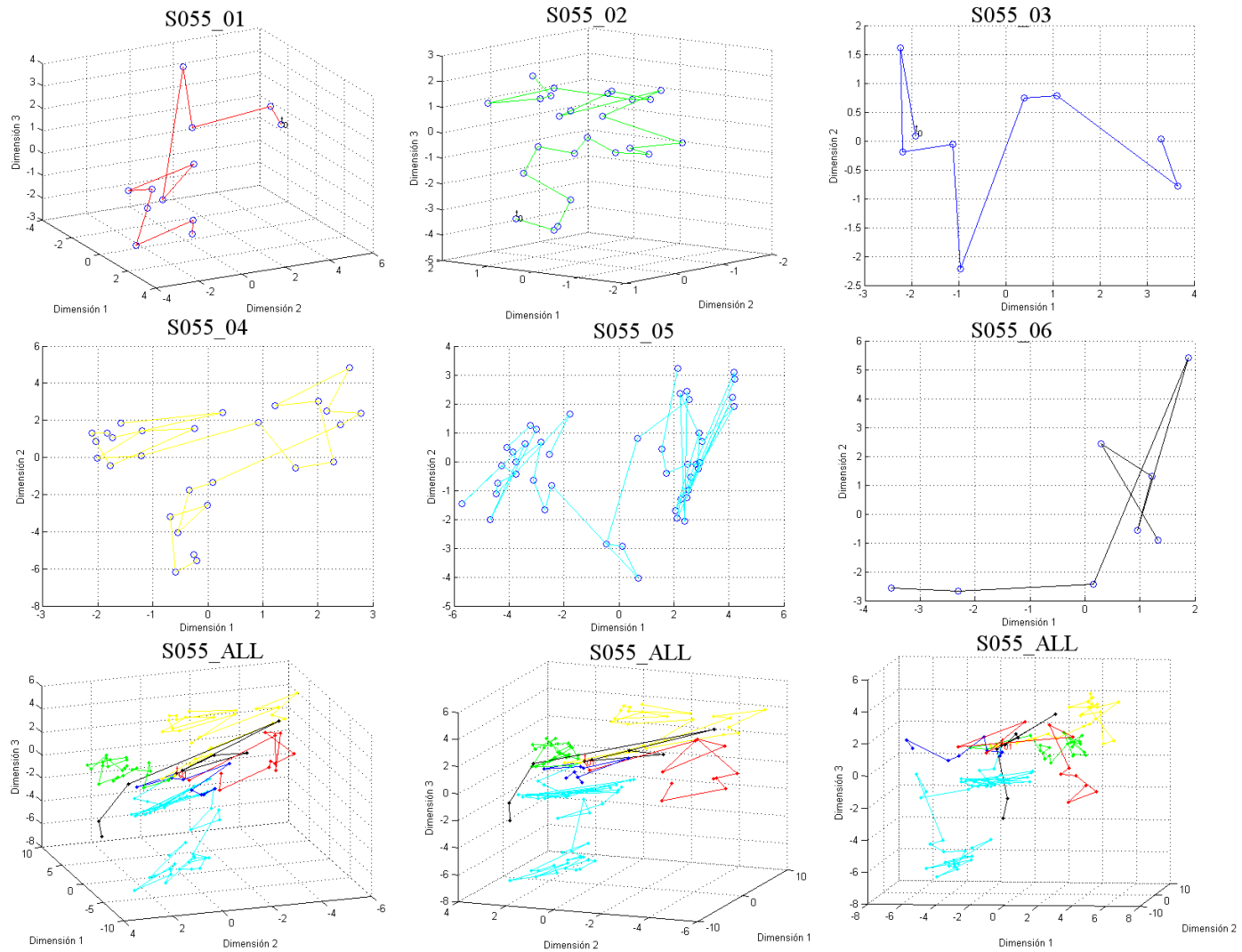


Figura 5.4: Resultados de aplicar $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], empleando la dimensión de correlación. Algoritmo de Vesanto [81]

que el método opera sobre un conjunto finito de datos, esta función también es escogida como acotada, de tal manera que no existan problemas para manejar distancias demasiado cortas o incluso nulas [82]. Lee [58] muestra que esta función se selecciona como

$$F_\lambda(d_{\mathbf{x}}(i, j)) = e^{-\frac{d_{\mathbf{x}}(i, j)}{\lambda}} \quad (5.44)$$

El método **CCA** puede describirse como [82] [65] [58]

1. Compute todas las distancias mutuas $d_{\mathbf{y}}(i, j)$ en el espacio euclidiano \mathbb{R}^N .
2. Se inicializan las coordenadas en el espacio n -dimensional de todos los puntos \mathbf{x}_i ya sea aleatoriamente o usando PCA. Se hace $q = 1$ (q es el número de época o iteración).
3. Se actualiza el valor de α y λ para el número de época q .
4. Se selecciona un punto \mathbf{x}_j y se actualizan los demás acorde a
`for i=1:1:P`

$$\mathbf{x}_j \leftarrow \mathbf{x}_j - \alpha \nabla_{\mathbf{x}_j} E_{CCA}^i$$

$$\leftarrow \mathbf{x}_j - \alpha \beta(i, j) \frac{\mathbf{x}_i - \mathbf{x}_j}{d_{\mathbf{x}}(i, j)} \quad (5.45)$$

$$(5.46)$$

`end`

5. Retorne al paso 4 hasta que cada punto \mathbf{x}_i haya sido seleccionado al menos una vez por época.
6. Incremente el valor de q , y si la convergencia no se ha logrado retorne al paso 3.

En el tercer paso se hace referencia a la actualización de los valores de α y λ . Éstos pueden ser actualizados de distintas maneras, sin embargo Vesanto en [83] ha encontrado adecuado realizar estas actualizaciones como

$$\alpha_l = \alpha_0 \left(\frac{\alpha_F}{\alpha_0} \right)^{\frac{l-1}{q_{\max} P - 1}} \quad (5.47)$$

$$\lambda_l = \lambda_0 \left(\frac{\lambda_F}{\lambda_0} \right)^{\frac{l-1}{q_{\max} P - 1}} \quad (5.48)$$

para $l = 1, \dots, q_{\max} P$. En particular, en [83] se muestra que se encuentran resultados satisfactorios para una gran variedad de problemas seleccionando $\alpha_0 = 0,5$, $\alpha_F = \frac{\alpha_0}{100}$, $\lambda_0 = \max \left\{ 3\sigma_{d_{\mathbf{y}}(i, j)} \right\}$ tomando las desviaciones correspondientes a las distancias finitas y $\lambda_F = 0,01$. Donde q_{\max} es el número máximo de iteraciones o épocas.

En el paso 4 se plantean las ecuaciones de actualización para cada uno de los puntos \mathbf{x}_j . Éstas se basan en un planteamiento propuesto en [82] [65]. La idea básica es realizar la minimización de la función de estrés usando el método de descenso por el gradiente. Para esto se usa la derivada de la función de estrés respecto de cada una de las coordenadas de los puntos en el espacio \mathbb{R}^n :

$$\nabla_{\mathbf{x}_i} E_{CCA} = \frac{\partial E_{CCA}}{\partial \mathbf{x}_i(k)} = \frac{\partial E_{CCA}}{\partial d_{\mathbf{x}}} \frac{\partial d_{\mathbf{x}}}{\partial \mathbf{x}_i(k)} = \sum_{j=1}^P (d_{\mathbf{y}} - d_{\mathbf{x}}) \left(2F_\lambda(d_{\mathbf{x}}) - (d_{\mathbf{y}} - d_{\mathbf{x}}) F'_\lambda(d_{\mathbf{x}}) \right) \frac{\mathbf{x}_j(k) - \mathbf{x}_i(k)}{d_{\mathbf{x}}} \quad (5.49)$$

de manera que la actualización de cada uno de los \mathbf{x}_i usando descenso por el gradiente, se da de la siguiente forma

$$\mathbf{x}_i \leftarrow \mathbf{x}_i - \alpha \nabla_{\mathbf{x}_i} E_{CCA} \quad (5.50)$$

siendo α una tasa positiva de aprendizaje seleccionada teniendo en cuenta las condiciones de Robins-Monro [58]. Demartines demostró en [82] [65] que esta actualización poseía algunos inconvenientes asociados al estancamiento en mínimos locales, por lo cual propuso realizar dicho proceso desacoplando la información de la función de estrés. Esto se hace teniendo en cuenta que:

$$E_{CCA} = \sum_{i=1}^P E_{CCA}^{(i)} \quad (5.51)$$

donde

$$E_{CCA}^{(i)} = \frac{1}{2} \sum_{j=1}^P (d_{\mathbf{y}}(i, j) - d_{\mathbf{x}}(i, j))^2 F_{\lambda}(d_{\mathbf{x}}(i, j)) \quad (5.52)$$

así en el nuevo procedimiento se realiza la actualización en todos los \mathbf{x}_j para cada i como:

$$\mathbf{x}_j \leftarrow \mathbf{x}_j - \alpha \nabla_{\mathbf{x}_j} E_{CCA}^{(i)} = \mathbf{x}_j - \alpha \beta(i, j) \frac{\mathbf{x}_i - \mathbf{x}_j}{d_{\mathbf{x}}(i, j)} \quad (5.53)$$

es decir primero se actualizan todos los \mathbf{x}_j para $i = 1$, luego se realiza la actualización de todos los \mathbf{x}_j para $i = 2$ y así sucesivamente. El término $\beta(i, j)$ se define como

$$\beta(i, j) = (d_{\mathbf{y}} - d_{\mathbf{x}}) \left(2F_{\lambda}(d_{\mathbf{x}}) - (d_{\mathbf{y}} - d_{\mathbf{x}}) F'_{\lambda}(d_{\mathbf{x}}) \right) \quad (5.54)$$

los resultados de aplicar esta técnica a los datos $\widehat{\mathcal{M}}_{FSF}$ del sujeto S055 de la base de datos Cohn-Kanade [7] se pueden ver en la figura 5.5, cada conjunto de features es representado en un espacio de dimensión igual a la dimensión estimada $\hat{n} = \dim_{corr}(\widehat{\mathcal{M}}_{FSF})$. Como se evidencia en esta gráfica hay solapamiento de la información en el espacio respecto al cual se ha dado la reducción de dimensionalidad aunque esto también puede deberse al igual que en el método anterior a una subestimación de la dimensión en el cálculo de la dimensión de correlación ya que cuando se analiza toda la información como parte de una sola variedad la dimensión estimada es de tres, y allí no se evidencia solapamiento.

Lee [58] analiza el comportamiento de **CCA** aplicándolo sobre las variedades *rollo suizo* y *caja abierta*, encontrando resultados superiores a los obtenidos con **NLM**. Además de esto Lee [58] muestra que **CCA** trata de *estirar* la variedad en casos en los que **NLM** la *aplata* superponiendo la información.

Desde el punto de vista computacional, **CCA** muestra ser mucho más rápido que **NLM**, sin embargo con la desventaja de que en **CCA** hay una gran dificultad para interpretar el error puesto que la función F_{λ} está cambiando en el proceso de iteración y el hecho de que el proceso de convergencia depende fuertemente de la actualización de α y λ . Esta técnica se comparará con otras técnicas que se presentarán más adelante.

5.7.2. Técnicas Basadas en Distancias de Grafo

A continuación se presentan las técnicas de conservación de la distancia en donde se emplean aproximaciones de geodésicas por medio de las distancias de grafo.

Isometric Mapping (Isomap)

Isomap es una técnica basada en **MDS** pero en la cual se usan aproximaciones de las geodésicas que unen los datos que recaen sobre la variedad \mathcal{M} , por medio de distancias de grafo [84] [58]. Así,

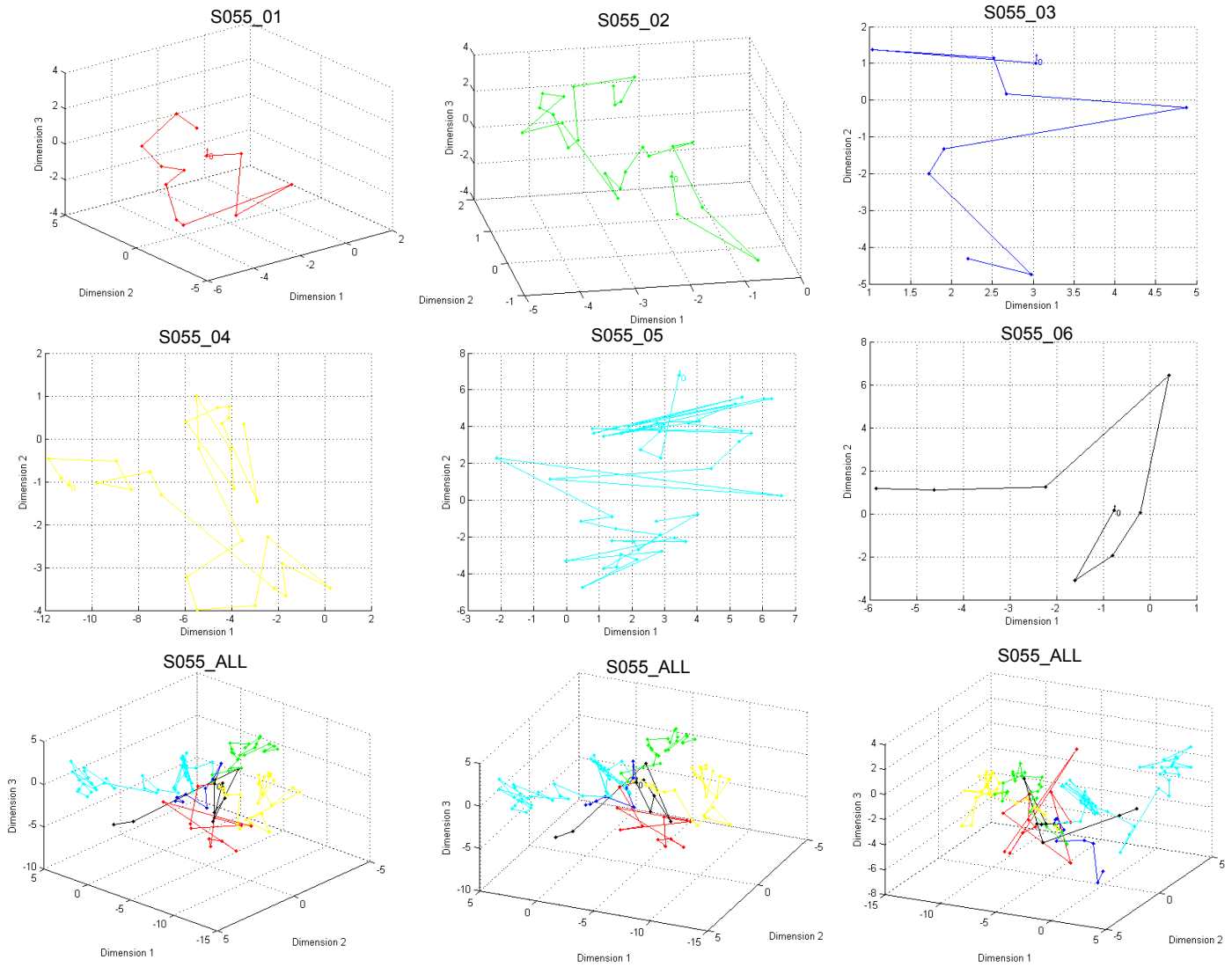


Figura 5.5: Resultados de aplicar **CCA** sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], empleando la dimensión de correlación. Algoritmo de Vesanto [83]

el manejo de la no linealidad usando **Isomap** reside únicamente en el hecho de emplear aproximaciones de distancias geodésicas. Acorde a Lee [58] y Tenenbaum [84] este procedimiento puede describirse como

1. Construya un grafo en el espacio \mathbb{R}^N con los puntos $\mathbf{Y} = [\mathbf{y}_1, \dots, \mathbf{y}_P]$.
2. Asigne a cada arista del grafo un peso dado por la distancia euclidiana entre los vértices del mismo.
3. Se calculan las distancias de grafo entre los puntos del mismo, por ejemplo usando el algoritmo de Dijkstra, y se almacenan en la matriz \mathbf{D} .
4. Realice doble centrado sobre la matriz $\tilde{\mathbf{D}}$, siendo $\tilde{\mathbf{D}}_{ij} = \mathbf{D}_{ij}^2$, obteniendo la matriz \mathbf{S} .
5. Se realiza la descomposición espectral

$$\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \quad (5.55)$$

siendo \mathbf{U} la matriz cuyos vectores columna están formados por los primeros n eigenvectores de la matriz \mathbf{S} con mayor valor de sus respectivos eigenvalores, y $\mathbf{\Lambda}$ la matriz diagonal cuyos elementos son estos eigenvalores ordenados de mayor a menor valor.

5. Finalmente se computa la representación en el espacio n -dimensional como

$$\mathbf{X} = \mathbf{I}_{n \times P} \mathbf{\Lambda}^{\frac{1}{2}} \mathbf{U}^T \quad (5.56)$$

Para conocer el efecto de tomar un número n de eigenvectores, o estimar la dimensión intrínseca de la variedad se analiza la varianza residual por medio de la ecuación [84] [58]

$$\sigma_n^2 = 1 - r_{ij}^2 \left(d_{\mathbf{x}}(i, j), \tilde{L}_{\mathbf{y}_i}^{\mathbf{y}_j} \right) \quad (5.57)$$

siendo r_{ij} el coeficiente de correlación sobre los índices i y j , y $\tilde{L}_{\mathbf{y}_i}^{\mathbf{y}_j}$ la distancia de grafo entre \mathbf{y}_i y \mathbf{y}_j . Para identificar la dimensión de los datos se realiza un análisis similar al que se hace para encontrar la dimensión intrínseca usando **PCA**, es decir se busca gráficamente dónde se ubica un punto de la gráfica a cuya derecha se encuentre una región casi plana y a la izquierda una región de elevada pendiente (en valor absoluto), ordenando las varianzas de mayor a menor [58] [84]. Lee [58] y Jolliffe [46] llaman a esto el *codo* de la curva, este procedimiento se puede apreciar en la presentación de **PCA- ℓm** (ver sección 5.6.8) en donde se emplea para la selección del número de componentes a fin de estimar correctamente la dimensión intrínseca.

Isomap muestra ser de acuerdo a los análisis realizados por Lee en [58] y a parte de los resultados experimentales de Ping en [57], una técnica más poderosa para manejar la no linealidad que las técnicas que se han presentado anteriormente. Sin embargo, **Isomap** solo tiene su *mejor comportamiento* cuando los datos $\hat{\mathcal{M}}$ que representan el patrón caen o hacen parte de una *variedad desarrollable* \mathcal{M} , y además de ello existe una isometría entre \mathcal{M} y un espacio euclidiano convexo [58] [59] [84]. Es decir, sea $L_{\mathbf{y}_i}^{\mathbf{y}_j}$ la distancia geodésica sobre \mathcal{M} de dimensión n , encajada en \mathbb{R}^N entre los puntos \mathbf{y}_i y \mathbf{y}_j , y sea $\mathcal{E} \subset \mathbb{R}^n$ convexo de dimensión n tal que $\mathbf{x}_i, \mathbf{x}_j \in \mathcal{E}$ son los puntos asociados a \mathbf{y}_i y \mathbf{y}_j por medio de **Isomap** respectivamente, $d_{\mathbf{x}}(i, j)$ es la distancia euclidiana entre los puntos \mathbf{x}_i y \mathbf{x}_j . Se dice que **Isomap** permite recuperar la estructura original de los datos si $\exists C \in \mathbb{R}$ tal que

$$L_{\mathbf{y}_i}^{\mathbf{y}_j} = C d_{\mathbf{x}}(i, j) \quad \forall i, j \quad (5.58)$$

Por otra parte, Bernstein muestra en [85] que a pesar del cumplimiento de las anteriores condiciones para asegurar un buen comportamiento de **Isomap**, debe considerarse la calidad de la aproximación de las distancias geodésicas por medio de las distancias de grafo. Bernstein [85] demuestra

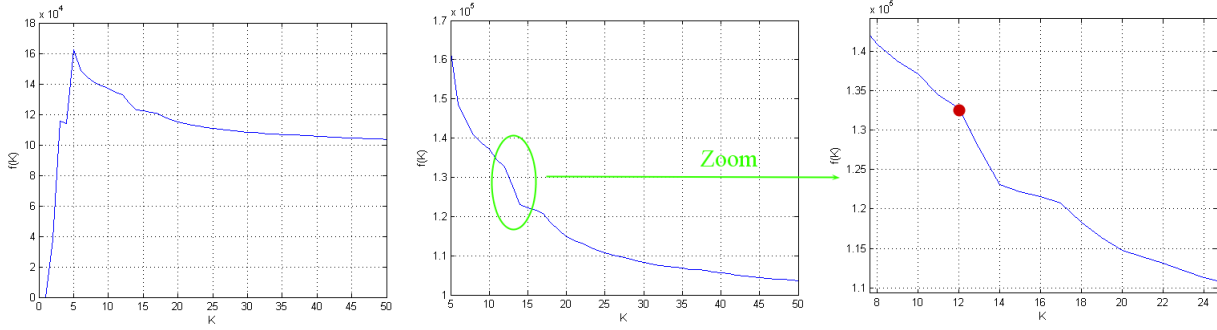


Figura 5.6: Gráfica para encontrar el valor óptimo de K propuesta por Shao [86] para los datos $\widehat{\mathcal{M}}_{FSF}$ del sujeto S055 de la base de datos Cohn-Kanade [7] construyendo el grafo a partir de distancias euclidianas, el valor óptimo sería de 12

que la curvatura de la variedad sobre la cual se asume que caen los datos patrón juega un papel importante para establecer la calidad de esta aproximación y adicionalmente la densidad de los datos también juega un papel fundamental.

A partir de este último argumento y a los análisis hechos por Lee en [58] se concluye que sería seriamente cuestionable emplear **Isomap** para una cantidad reducida de datos.

Por esto, se pondrá a consideración el uso de **Isomap** realizando su aplicación asumiendo que las variedades $\mathcal{M}_{FSF}, \mathcal{M}_{FST}, \mathcal{M}_{FIF}, \mathcal{M}_{FIT}, \mathcal{M}_{FIS}$ sobre las cuales recaen los datos patrón, están constituidas por todas las expresiones de un determinado individuo para garantizar una mayor cantidad de datos y con ello una posible mayor densidad de los mismos.

Otro aspecto a mencionar de suma importancia es la selección de los parámetros para la construcción de los grafos, en concreto el valor K que define la construcción del grafo por medio de los K -vecinos. Aunque algunos autores como Lee [58] proponen realizar una *sintonización empírica* de este valor observando los resultados, Shao propone en [86] una alternativa más rigurosa para esta selección, teniendo en cuenta las características de los datos a tratar sin realizar el mapeo al espacio de la dimensión estimada. Sea $\mathbf{D}^{(K)}$ la matriz en donde se almacenan las distancias de grafo, para un grafo construido con una valor de K empleando la técnica de los K -vecinos, Shao [86] propone analizar la función

$$f(K) = \sum_{i,j=1}^P \mathbf{D}^{(K)}(i, j) \quad (5.59)$$

y encontrar en ella el punto donde se registre un descenso abrupto en el comportamiento de $f(K)$. Esta gráfica se presenta por practicidad desde el primer valor de K donde $f(K)$ alcanza su máximo global. En la figura 5.6 se muestra el valor de K óptimo para la construcción del grafo empleando los datos $\widehat{\mathcal{M}}_{FSF}$ para todas las expresiones faciales del sujeto S055 de la base de datos Cohn-Kanade [7]. Para realizar este cálculo de manera automática se adopta un procedimiento similar al empleado en la búsqueda del *codo* de la curva empleado en **PCA**, pero con modificaciones respecto a los cambios de pendiente.

Entonces, para realizar la ubicación sistemática de este punto en el gráfico de $f(K)$ vs K se emplea el siguiente algoritmo

Sea Q el máximo tamaño de vecindad considerado, si no existe un estimado se puede hacer $Q = P$.

```
1. for ii=Kfmax + 1:1:Q-1
    if |f(ii + 1) - f(ii)| - |f(ii) - f(ii - 1)| > 0
```

$$f_{\partial}(ii) = ||f(ii + 1) - f(ii)| - |f(ii) - f(ii - 1)|| \quad (5.60)$$

```

end
end
 $f_{\partial}(1 : K_{fmax}) = 0$ 

```

2. Se busca el $ii = ii_{max}$ en el cual $f_{\partial}(ii)$ alcanza su valor máximo.

3. Se calcula el valor óptimo de K como $K = ii_{max}$

siendo $K = K_{fmax}$ el valor en el cual $f(K)$ alcanza su máximo global.

como se aprecia en este algoritmo los cambios de pendiente en donde se considera que se puede asociar el valor óptimo de K , serán los puntos cuya vecindad derecha tenga una gran pendiente (en valor absoluto) mientras que la izquierda una pendiente pequeña (en valor absoluto).

En las figura 5.7 se aprecian los resultados de emplear **Isomap** sobre la variedad $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7] visualizando los datos en 2 o 3 dimensiones dependiendo del comportamiento de la varianza.

Isomap se comparará con las demás técnicas empleadas para la reducción de dimensionalidad.

Supervised Isomap (s-Isomap)

Geng propone en [87] una variante de **Isomap** denominada *Supervised Isomap* en la cual se busca principalmente reducir la sensibilidad de **Isomap** al ruido, y brindar ventajas para fines de clasificación. La propuesta de Geng se basa en realizar esta mejora empleando una matriz de *disimilaridad* en la construcción del grafo teniendo en cuenta la clase o categoría a la que pertenecen los datos [87]. Sea $d_{\mathbf{y}}(i, j)$ la distancia euclidiana entre los puntos \mathbf{y}_i y \mathbf{y}_j que caen o hacen parte de la variedad \mathcal{M} . Se representarán como \mathcal{A}_{θ} , $\theta = 1, 2, \dots$ las distintas clases a las que pertenecen los datos. La matriz de disimilaridad propuesta por Geng se define como [87]

$$\mathfrak{D}(\mathbf{y}_i, \mathbf{y}_j) = \begin{cases} \sqrt{1 - e^{-\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} & \text{Si } \mathbf{y}_i, \mathbf{y}_j \in \mathcal{A}_{\theta} \quad \wedge \quad \sqrt{e^{\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} - \alpha \geq \sqrt{1 - e^{-\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} \\ \frac{\sqrt{e^{\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} - \alpha + \sqrt{1 - e^{-\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}}}{2} & \text{Si } \mathbf{y}_i, \mathbf{y}_j \in \mathcal{A}_{\theta} \quad \wedge \quad \sqrt{e^{\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} - \alpha < \sqrt{1 - e^{-\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} \\ \sqrt{e^{\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} - \alpha & \text{Si } \mathbf{y}_i \in \mathcal{A}_{\theta'}, \mathbf{y}_j \in \mathcal{A}_{\theta} \quad \wedge \quad \sqrt{1 - e^{-\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} \leq \sqrt{e^{\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} - \alpha \\ \frac{\sqrt{e^{\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} - \alpha + \sqrt{1 - e^{-\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}}}{2} & \text{Si } \mathbf{y}_i \in \mathcal{A}_{\theta'}, \mathbf{y}_j \in \mathcal{A}_{\theta} \quad \wedge \quad \sqrt{e^{\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} - \alpha > \sqrt{1 - e^{-\frac{d_{\mathbf{y}}^2(i, j)}{\beta}}} \end{cases} \quad (5.61)$$

Siendo β seleccionada como el promedio de las distancias euclidianas $d_{\mathbf{y}}(i, j)$ y α una constante seleccionada con un valor cercano a 0,5. De acuerdo a Geng [87] el uso de esta matriz en lugar de la matriz de distancias euclidianas presenta tres grandes ventajas

- ★ Cuando la distancia Euclidiana entre dos puntos de la variedad es cero o cercana a ser cero, la disimilaridad entre elementos de diferente clase (*interclase*) es mayor a la disimilaridad entre elementos de la misma clase (*intraclase*), lo cual permite lograr diferenciación geométrica entre secuencias de datos de distinta naturaleza sobre una misma variedad.
- ★ Cada función de disimilaridad es monótona creciente respecto a la distancia euclidiana lo que permite conservar las cualidades esenciales de la información, es decir su estructura geométrica esencial se preserva.
- ★ Dado que la disimilaridad *interclase* es $\geq 1 - \alpha$ y la disimilaridad *intraclase* es ≤ 1 es posible analizar sin mayores inconvenientes datos con niveles de ruido dentro de rangos más amplios que con **Isomap** original.

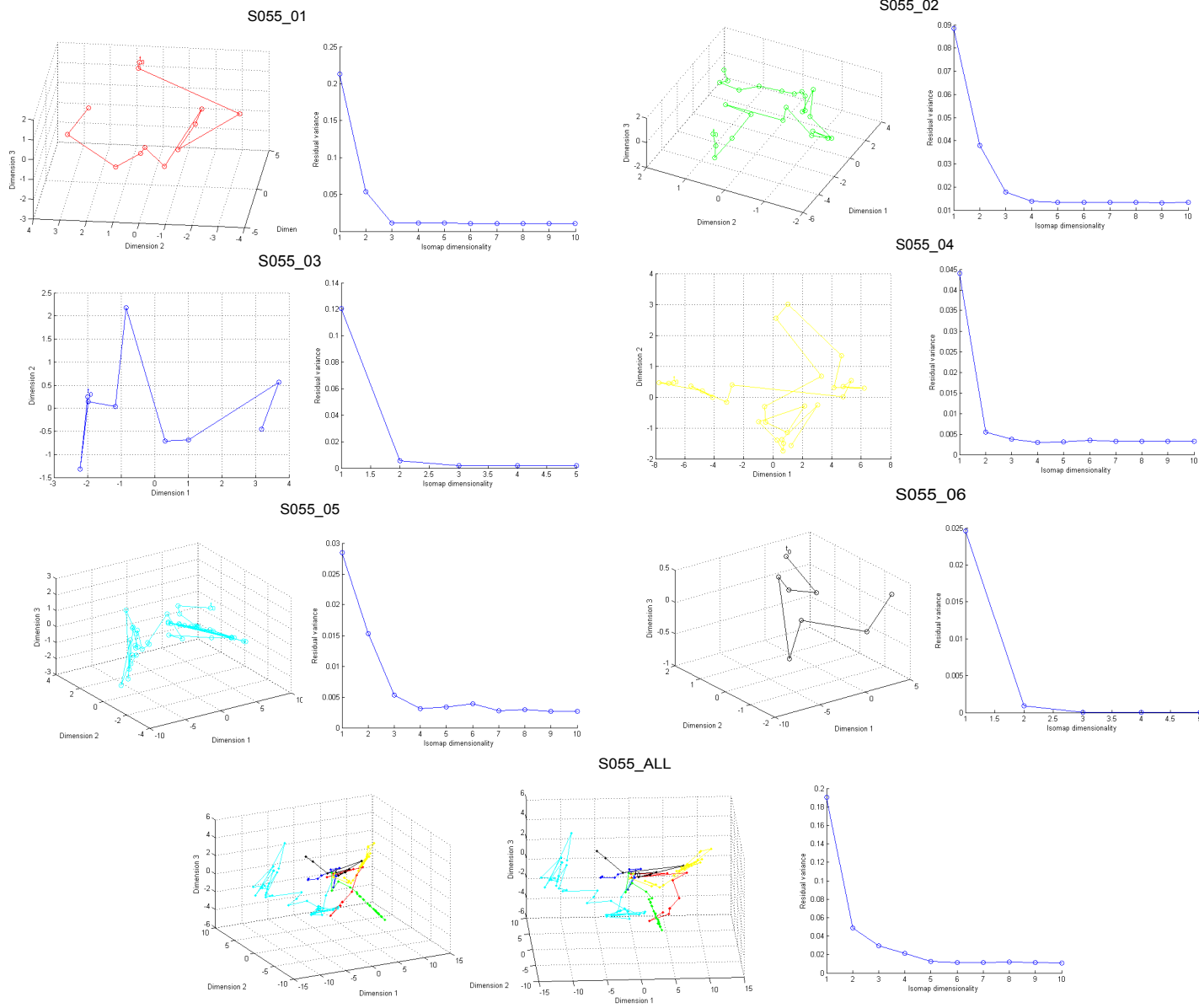


Figura 5.7: Resultados de aplicar **Isomap** sobre \widehat{M}_{FSF} para el sujeto S055 de la base de datos Cohn-Kanade [7], usando $K = 12$ seleccionado con el criterio asociado a $f(K)$. Algoritmo de Tenenbaum [84]

- ★ Con el crecimiento de la distancia euclidiana la disimilaridad interclase aumenta mucho más rápido que la disimilaridad intraclase, Geng describe este aspecto como una capacidad de $\mathfrak{D}(\mathbf{y}_i, \mathbf{y}_j)$ para distinguir entre la información y el ruido.

Teniendo en cuenta estas características, es claro que esta nueva versión de **Isomap** es mucho más robusta frente al ruido que la versión original, y esto brinda ventajas tanto para fines de visualización como de clasificación. Ping [57] obtiene resultados altamente satisfactorios para fines de clasificación empleando *s-Isomap* en conjuntos de expresiones faciales obtenidos de la base de datos Cohn-Kanade [7]. Este algoritmo puede presentarse como

1. Construya un grafo en el espacio \mathbb{R}^N con los puntos $\mathbf{Y} = [\mathbf{y}_1 \dots, \mathbf{y}_P]$.
2. Asigne a cada arista del grafo un peso dado por la matriz de disimilaridad $\mathfrak{D}(\mathbf{y}_i, \mathbf{y}_j)$ correspondiente a los vértices del mismo. Esta matriz se calcula empleando la ecuación 5.61
3. Se calculan las distancias de grafo entre los puntos del mismo, por ejemplo usando el algoritmo de Dijkstra, y se almacenan en la matriz \mathbf{D} .
4. Realice doble centrado sobre la matriz $\tilde{\mathbf{D}}$, siendo $\tilde{\mathbf{D}}_{ij} = \mathbf{D}_{ij}^2$, obteniendo la matriz \mathbf{S} .
5. Se realiza la descomposición espectral

$$\mathbf{S} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \quad (5.62)$$

siendo \mathbf{U} la matriz cuyos vectores columna están formados por los primeros n eigenvectores de la matriz \mathbf{S} con mayor valor de sus respectivos eigenvalores, y $\mathbf{\Lambda}$ la matriz diagonal cuyos elementos son estos eigenvalores ordenados de mayor a menor valor.

5. Finalmente se computa la representación en el espacio n -dimensional como

$$\mathbf{X} = \mathbf{I}_{n \times P} \mathbf{\Lambda}^{\frac{1}{2}} \mathbf{U}^T \quad (5.63)$$

Geng [87] compara *s-Isomap* con **Isomap** y otras de sus variantes encontrando resultados muy superiores para los propósitos de visualización. Adicionalmente Geng [87] demuestra que *s-Isomap* es una técnica robusta para fines de clasificación, de hecho ésta muestra ser ligeramente más robusta dentro de los experimentos realizados que otras técnicas de clasificación ya consolidadas como *SVM* (*support vector machines*).

Los resultados de aplicar *s-Isomap* a $\widehat{\mathcal{M}}_{FSF}$ empleando la totalidad de las expresiones para el sujeto S055 de la base de datos Cohn-Kanade [7] se pueden apreciar en la gráfica 5.8 usando los parámetros $\alpha = 0,5$, $K = 11$, en donde K se seleccionó siguiendo el procedimiento propuesto por Shao [86]. Teniendo en cuenta todos estos argumentos sobre *s-Isomap*, y los resultados obtenidos sobre $\widehat{\mathcal{M}}_{FSF}$, resulta evidente su ventaja para aplicaciones de reducción de dimensionalidad, ésta se comparará con las demás técnicas presentadas.

Geodesic NLM(GNLM)

GNLM se basa en la minimización de la función de estrés de Sammon presentada en la sección 5.7.1, pero emplea la aproximación de las distancias geodésicas en el espacio original en el cual se encuentran los datos [58]. De esta manera la función de estrés para esta técnica puede expresarse

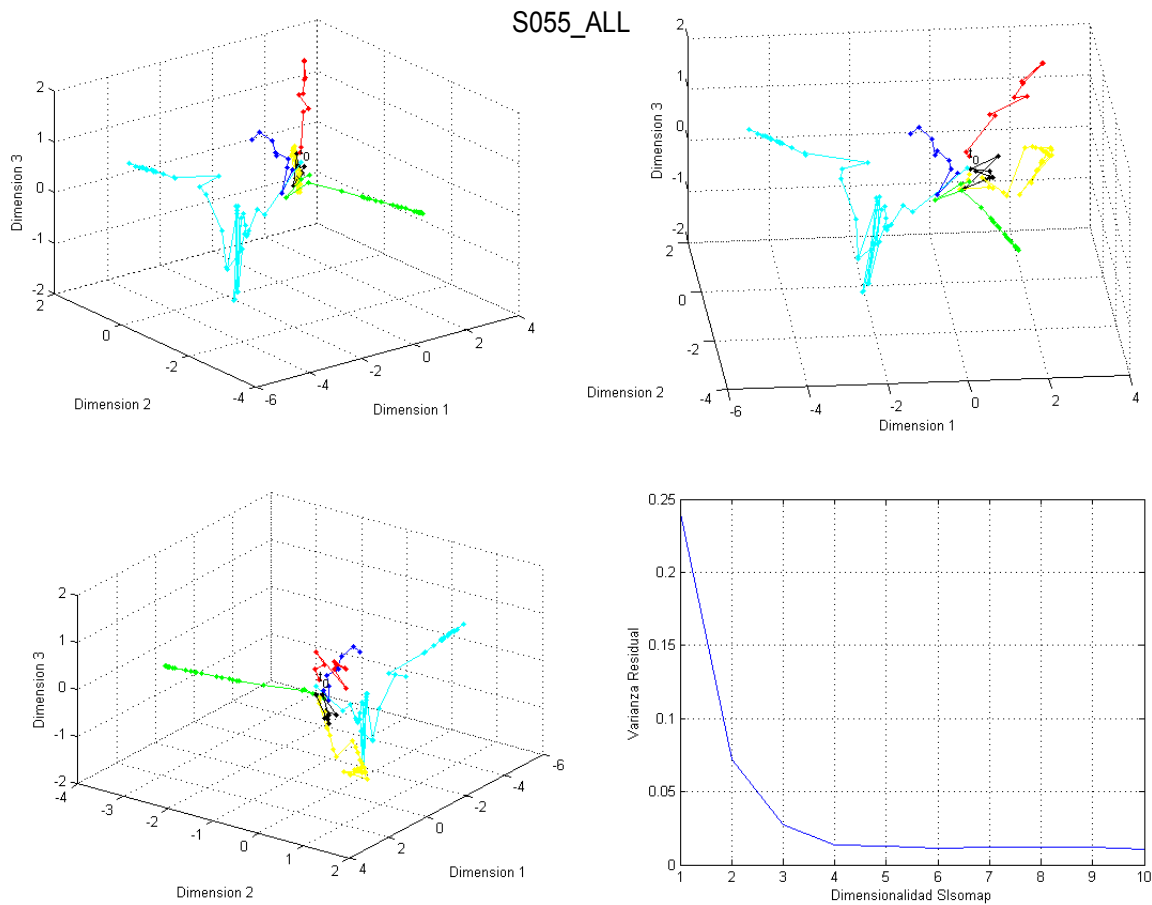


Figura 5.8: Resultados de aplicar s -**Isomap** sobre la variedad $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], usando $K = 11$ seleccionado con el criterio asociado a $f(K)$. Algoritmo Geng-Tenebaum [87]

como

$$E_{GNLM} = \frac{1}{c} \sum_{j=2}^P \left(\sum_{i=1}^{j-1} \frac{(\tilde{L}_{\mathbf{y}_i}^{y_j} - d_{\mathbf{x}}(i, j))^2}{\tilde{L}_{\mathbf{y}_i}^{y_j}} \right) \quad (5.64)$$

siendo $\tilde{L}_{\mathbf{y}_i}^{y_j}$ la aproximación de la distancia geodésica sobre \mathcal{M} entre los puntos $\mathbf{y}_i, \mathbf{y}_j \in \widehat{\mathcal{M}}$ por medio de la distancia de grafo, y al igual que en secciones anteriores $d_{\mathbf{x}}(i, j)$ es la distancia euclidiana entre los mapeos de los puntos $\mathbf{y}_i, \mathbf{y}_j$ al espacio de dimensión intrínseca. La constante c se calcula como

$$c = \sum_{j=2}^P \left(\sum_{i=1}^{j-1} \tilde{L}_{\mathbf{y}_i}^{y_j} \right) \quad (5.65)$$

Al igual que en **Isomap**, la construcción de un grafo involucra el problema de seleccionar un valor de óptimo de K en la regla de los K -vecinos. Dado que la propuesta de Shao [86] mencionada anteriormente hace referencia a la calidad de la aproximación de las distancias geodésicas sin necesidad de involucrar el mapeo o encajamiento en el espacio de dimensión intrínseca, se propone emplear esta misma regla para dicha selección.

El procedimiento de optimización asociado sigue siendo idéntico al asociado a NLM, de manera que el algoritmo que describe esta técnica se puede presentar como [58]

1. Construya un grafo en el espacio \mathbb{R}^N con los puntos $\mathbf{Y} = [\mathbf{y}_1 \dots, \mathbf{y}_P]$.
2. Asigne a cada arista del grafo un peso dado por la distancia euclidiana correspondiente a los vértices del mismo.
3. Se calculan las distancias de grafo entre los puntos del grafo, por ejemplo usando el algoritmo de Dijkstra, y se almacenan en la matriz \mathbf{D} .
4. Se inicializan los puntos \mathbf{x}_i en el espacio n -dimensional, esta inicialización puede ser aleatoria, o aplicando PCA y tomando los primeros n eigenvectores de acuerdo a la energía asociada a sus autovalores.
5. Se actualiza

$$\mathbf{x}_i(k) \leftarrow \mathbf{x}_i(k) - \alpha \frac{\frac{\partial E_{GNLM}}{\partial \mathbf{x}_i(k)}}{\left| \frac{\partial^2 E_{GNLM}}{\partial \mathbf{x}_i(k)^2} \right|} \quad (5.66)$$

donde de acuerdo con Sammon [80] el valor de α se puede asignar cercana al intervalo entre 0,3 y 0,4.

6. Se computa

$$E_{GNLM} = \frac{1}{c} \sum_{j=2}^P \left(\sum_{i=1}^{j-1} \frac{(\tilde{L}_{\mathbf{y}_i}^{y_j} - d_{\mathbf{x}}(i, j))^2}{\tilde{L}_{\mathbf{y}_i}^{y_j}} \right) \quad (5.67)$$

5. Si $E_{GNLM} < \Delta_{err}$ se termina el proceso, de lo contrario se regresa al paso 3. Donde Δ_{err} es el límite del error deseado.

Lee [58] menciona como una desventaja de la técnica el problema de la selección del valor óptimo de K en la construcción del grafo y la dependencia de la técnica de los valores de otros parámetros como α . Sin embargo, al tener en cuenta la propuesta de Shao [86] para conocer el valor óptimo de K , el primer aspecto deja de ser un problema y solo queda por analizar el comportamiento dependiente de α que de acuerdo a Vesanto [81] puede seleccionarse con un valor de 0,5 de tal manera

que se obtengan resultados satisfactorios para amplias variedades de datos.

Otras limitantes de **GNLM** están relacionadas, al igual que en **Isomap**, con la calidad de aproximación las distancias geodésicas sobre la variedad \mathcal{M} por medio de las distancias de grafo, descritas a partir de los datos $\widehat{\mathcal{M}}$. Como se mencionó anteriormente Bernstein en [85] demostró que tanto la densidad de los datos como el radio de curvatura de la variedad sobre la cual caen éstos, afecta de manera significativa la calidad de la aproximación de las distancias geodésicas dadas por las distancias de grafo. Por ello aplicar esta técnica a conjuntos reducidos de datos sería bastante cuestionable.

Por otra parte **GNLM** tiene la ventaja de manejar la no linealidad por medio de las distancias de grafo como lo hace **Isomap** y además por medio de la minimización de una función de estrés. Lee en [58] realiza pruebas con esta técnica sobre las variedades *rollo suizo* y *caja abierta* obteniendo resultados altamente satisfactorios, y por supuesto **GNLM** tiene un mejor comportamiento que **Isomap** para variedades que no son desarrollables, pues no tiene restricciones al respecto.

Considerando las enormes ventajas que representó para el uso de **Isomap**, la introducción por parte de Geng [87] de una matriz de disimilaridad en el manejo del ruido en la construcción del grafo, se propone usar una nueva versión de **GNLM** denominada s -**GNLM** en donde se emplea esta matriz de disimilaridad para la construcción de este grafo. En teoría esto permitirá manejar mayores niveles de ruido y evidenciar mejorías como las que se logran en s -**Isomap**. Entonces el procedimiento asociado a s -**GNLM** puede describirse como

1. Construya un grafo en el espacio \mathbb{R}^N con los puntos $\mathbf{Y} = [\mathbf{y}_1 \dots, \mathbf{y}_P]$.
2. Asigne a cada arista del grafo un peso dado por la matriz de disimilaridad $\mathfrak{D}(\mathbf{y}_i, \mathbf{y}_j)$ entre los vértices del mismo, esta matriz se calcula empleando la ecuación 5.61
3. Se calculan las distancias de grafo entre los puntos del mismo, por ejemplo usando el algoritmo de Dijkstra, y se almacenan en la matriz **D**.
4. Se inicializan los puntos \mathbf{x}_i en el espacio n -dimensional, esta inicialización puede ser aleatoria, o aplicando PCA y tomando los primeros n eigenvectores de acuerdo a la energía asociada a sus autovalores.
5. Se actualiza

$$\mathbf{x}_i(k) \leftarrow \mathbf{x}_i(k) - \alpha \frac{\frac{\partial E_{GNLM}}{\partial \mathbf{x}_i(k)}}{\left| \frac{\partial^2 E_{GNLM}}{\partial \mathbf{x}_i(k)^2} \right|} \quad (5.68)$$

donde de acuerdo con Sammon [80] el valor de α se puede asignar cercano al intervalo entre 0,3 y 0,4.

6. Se computa

$$E_{GNLM} = \frac{1}{c} \sum_{j=2}^P \left(\sum_{i=1}^{j-1} \frac{\left(\tilde{L}_{\mathbf{y}_i}^{y^j} - d_{\mathbf{x}}(i, j) \right)^2}{\tilde{L}_{\mathbf{y}_i}^{y^j}} \right) \quad (5.69)$$

5. Si $E_{GNLM} < \Delta_{err}$ se termina el proceso, de lo contrario se regresa al paso 3. Donde Δ_{err} es el límite del error deseado.

En la figura 5.9 se muestran los resultados de aplicar **GNLM** y s -**GNLM** sobre el conjunto $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7] empleando la totalidad de las expresiones, y como puede verse los resultados son altamente satisfactorios para s -**GNLM**, pues se logra una separación entre los flujos de información correspondientes a cada expresión, sin solapamientos.

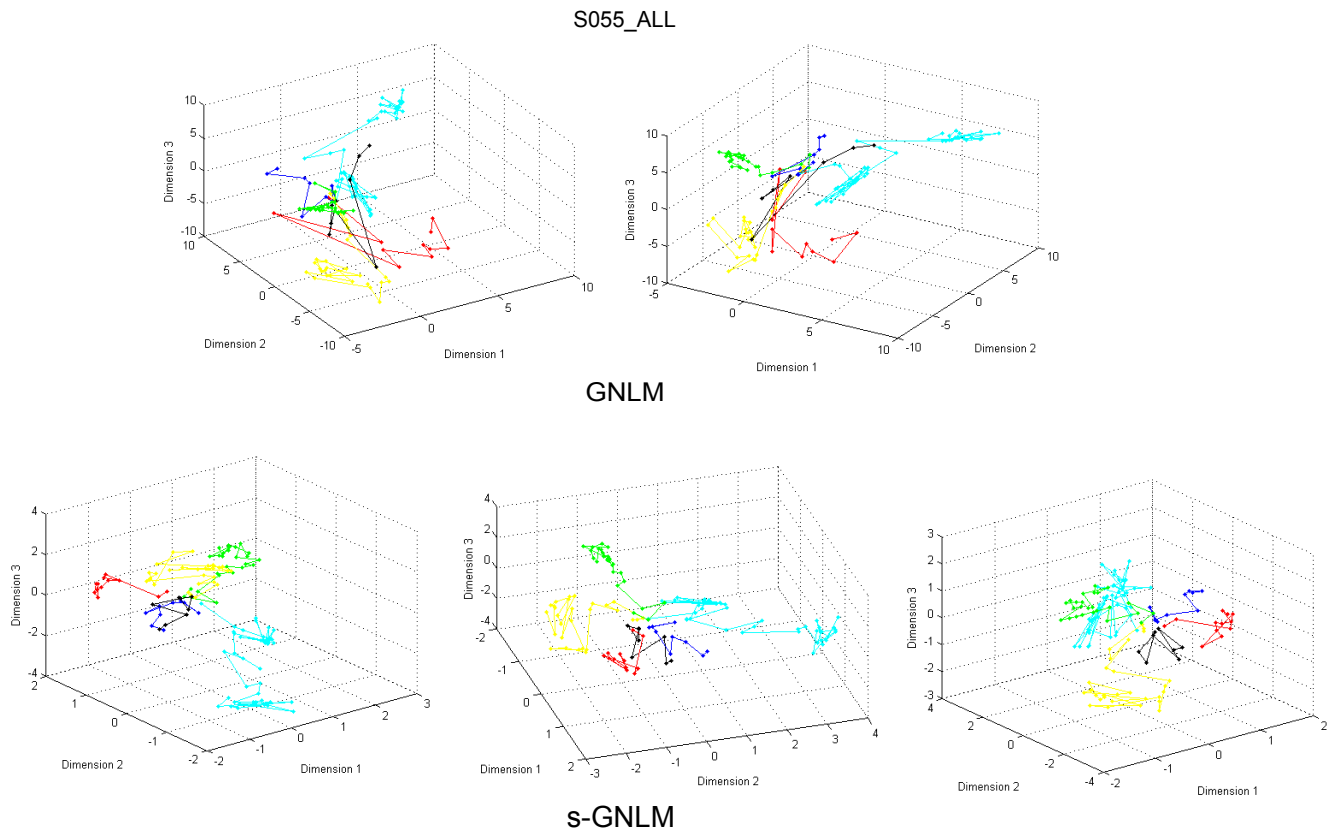


Figura 5.9: Resultados de aplicar **GNLM** y **s-GNLM** sobre $\widehat{\mathcal{M}}_{FSF}$, empleando la dimensión de correlación. Con $K = 12, 11$ respectivamente, seleccionado con el criterio asociado a $f(K)$

Son enormes las fortalezas de s -**GNLM** al permitir manejar mayores niveles de ruido de los que se pueden manejar con **GNLM**, y además las ventajas de **GNLM** en sí mismo permiten dar como una excelente opción para su aplicación en el análisis del comportamiento dinámico de las expresiones faciales. Ésta técnica se contrastará con otras técnicas presentadas.

Curvilinear distance Analysis (CDA)

Lee [88] [89] presenta **CDA** como una versión mejorada de **CCA** en donde se emplea la aproximación de las distancias geodésicas en el espacio en el cual se encuentran los datos en la función de estrés correspondiente a **CCA**. De esta manera la función de estrés asociada a **CDA** se escribe como [88] [89] [58]

$$E_{CDA} = \frac{1}{2} \sum_{\substack{i=1 \\ j=1}} \left(\tilde{L}_{\mathbf{y}_i}^{\mathbf{y}_j} - d_{\mathbf{x}}(i, j) \right)^2 F_{\lambda}(d_{\mathbf{x}}(i, j)) \quad (5.70)$$

y el proceso de actualización se realiza de la misma manera que se planteó para **CCA**. Teniendo en cuenta eso, el algoritmo para **CDA** se presenta como

1. Construya un grafo en el espacio \mathbb{R}^N con los puntos $\mathbf{Y} = [\mathbf{y}_1 \dots, \mathbf{y}_P]$.
2. Asigne a cada arista del grafo un peso dado por la distancia euclidiana entre los vértices del mismo.
3. Se calculan las distancias de grafo entre los puntos del grafo, por ejemplo usando el algoritmo de Dijkstra, y se almacenan en la matriz **D**.
4. Se inicializan las coordenadas en el espacio n -dimensional de todos los puntos \mathbf{x}_i ya sea aleatoriamente o usando PCA. Se hace $q = 1$ (q es el número de época o iteración).
5. Se actualiza el valor de α y λ para el número de época q .
6. Se selecciona un punto \mathbf{x}_j y se actualizan los demás acorde a
`for i=1:1:P`

$$\mathbf{x}_j \leftarrow \mathbf{x}_j - \alpha \nabla_{\mathbf{x}_j} E_{CDA}^i$$

$$\leftarrow \mathbf{x}_j - \alpha \beta(i, j) \frac{\mathbf{x}_i - \mathbf{x}_j}{d_{\mathbf{x}}(i, j)} \quad (5.71)$$

$$(5.72)$$

`end`

7. Retorne al paso 4 hasta que cada punto \mathbf{x}_i haya sido seleccionado al menos una vez por época.
8. Incremente el valor de q , y si la convergencia no se ha logrado retorne al paso 3.

En el paso 6 la actualización se realiza igual que en **CCA**, empleando las ecuaciones 5.53 y 5.54. En tanto los valores de α, λ se actualizan empleando 5.47 y 5.48. Este planteamiento de **CDA** se basa en las ideas originales del método publicado en el año 2000 [88]. No obstante Lee propone en [58] (año 2007) algunos cambios respecto de la actualización e inicialización del valor de λ partiendo de una versión modificada de **CCA** propuesta por Demartines [82] en la que se asocia a

CCA una función de estrés para comportamiento local y otra para comportamiento global [58]. En este trabajo no se adopta este último enfoque, ya que no se introducen cambios sustanciales en la técnica.

Teniendo en cuenta que en esta técnica se requiere de la construcción de un grafo y de la aproximación de las distancias geodésicas sobre el mismo, **CDA** podría poseer limitaciones similares a las asociadas inicialmente a **Isomap** y **GNLM** respecto de la calidad de aproximación de las geodésicas por medio de las distancias de grafo. Por lo tanto es una técnica inadecuada si se cuenta con una baja cantidad y densidad de datos, y adicionalmente existe una dependencia del radio de curvatura de la variedad sobre la cual caen los datos patrón.

Su similaridad con **GNLM** e **Isomap** permite adecuar el planteamiento de Geng [87] a **CDA** como se hizo para presentar s -**GNLM**. Por ello, en este trabajo se propone construir el grafo empleando la matriz de disimilaridad creada por Geng para s -**Isomap**, de esta manera resulta el algoritmo que se denominará en adelante s -**CDA**, el cual describe como

1. Construya un grafo en el espacio \mathbb{R}^N con los puntos $\mathbf{Y} = [\mathbf{y}_1 \dots, \mathbf{y}_P]$.
2. Asigne a cada arista del grafo un peso dado por la matriz de disimilaridad $\mathcal{D}(\mathbf{y}_i, \mathbf{y}_j)$ entre los vértices del mismo, esta matriz se calcula empleando la ecuación 5.61.
3. Se calculan las distancias de grafo entre los puntos del mismo, por ejemplo usando el algoritmo de Dijkstra, y se almacenan en la matriz **D**.
4. Se inicializan las coordenadas en el espacio n -dimensional de todos los puntos \mathbf{x}_i ya sea aleatoriamente o usando PCA. Se hace $q = 1$ (q es el número de época o iteración).
5. Se actualiza el valor de α y λ para el número de época q .
6. Se selecciona un punto \mathbf{x}_j y se actualizan los demás acorde a
`for i=1:1:P`

$$\mathbf{x}_j \leftarrow \mathbf{x}_j - \alpha \nabla_{\mathbf{x}_j} E_{CDA}^i$$

$$\leftarrow \mathbf{x}_j - \alpha \beta(i, j) \frac{\mathbf{x}_i - \mathbf{x}_j}{d_{\mathbf{x}}(i, j)} \quad (5.73)$$

$$(5.74)$$

`end`

7. Retorne al paso 4 hasta que cada punto \mathbf{x}_i haya sido seleccionado al menos una vez por época.
8. Incremente el valor de q , y si la convergencia no se ha logrado retorne al paso 3.

Al igual que en **GNLM** y s -**GNLM** la selección del valor del K óptimo para la construcción del grafo empleando la regla de los K -vecinos se hace usando la propuesta de Shao [86] por las mismas razones expuestas en la presentación de **GNLM** y s -**GNLM**. Los resultados de aplicar **CDA** y s -**CDA** sobre el conjunto $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7] se evidencian en la figura 5.10, y como puede verse también los resultados obtenidos son altamente satisfactorios pues se logra una separación interclase sin solapamiento, y se aprecia una gran similitud entre éstos y los obtenidos usando s -**GNLM**.

Esta técnica se contrastará con otras técnicas presentadas.

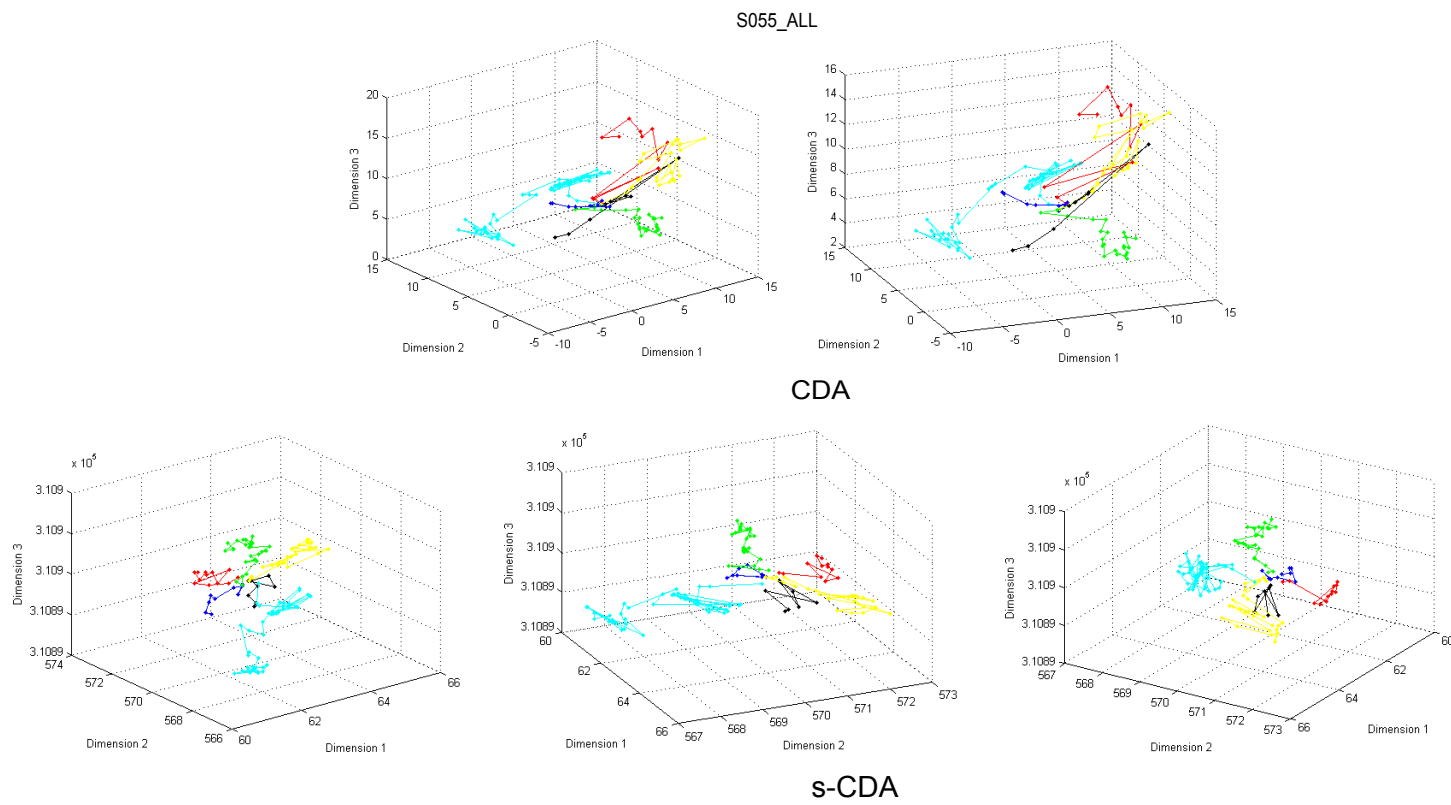


Figura 5.10: Resultados de aplicar **CDA** y **s-CDA** sobre el conjunto $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], usando $K = 12, 11$ respectivamente seleccionado con el criterio asociado a $f(K)$

5.7.3. Otras Técnicas

Una técnica sobre la cual se han realizado un número considerable de aportes es, **KPCA** (*Kernel PCA*). A pesar de su sólida fundamentación matemática y de su potencial capacidad para el manejo de datos que hagan parte de variedades altamente no lineales, posee un serio talón de aquiles para enfrentar problemas con datos de naturaleza arbitraria, la selección del kernel [90] [58]. Acorde a Lee [58] no hay una estrategia clara y consolidada, por el momento, para seleccionar este Kernel a fin de tratar conjuntos arbitrarios de datos. Por estas razones no se empleará esta técnica, pero la misma puede representar para futuro una estrategia valiosa.

Otra técnica destacada, es **SDE** (*semidefinite embedding*). Una fuerte desventaja de este método es la ausencia de un criterio para la determinación del valor de K en la construcción del grafo empleando los K -vecinos, ya que en esta selección juega un papel la isometría local que busca garantizar este método. Por lo tanto, la propuesta de Shao [86] no sería pertinente para aplicarse, porque hace alusión a la calidad de la aproximación de las distancias geodésicas y no a qué tan bien se está preservando dicha isometría local. Sumado a esto, se tiene que incluso para conjuntos pequeños de datos, pero con grandes dimensiones una selección satisfactoria por sintonización de K puede implicar un exagerado costo computacional [58]. Por estas razones no se empleará **SDE** en este trabajo, aunque puede ser una alternativa acorde a los análisis realizados por Lee en [58], valiosa para ser explorada en trabajos futuros cuando se hayan realizado mejoras a la misma.

Grimes desarrolla en [59] una variante de **Isomap** denominada *Piecewise Isomap* en la cual la idea es la aplicación de **Isomap** a nivel local para lograr tratar con mayor éxito los comportamientos locales y ensamblar todas las incrustaciones al final en un resultado global. Sin embargo esta misma propiedad es recogida en **HLLE**, y aunque se aprovecha el comportamiento local de la variedad, **Isomap** es cuestionable para conjuntos reducidos de datos lo cual puede poner muy en duda los resultados obtenidos.

5.8. Técnicas Basadas en Preservación de la Topología

A continuación se presentan y discuten las técnicas basadas en preservación de las propiedades topológicas de las variedad sobre la cual recaen los datos patrón.

5.8.1. Retícula definida por Datos

En esta sección se presentan las técnicas de preservación de la topología más usadas para realizar la reducción de dimensionalidad, usando la contrucción de una retícula de acuerdo al comportamiento de los datos sin restricciones anteriores al uso de los mismos.

Locally Linear Embedding (LLE)

LLE se plantea como un método de preservación de la topología por medio de la conservación de los ángulos a nivel local usando *mapeos conformes* [91] [92]. Éste puede presentarse como

```
1. for i=1:1:P
```

- Encuentre los K -vecinos más cercanos de \mathbf{y}_i
- Calcule la matriz \mathbf{G}_i como

$$\mathbf{G}_i(r, s) = (\mathbf{y}_i - \boldsymbol{\nu}_r)^T (\mathbf{y}_i - \boldsymbol{\nu}_s) \quad (5.75)$$

siendo $\boldsymbol{\nu}_r$ el r -ésimo vecino de \mathbf{y}_i

- Aplique el proceso de regularización

$$\mathbf{G} = \mathbf{G} + \frac{\Delta^2 \text{Tr}(\mathbf{G})}{K} \mathbf{I} \quad (5.76)$$

seleccionando Δ como un valor pequeño en comparación con $\text{Tr}(\mathbf{G})$.

- Se calcula

$$\omega_i^{(r)} = \frac{\sum_{s=1}^K (\mathbf{G}_i^{-1})_{r,s}}{\sum_{r,s=1}^K (\mathbf{G}_i^{-1})_{r,s}} \quad (5.77)$$

end

2. Se construye la matriz \mathbf{W} empleando ω_i que representa los elementos no nulos del renglón i , y además teniendo en cuenta que

- $\mathbf{W}(i, j) = 0$ si \mathbf{y}_i y \mathbf{y}_j no son K -vecinos.

3. Calule la matriz

$$\mathbf{M} = (\mathbf{I} - \mathbf{W})^T (\mathbf{I} - \mathbf{W}) \quad (5.78)$$

4. Se realiza la descomposición en eigenvalores y eigenvectores de la matriz \mathbf{M} , de manera que la representación en el espacio n -dimensional se obtiene tomando los n eigenvectores con menores eigenvalores comenzando desde el segundo más pequeño.

La deducción de las ecuaciones empleadas está relacionada en primera instancia con la minimización de

$$\mathcal{E}(\mathbf{W}) = \sum_{i=1}^P \left\| \mathbf{y}_i - \sum_{j \in \mathcal{K}(i)} \mathbf{W}(i, j) \mathbf{y}_j \right\|^2 \quad (5.79)$$

siendo $\mathcal{K}(i)$ el conjunto de K -vecinos de \mathbf{y}_i . Esta ecuación representa una medida del error asociado a qué tan parecido es \mathbf{y}_i respecto a sus K -vecinos, por lo cual al minimizar $\mathcal{E}(\mathbf{W})$ se espera encontrar los valores $\mathbf{W}(i, j)$ que representen a nivel de estos K -vecinos las propiedades de la variedad sobre la cual recaen los datos patrón [91] [92]. Esta minimización se realiza teniendo en cuenta que

- $\mathbf{W}(i, j) = 0$ si \mathbf{y}_i y \mathbf{y}_j no son K -vecinos.

- $\sum_{j=1}^P \mathbf{W}(i, j) = 1$

Los valores de \mathbf{W} asociados a un determinado punto, tienen la propiedad de ser invariantes a transformaciones afines; de manera que éstos solo representan propiedades geométricas intrínsecas de la variedad [58].

Ahora, empleando los valores de \mathbf{W} se minimiza la función

$$\Phi(\hat{\mathbf{X}}) = \sum_{i=1}^P \left\| \hat{\mathbf{x}}_i - \sum_{j \in \mathcal{K}(i)} \mathbf{W}(i, j) \hat{\mathbf{x}}_j \right\|^2 \quad (5.80)$$

en este caso se busca encontrar los puntos $\hat{\mathbf{x}}_i$ que minimicen la función $\Phi(\hat{\mathbf{X}})$ teniendo en cuenta las propiedades geométricas esenciales de la variedad, sintetizadas en los coeficientes $\mathbf{W}(i, j)$ [91].

Para realizar la minimización de estas funciones se hace primero un calculo directo para los valores de \mathbf{W} teniendo en cuenta que el aporte de un punto \mathbf{y}_i a la función de error $\mathcal{E}(\mathbf{W})$ se puede

escribir como [92]

$$\mathcal{E}_i(W) = \left\| \mathbf{y}_i - \sum_{j \in \mathcal{K}(i)} \mathbf{W}(i, j) \mathbf{y}_j \right\|^2 \quad (5.81)$$

que puede ser reformulado como [58]

$$\mathcal{E}_i(\boldsymbol{\omega}_i) = \left\| \mathbf{y}_i - \sum_{r=1}^K \boldsymbol{\omega}_i(r) \boldsymbol{\nu}(r) \right\|^2 = \left\| \sum_{r=1}^K \boldsymbol{\omega}_i(r) (\mathbf{y}_i - \boldsymbol{\nu}(r)) \right\|^2 = \sum_{r,s=1}^K \boldsymbol{\omega}_i(r) \boldsymbol{\omega}_i(s) \mathbf{G}_i(r, s) \quad (5.82)$$

de manera que los valores de $\boldsymbol{\omega}_i(r)$ pueden ser hallados así [91] [92] [58]

$$\boldsymbol{\omega}_i(r) = \frac{\sum_{s=1}^K (\mathbf{G}_i^{-1})_{r,s}}{\sum_{r,s=1}^K (\mathbf{G}_i^{-1})_{r,s}} \quad (5.83)$$

En ocasiones la matriz \mathbf{G} obtenida es singular o mal condicionada, por lo cual debe aplicarse el proceso de regularización empleando la ecuación 5.76.

Una vez se cuenta con los valores de \mathbf{W} y teniendo en cuenta que $\boldsymbol{\omega}_i$ representa las entradas no nulas del renglón i de la matriz \mathbf{W} , se puede proceder a la minimización de $\Phi(\hat{\mathbf{X}})$ considerando que [91] [92]:

$$\Phi(\hat{\mathbf{X}}) = \sum_{i=1}^P \left\| \hat{\mathbf{x}}_i - \sum_{j \in \mathcal{K}(i)} \mathbf{W}(i, j) \hat{\mathbf{x}}_j \right\|^2 = \sum_{i=1}^P \left\| \sum_{j \in \mathcal{K}(i)} \mathbf{W}(i, j) (\hat{\mathbf{x}}_i - \hat{\mathbf{x}}_j) \right\|^2 = \sum_{i,j=1}^P \mathbf{M}(i, j) (\hat{\mathbf{x}}_i^T \hat{\mathbf{x}}_j) \quad (5.84)$$

de manera que los valores por determinar $\mathbf{M}(i, j)$ corresponden a la matriz

$$\mathbf{M} = (\mathbf{I} - \mathbf{W})^T (\mathbf{I} - \mathbf{W}) \quad (5.85)$$

LLE aprovecha propiedades de tipo local para lograr la reducción de dimensionalidad, por lo cual es una técnica que aprovecha los conceptos esenciales de variedades para preservar la topología. Además de ello, su fundamentación teórica es muy sólida lo cual la convierte en una técnica con grandes atractivos; pero aun así posee una seria desventaja para el análisis automático de datos arbitrarios, la variabilidad de comportamiento respecto a K y Δ [58] [92]. Lee demuestra en [58] que la selección de éstos parámetros debe sintonizarse cuidadosamente para obtener resultados satisfactorios, no existiendo por el momento criterios para la selección óptima de los mismos; y de hecho pueden obtenerse representaciones totalmente distintas para valores de parámetros cercanos entre sí. Este mismo comportamiento se evidencia en la figura 5.11 en donde para un valor dado de Δ se llega a representaciones muy distintas usando diferentes valores de K .

Teniendo en cuenta estos dos últimos aspectos, en este trabajo se reconoce el potencial de esta técnica pero no se usará ya que el análisis que se haga para un tipo particular de información, a fin de seleccionar los parámetros, no será de utilidad para conjuntos de datos arbitrarios.

Los resultados de emplear esta técnica sobre los datos $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7] se pueden apreciar en la figura 5.11 para distintos valores de K . Se observa el comportamiento de la reducción de dimensionalidad que es extremadamente sensible con relación a los cambios en K .

Laplacian Eigenmaps (LE)

Laplacian Eigenmaps se basa en la hipótesis central de que los datos patrón a los cuales se les va a aplicar reducción de dimensionalidad caen o hacen parte de una *variedad Riemanniana* [62]. A partir de esto se emplea la adaptación o aproximación del operador de Laplace-Beltrami en la variedad, sobre grafos [62] [58]. El algoritmo de este método puede presentarse como [62]

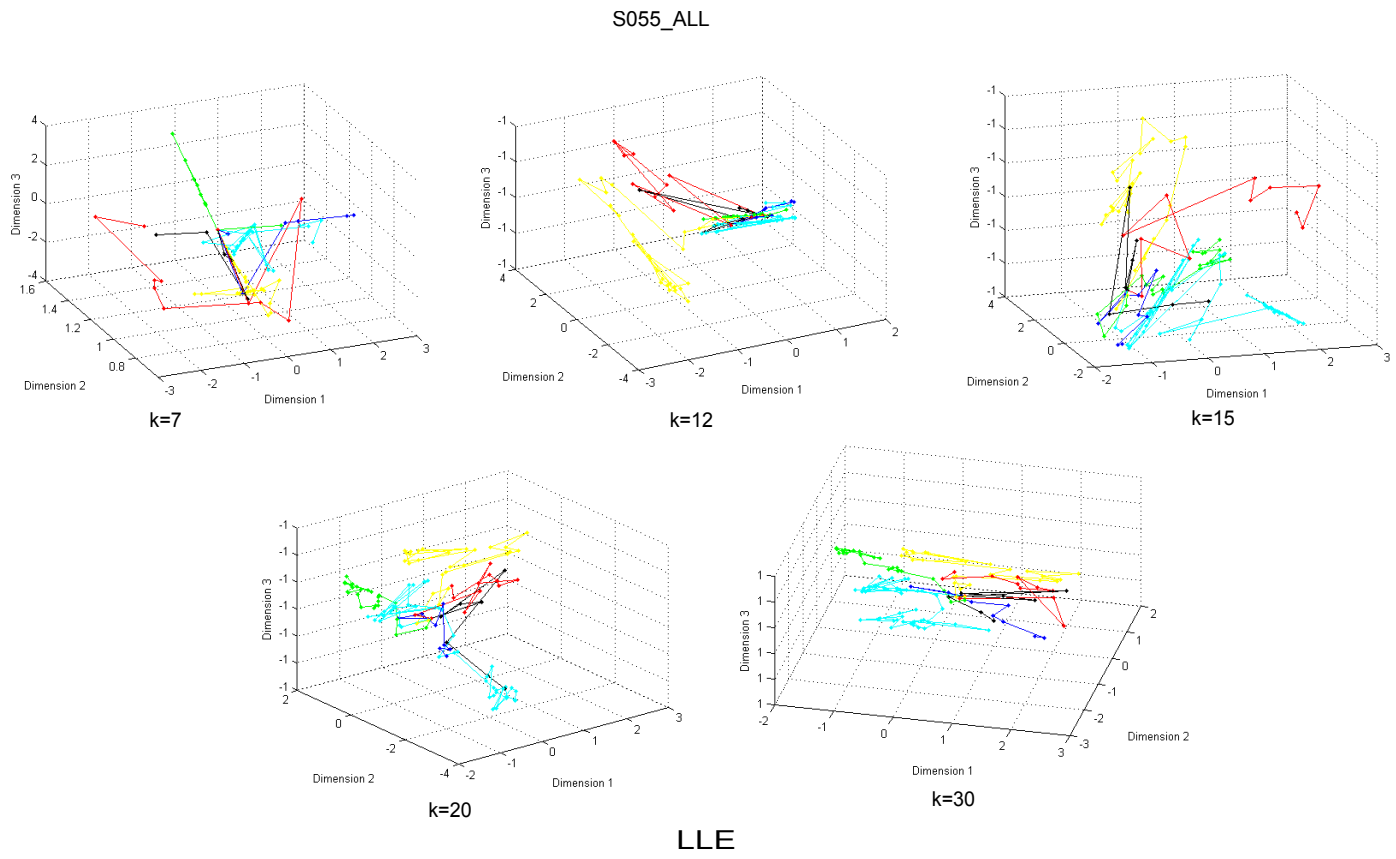


Figura 5.11: Resultados de aplicar **LLE** sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-kanade [7], empleando la dimensión de correlación. Algoritmo de Roweis [91]

1. Compute las distancias entre pares de puntos $d_y(i, j) \forall i, j$.
2. Construya un grafo asignando a cada arista del mismo un peso dado por

$$\mathbf{W}(i, j) = e^{-\frac{d_y^2(i, j)}{t}} \quad (5.86)$$

siendo $t \in \mathbb{R}$ un valor constante asignado.

3. Se calculan las matrices \mathbf{D} y \mathbf{L} como

$$\mathbf{D}(j, j) = \sum_{i=1}^P \mathbf{W}(i, j), \quad \mathbf{D}(i, j) = 0 \quad \forall i \neq j \quad (5.87)$$

$$\mathbf{L} = \mathbf{D} - \mathbf{W} \quad (5.88)$$

en donde \mathbf{L} es llamada la matriz laplaciana, que de acuerdo a Belkin [62] puede ser vista como un operador sobre funciones definidas en los vértices del grafo construido.

4. Se soluciona el problema de eigenvalores y eigenvectores generalizado

$$\mathbf{L}\mathbf{f} = \lambda\mathbf{D}\mathbf{f} \quad (5.89)$$

5. Teniendo en cuenta que los eigenvalores obtenidos son $0 = \lambda_0 \leq \lambda_1, \dots, \lambda_{P-1}$ y sus respectivos eigenvectores $\mathbf{f}_0, \mathbf{f}_1, \mathbf{f}_{P-1}$. Los datos en el espacio n -dimensional se representaran considerando los eigenvectores correspondientes a los eigenvalores $\lambda_1, \dots, \lambda_n$, $n \leq P-1$ como

$$\begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_i \\ \vdots \\ \mathbf{x}_P \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1(1), \mathbf{f}_2(1), \dots, \mathbf{f}_n(1) \\ \mathbf{f}_1(2), \mathbf{f}_2(2), \dots, \mathbf{f}_n(2) \\ \vdots \\ \mathbf{f}_1(i), \mathbf{f}_2(i), \dots, \mathbf{f}_n(i) \\ \vdots \\ \mathbf{f}_1(P), \mathbf{f}_2(P), \dots, \mathbf{f}_n(P) \end{bmatrix} \quad (5.90)$$

En el paso 4 en donde se soluciona el problema generalizado de eigenvalores y eigenvectores es posible realizar, si se desea, una normalización de \mathbf{L} [58], como

$$\mathbf{L}' = \mathbf{D}^{-\frac{1}{2}} \mathbf{L} \mathbf{D}^{-\frac{1}{2}} \quad (5.91)$$

y así el problema se trasladaría al de hallar los eigenvalores y eigenvectores de \mathbf{L}' , teniendo en cuenta que los eigenvalores son los mismos que los del problema generalizado y los nuevos eigenvectores \mathbf{u}_i guardan la siguiente relación con los originales [58]

$$\mathbf{u}_i^T = \mathbf{D}^{\frac{1}{2}} \mathbf{f}_i \quad (5.92)$$

Lee en [58] y Belkin en [62] muestran que el valor de los parámetros t en el kernel de la función empleada en la construcción del grafo y el valor de K empleado en la regla de los K -vecinos juega un papel dramático en los resultados de la reducción de la dimensionalidad. Actualmente no existen procedimientos o reglas que permitan seleccionar un valor óptimo para los mismos, y ésta selección debe hacerse por sintonización [58].

Por otra parte, Lee [58] también demuestra que el planteamiento de $\mathbf{L}\mathbf{E}$ puede reformularse de

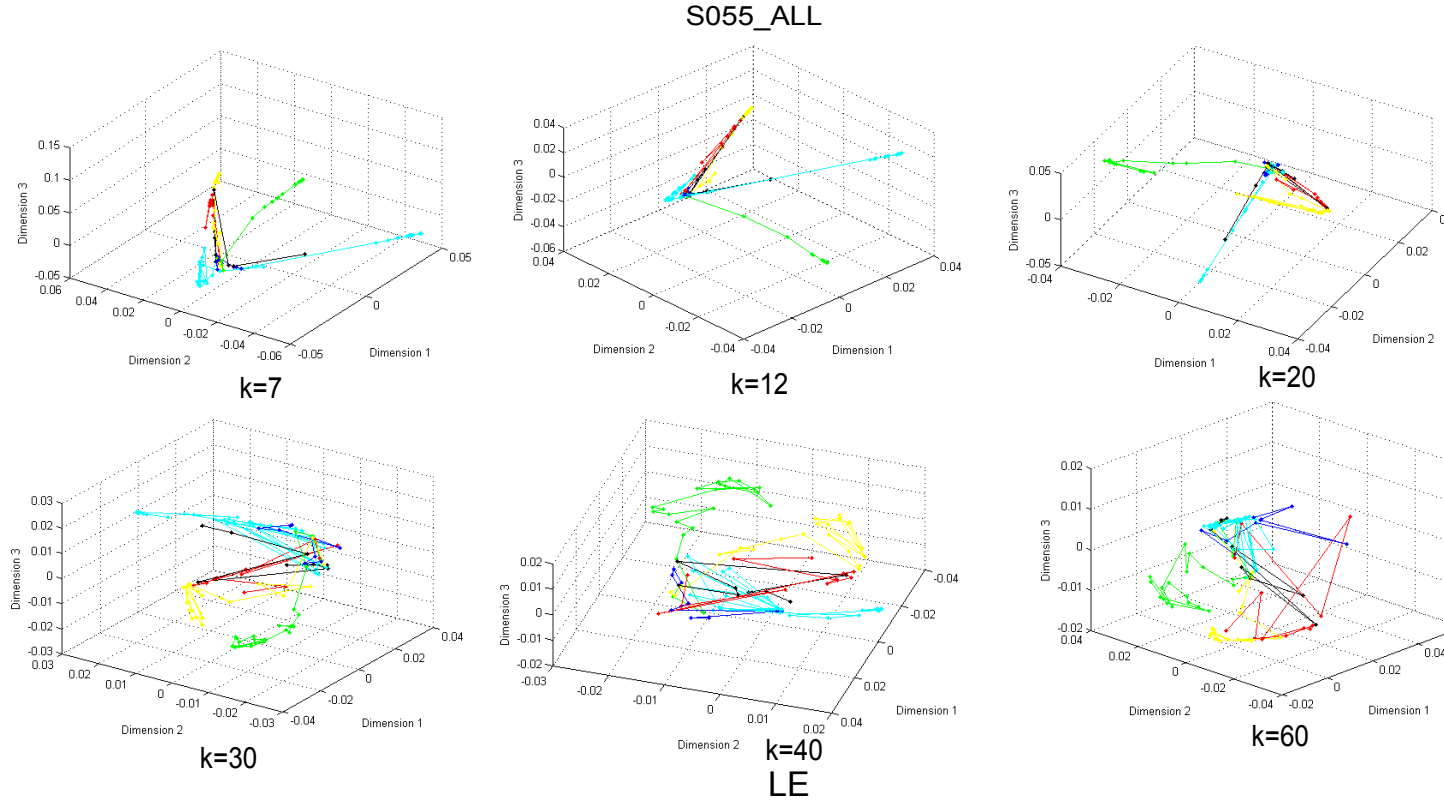


Figura 5.12: Resultados de aplicar **LE** sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kande [7], empleando la dimensión de correlación. Algoritmo de Belkin [62]

manera equivalente al de minimizar la función de estrés

$$E_{LE} = \frac{1}{2} \sum_{i,j=1}^P \mathbf{W}(i,j) \|\mathbf{x}_i - \mathbf{x}_j\|^2 \quad (5.93)$$

la cual evidencia minimización de distancias entre vecinos, que puede conducir a comprimir excesivamente los datos llevando a encajamientos pobres, pero esto a su vez convierte esta técnica en conveniente para *clustering* de datos [58] [62]. Estos aspectos convierten a **LE** en una técnica con una fundamentación que permitiría abrir nuevas rutas en la investigación sobre reducción de dimensionalidad, pero altamente inadecuada. Además en la figura 5.12 se evidencian los pobres resultados obtenidos con los datos empleados en este trabajo asociados a la variedad \mathcal{M}_{FSF} para el sujeto S055 de la base de datos Cohn-Kanade [7].

Adicionalmente, un aspecto esencial encontrado por Lee [58] en esta técnica es que **LE** tiende a curvar la variedad en el espacio de dimensión intrínseca, lo que produce fuertes cuestionamientos sobre la misma.

Con todos estos argumentos, en este trabajo no se empleará **LE**. Los resultados de aplicar esta técnica al conjunto $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 se pueden apreciar en la figura 5.12, como se ve los resultados no son satisfactorios.

Hessian Eigenmaps (HLE)

Donoho y Grimes en [59] [61] desarrollan **HLE** como un método intermedio entre **LLE** y **LE**. Del primer método toma la idea de preservar una propiedad a nivel local, y de **LE** el hecho de basar la

descripción de esta propiedad por medio de la estimación de una cantidad asociada a la variedad, el Hessiano. El desarrollo de esta técnica se basa en la hipótesis de que los datos caen sobre una *variedad Riemanniana*, y se busca lograr una isometría a nivel local. El algoritmo de esta técnica puede presentarse como [61]

1. Para cada punto \mathbf{y}_i encuentre los K -vecinos más cercanos $\mathcal{K}(i)$.

2. Se construyen las matrices $\mathcal{K}^{(i)}$

for $i=1:P$

$$\mathcal{K}^{(i)}(:, j) = \mathbf{y}_j - \left(\frac{1}{|\mathcal{K}(i)|} \sum_{r=1}^{|\mathcal{K}(i)|} \mathbf{y}_r \right) \quad (5.94)$$

end

siendo $|\mathcal{K}(i)|$ la cantidad de elementos en $\mathcal{K}(i)$.

3. Se realiza la descomposición en valores singulares para cada matriz $\mathcal{K}^{(i)}$

$$\mathcal{K}^{(i)} = \mathbf{U} \mathbf{D} \mathbf{V} \quad (5.95)$$

4. Empleando la anterior descomposición se construyen las matrices

$$\mathcal{X}^{(i)} = [\mathbf{1}, \mathbf{U}, \mathbf{u}_1^2, \dots, \mathbf{u}_n^2, \mathbf{u}_1 \mathbf{u}_2, \dots, \mathbf{u}_r \mathbf{u}_{r+s}, \dots], \quad r = 1, \dots, n \quad s = 1, \dots, n - r \quad (5.96)$$

5. Se realiza el proceso de ortonormalización de Gram-Schmidt sobre las columnas de $\mathcal{X}^{(i)}$ obteniendo $\tilde{\mathcal{X}}^{(i)}$.

6. El estimado del Hessiano para $\mathcal{K}(i)$ se obtiene como

$$\mathcal{H}^{(i)} = \left[\tilde{\mathcal{X}}^{(i)}(:, n+1 : \rightarrow) \right]^T \quad (5.97)$$

es decir la transpuesta de $\tilde{\mathcal{X}}^{(i)}$ considerando los datos solamente desde la columna $n+1$

7. Se ensamblan estos estimados en la matriz \mathbf{W} como

$$\mathbf{W} \left((i-1) \frac{n(n+1)}{2} + j, \zeta_i(2 : K+1) \right) = \mathcal{H}^{(i)}(j, :) \quad (5.98)$$

en donde el vector ζ_i contiene el orden de cercanía de todos los puntos al punto i . Así $\zeta_i(2 : K+1)$ indica los K puntos más cercanos al i -ésimo punto. La matriz \mathbf{W} contiene ceros en las entradas no especificadas.

8. Se calculan los eigenvectores de la matriz $\mathbf{G} = \mathbf{W}^T \mathbf{W}$ correspondientes a los $n+1$ eigenvalores más pequeños $0 \leq \lambda_0 \leq \dots \lambda_n$. Con los eigenvectores $\mathbf{V}_G = [\mathbf{v}_{g1}, \dots, \mathbf{v}_{gn}]$ asociados a los eigenvalores $\lambda_1, \dots, \lambda_n$ se calcula

$$\mathbf{R} = \mathbf{P} \mathbf{V}_G \mathbf{V}_G^T \quad (5.99)$$

9. La representación en el espacio n -dimensional se obtiene como

$$\mathbf{X} = \sqrt{\mathbf{P}} \mathbf{V}_G^T \mathbf{R}^{-\frac{1}{2}}$$

En el paso 4 es necesario tener en cuenta que

$$\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_n] \quad (5.100)$$

Grimes [59] considera el problema de reducción de dimensionalidad como un problema de reparametrización. Es decir si \mathcal{M} es la variedad de dimensión n encajada en \mathbb{R}^N sobre la cual caen los datos $\mathbf{y}_1, \dots, \mathbf{y}_P$, se asume que existe un conjunto abierto $\mathcal{N} \subset \mathbb{R}^n$ y que además existe $\psi : \mathcal{N} \rightarrow \mathcal{M}$, tal que $\psi(\mathcal{N}) = \mathcal{M}$ siendo ψ un encajamiento suave localmente isométrico¹. El problema se plantea entonces como el de encontrar un homeomorfismo $f : \mathcal{M} \rightarrow \mathcal{N}$, tal que se preserven las propiedades esenciales de la variedad. De acuerdo a Grimes [59] es posible encontrar una base para dicha representación analizando el funcional

$$\mathcal{H}(f) = \int_{\mathcal{M}} \|H_f(m)\|^2 dm \quad (5.101)$$

que brinda una medida de la curvatura promedio inducida² por \mathcal{M} sobre f . De hecho Grimes [59] [61] demuestra como $\mathcal{H}(f)$ posee un espacio nulo de $n + 1$ dimensiones, lo cual implica que es posible recuperar la estructura original de los datos (e.g recuperar \mathcal{N}) salvo por un movimiento rígido.

Una clara fortaleza de esta técnica es el aprovechamiento de una propiedad de tipo local, y su sólida fundamentación en conceptos precisos sobre variedades Riemannianas, pero por otra parte evidencia una de las mismas debilidades de **LLE**, la selección de un valor óptimo de K para la construcción del grafo empleando la regla de los K -vecinos. Como se ve en la figura 5.13, en donde se aplica **HLLE** sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], diferentes valores de K incluso cercanos entre sí, pueden llevar a resultados completamente diferentes.

Por otra parte en [61] [59] se evalúa el comportamiento de **HLLE** con respecto a **Isomap**, **LLE** y **LE** dejando entrever la superioridad de **HLLE** para variedades no desarrollables y no convexas.

Teniendo en cuenta estos aspectos, en este trabajo se considera que aunque **HLLE** es una técnica promisoría con una sólida fundamentación que puede ser extremadamente útil para análisis no automáticos de la información, presenta una limitante para la selección de un valor óptimo de K que resulte adecuado a los datos que se emplearán en este trabajo, los cuales son en general escasos, dispersos, y de naturaleza cambiante de un sujeto a otro. Por todo ello no se empleará.

Otras Técnicas

Lee y Verleysen proponen en [94] [95] **Isotop** como una técnica de preservación de la topología en la que se emplean, al igual que en métodos anteriores, las distancias de grafo a fin de aproximar las distancias geodésicas. Sin embargo, la forma en como se construye la representación n -dimensional es totalmente nueva, ya que una vez construido el grafo con la información en el espacio de representación original se busca actualizar una inicialización de los puntos \mathbf{x}_i asignados a cada vértice del grafo, teniendo en cuenta la aproximación de las geodésicas y usando Kernels Gaussianos asociados a cada \mathbf{x}_i . En [58] los resultados obtenidos por Lee son los mejores para un conjunto de variedades *benchmark* en comparación con otros métodos, sin embargo no hay un suficiente nivel de detalle como para realizar una implementación de esta técnica. El autor de esta investigación contactó directamente con el doctor Lee [58], quien facilitó un código desarrollado en febrero de 2010, con algunas modificaciones substanciales. Por ejemplo en éste ya no se realiza la construcción de un grafo y se integran las ideas de técnicas como *SNE* (*stochastic neighbor embedding*). El doctor Lee [58] afirma que el nuevo código representa un algoritmo de muchísima más calidad, y con resultados superiores a los que se obtienen con el algoritmo original. Sin embargo, en virtud de que no existen publicaciones en las cuales se plasmen claramente las diferencias de este nuevo algoritmo, con el que se presenta en [58], en esta investigación no se realiza el uso de esta técnica. No habría coherencia en hacer alusión a un marco teórico bien definido, pero representado en un código que no sigue estrictamente estas ideas. Se puede afirmar que para trabajos futuros esta técnica muestra ser promisoría, en tanto se publiquen en detalle las modificaciones realizadas usando *SNE* al código

¹ver [93]

²En [61] se usa la palabra *curviness* para referirse a esto

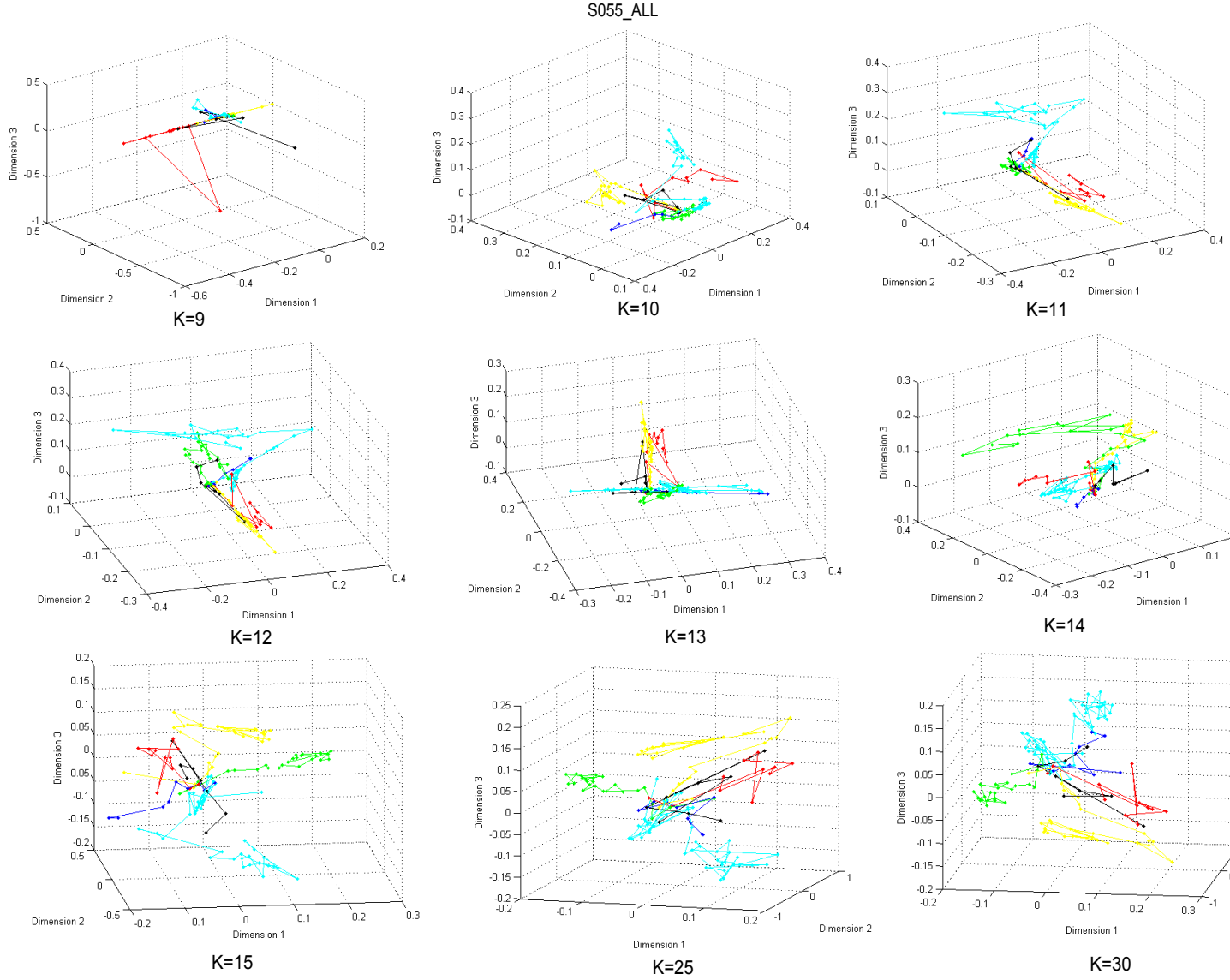


Figura 5.13: Resultados de aplicar **HLLS** sobre $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], empleando la dimensión de correlación. Algoritmo de Grimes [59]

original de **Isotop**.

Zhang presenta en [96] **LTSA**, como una técnica que busca aprovechar el concepto de espacio tangente a una variedad diferencial, considerando los puntos originales con un nivel de ruido asociado. En ésta se busca representar la geometría local usando espacios tangentes construidos ajustando un subespacio afín en una vecindad de cada punto. Dichos espacios tangentes son alineados y a partir de una descomposición espectral se propone el encajamiento en el espacio de dimensión intrínseca. En virtud de esto, posee una enorme cercanía con **LLE** y **HLLE**, pero así mismo comparte con éstas la desventaja de que no existe un criterio automático para conjuntos arbitrarios de datos, que permita seleccionar el valor de K en la selección de los K -vecinos más cercanos para la construcción de los espacios tangentes. Acorde a los resultados obtenidos en [96] esta selección juega un papel crucial y es particularmente difícil si los datos están contaminados con ruido y/o si la variedad sobre la cual recaen los datos posee un gran radio de curvatura. Por estas razones no se empleará **LTSA**, sin embargo se deja claro que es una de las técnicas con mejor fundamentación, y sobre la cual se han indicado de manera precisa limitaciones y posibles mejoras, por esto puede representar en un futuro una alternativa a considerar.

Brun propone en [97] **LogMaps** como una técnica para la reducción de dimensionalidad. Ésta se basa en la representación de la información por medio de coordenadas Riemannianas normales [93]. Aunque su fundamentación teórica es sólida, posee una seria desventaja para el manejo de ruido dentro de los datos, puesto que la misma solo ha sido probada con éxito con niveles de ruido prácticamente nulos [97]. Por otra parte, para la construcción del sistema de coordenadas normales se requiere de la selección de un punto sobre la variedad, y esta selección juega un papel crítico en los resultados. Actualmente no existe un criterio para la localización adecuada de dicho punto para un conjunto arbitrario de datos. Por ello esta técnica no se empleará en esta investigación, pero se presenta como una técnica promisoría para futuros análisis del comportamiento dinámico de las expresiones faciales.

5.9. Comparación de los Métodos

En esta sección se especifican las técnicas que serán empleadas en el esquema de reducción de dimensionalidad general. Se hace alusión a aspectos discutidos en secciones anteriores y se retomarán brevemente los comentarios adecuados respecto a cada selección.

5.9.1. Estimación de la Dimensión

Teniendo en cuenta los argumentos teóricos correspondientes a cada una de las técnicas presentadas para realizar la estimación de la dimensión, y los resultados experimentales obtenidos se afirma que los métodos más adecuados para los fines de esta investigación son

- **Dimensión de Correlación**. Procede de una definición formalmente construida en el campo de la geometría fractal, ha sido ampliamente usada con resultados satisfactorios [58] [69], y además de ello existe un criterio establecido de manera rigurosa (*la regla del cinco* [76]) para definir sus parámetros.
- **PCA** – ℓm . Aprovecha las ventajas esenciales de **PCA**– ℓ , pero se adecua bien a conjuntos de datos reducidos. Teniendo en cuenta el criterio propuesto basado en la *búsqueda del código de la curva AIC* se puede afirmar que existe un criterio bien definido para la selección del número de eigenvalores.
- **PCA** – ω . Considera la interpretación del teorema de Whitney cuando el número de variables asociadas a la descripción original del patrón supera enormemente la dimensión intrínseca estimada por correlación y/o **PCA** – ℓm . Brinda un estimado teniendo en cuenta que existe un número máximo de variables necesario para la representación, por encima del cual existe redundancia que puede ser fácilmente eliminada por **PCA**. Los resultados experimentales

muestran su cercanía a los valores de las otras dimensiones, y al usar el criterio basado en la *búsqueda del codo de la curva AIC* se puede garantizar que existe un criterio bien definido para la selección de los eigenvalores.

Para los fines de estimación de la dimensión se emplearán estas tres técnicas, dando como valor final estimado un valor promedio de éstas, redondeado al entero más cercano.

5.9.2. Reducción de Dimensionalidad

Teniendo en cuenta los argumentos teóricos correspondientes a cada una de las técnicas presentadas anteriormente, y los resultados experimentales obtenidos sobre los datos $\widehat{\mathcal{M}}_{FSF}$ para el sujeto S055 de la base de datos Cohn-Kanade [7], se puede afirmar que los métodos con mejores resultados y características en reducción de dimensionalidad para los propósitos de esta investigación son

- **s -Isomap.** Aunque no se puede garantizar que la variedad sobre la cual recaen los datos sea desarrollable, la posible distorsión del espacio original es aceptable siempre y cuando no haya solapamiento de información. Además existe un criterio para la selección óptima del valor de K en la construcción del grafo empleando la regla de los K -vecinos. En caso de considerarse varias *clases* sobre una misma variedad, al usar **s -Isomap** se logran manejar niveles de ruido considerables.
- **s -GNLM.** Permite manejar la no linealidad por medio de las distancias de grafo y la minimización de una función de estrés. No produce deformaciones del mismo grado que se producen por **Isomap** en variedades no desarrollables. Y posee plenas capacidades para reducir dimensiones sobre variedades con *nudos y lazos esenciales*, acorde a los experimentos realizados por Lee en [58]. La misma regla empleada en **Isomap** para la selección óptima del valor de K en la construcción del grafo aplica para este método. El uso de la matriz de disimilaridad de Geng [87] permite manejar niveles de ruido considerables.
- **s -CDA.** Permite manejar la no linealidad por medio de las distancias de grafo y la minimización de una función de estrés. No produce deformaciones del mismo grado que se producen en **Isomap**. Según los resultados de Lee en [58], la deformación es incluso menor a la producida por **GNLM**. Además posee incluso mejores capacidades que **GNLM** para reducir dimensiones sobre variedades con *nudos y lazos esenciales* [58]. La misma regla empleada en **Isomap** para la selección óptima del valor de K en la construcción del grafo aplica para este método. El uso de la matriz de disimilaridad de Geng [87] permite manejar niveles de ruido considerables.

Los resultados obtenidos sobre los datos $\widehat{\mathcal{M}}_{FSF}$ usando estas técnicas, fueron los mejores dentro de las métodos presentados, puesto que se logró la mejor separación interclase sin solapamiento de información en un espacio euclidiano de la dimensión intrínseca estimada.

5.10. Esquema propuesto para la Reducción de Dimensionalidad

Teniendo en cuenta, como se mencionó en secciones anteriores, que la baja densidad y cantidad de datos puede hacer cuestionables los resultados de la aplicación de técnicas de reducción de dimensionalidad, el esquema planteado considera que los conjuntos $\widehat{\mathcal{M}}_{FSF}, \widehat{\mathcal{M}}_{FST}, \widehat{\mathcal{M}}_{FIF}, \widehat{\mathcal{M}}_{FIS}, \widehat{\mathcal{M}}_{FIT}$ están formados por los datos correspondientes a todas las expresiones de un determinado sujeto, es decir la variedad sobre la cual recaen estos puntos es una variedad *interclase*.

Sea $\widehat{\mathbf{F}}_{\nu, \hat{n}}^{i, \xi}$ el arreglo de los datos obtenidos después de la reducción de dimensionalidad sobre los *features* $\widehat{\mathcal{M}}_{\nu}$, $\nu = FSF, FST, FIS, FIF, FIT$, empleando la técnica $\xi = s\text{-CDA}, s\text{-GNLM}, s\text{-Isomap}$ correspondiente al sujeto i . El esquema propuesto se puede apreciar en la figura 5.14.

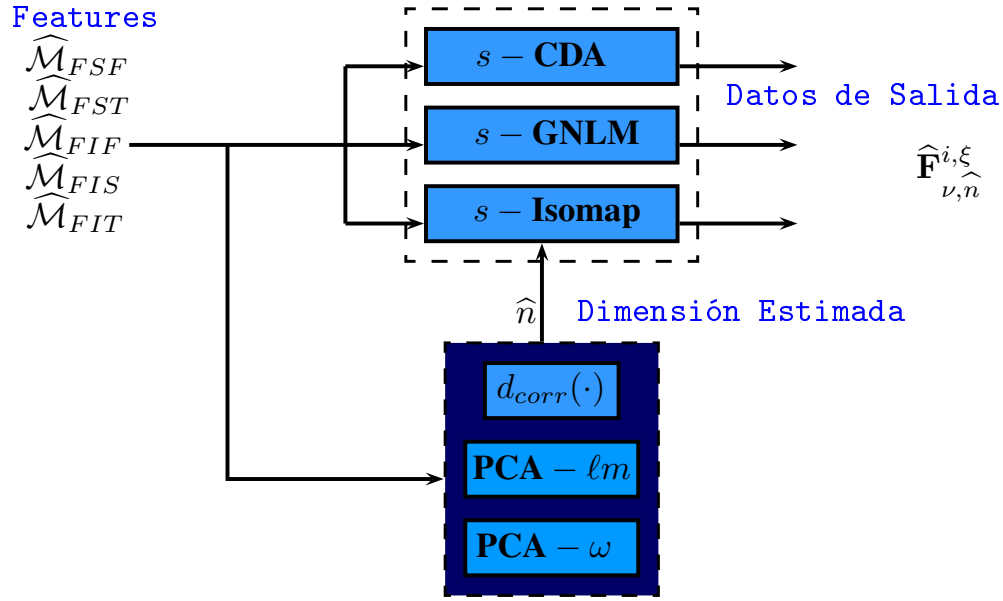


Figura 5.14: Esquema General de Reducción de Dimensionalidad

De esta manera la información total de salida del esquema de reducción de dimensionalidad es un arreglo de $15i_T$ elementos cada uno de los cuales es un vector de dimensión \widehat{n} . Siendo i_T la cantidad total de sujetos empleados en el proceso de extracción de información. Estos datos tendrán una dimensión \widehat{n} que no necesariamente será igual para todos los individuos, a menos que se trate de expresiones faciales con patrones muy similares entre sí.

5.11. Implementación

5.11.1. Estimación Dimensión Intrínseca

La dimensión de correlación se calcula empleando parte del código desarrollado por Maaten [75], mientras que las demás técnicas son desarrollos propios del autor de esta investigación, realizados en Matlab.

5.11.2. Reducción de Dimensionalidad

Para realizar la implementación de las técnicas propuestas para la reducción de dimensionalidad se usa la herramienta de simulación Matlab. El código de $s - \text{CDA}$ es construido empleando parte del código desarrollado por Tenenbaum [84] para la construcción del grafo con cálculos relacionados y empleando parte del código desarrollado por Vesanto [83] dedicado a CCA . Para computar la matriz de disimilaridad empleada se usa el código de desarrollado por Geng [87].

En la construcción del código de $s - \text{GNLM}$ se emplea parte de un programa desarrollado por Vesanto [81] para el cómputo de NLM . Mientras que el código desarrollado por Tenenbaum [84] es empleado para el manejo del grafo con cantidades relacionadas. Al igual que en $s - \text{CDA}$ se emplea el código de Geng [87] para el cálculo de la matriz de disimilaridad.

El código de $s - \text{Isomap}$ empleado es desarrollado por Tenenbaum [84] y adecuado por Geng [87] para trabajar con la matriz de disimilaridad. En este trabajo se realizó la una revisión detallada del mismo, logrando así realizar una corrección en este código en los umbrales establecidos para definir la matriz de disimilaridad.

5.12. Discusión y Conclusiones

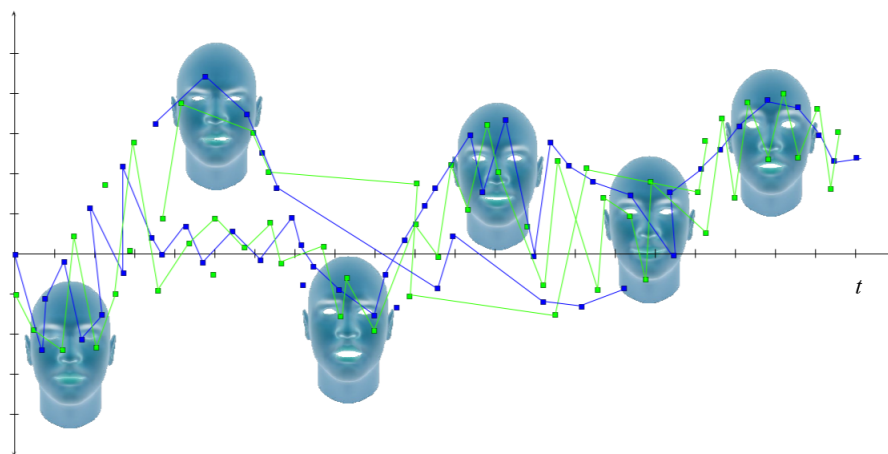
En este capítulo se abordó el problema de determinar los métodos más pertinentes y/o adecuados para realizar la reducción de dimensionalidad en el seguimiento automático de las expresiones faciales en una secuencia de vídeo. Se realizó un análisis teórico de las técnicas más empleadas en el análisis de información facial, teniendo en cuenta desarrollos analíticos y resultados experimentales de otros autores, así como resultados y consideraciones propias que dan como consecuencia el planteamiento del esquema de la figura 5.14.

Algunas técnicas no seleccionadas, cuentan con características que las hacen promisorias para trabajos futuros en tanto se sigan realizando investigaciones sobre éstas.

Además, como parte de la contribución propia de este trabajo de investigación se realizó una propuesta para estimar la dimensión intrínseca teniendo en cuenta el teorema de Whitney para variedades diferenciales y un criterio basado en la búsqueda del *codo* de la curva *AIC*.

6

Resultados Generales



En este capítulo se da una presentación de los códigos entregados como productos asociados a los resultados del proyecto de investigación. Se discutirán a grandes rasgos los principales archivos, su manejo y datos de salida dejando la presentación de los mismos para los apéndices. Adicionalmente se delimita el escenario de aplicación de este desarrollo, para así puntualizar las alcances del mismo, y los aspectos a mejorar y/o desarrollar en trabajos futuros.

6.1. Escenario de Aplicación

Los resultados de este proyecto de investigación, están asociados al seguimiento dinámico de la información que representa las expresiones faciales, dentro de un escenario dado por los alcances de las herramientas desarrolladas.

El esquema de detección de rostro propuesto aprovecha las fortalezas de *Adaboost*, teniendo en cuenta en el caso de las imágenes a color, regiones más acotadas dadas por la segmentación cromática propuesta. Y si bien el detector de rostro de Viola y Jones [10] es una solución contundente para condiciones controladas, no lo es para condiciones extremas y arbitrarias de iluminación. Por lo tanto

1. *El software desarrollado se aplica a vídeos o secuencias de imágenes en ambientes controlados de iluminación adecuados para el estudio y análisis de expresiones faciales*

Los *Active Appearance Models (AAM)* bajo los argumentos presentados en el capítulo 3, brindan la mejor opción para la localización de puntos faciales. Ésta debe estar asociada a conjuntos de imágenes de sujetos cuya correlación con las imágenes de entrenamiento sea alta. Por ello una de las características del escenario de aplicación es la siguiente

2. *El software desarrollado se aplica para realizar el seguimiento de expresiones faciales en vídeos o secuencias de imágenes, asociadas a sujetos de los que se cuenta con algunas imágenes para realizar el entrenamiento del mismo*

es decir se debe contar con unas cuantas imágenes del sujeto para el entrenamiento, y la herramienta permitirá realizar el seguimiento dinámico de las expresiones de este mismo sujeto en secuencias de vídeo arbitrarias.

Es particularmente importante, aclarar que este mismo escenario es semejante al de entrenar un sistema para lograr el reconocimiento facial.

Por otra parte, el análisis de esta información no se realiza en tiempo real. A pesar de que la extracción de la información se realiza *frame por frame*, el tratamiento en la reducción de dimensionalidad se realiza con la totalidad de la información extraída de la secuencia de vídeo. Este hecho permite, dar más confiabilidad a los métodos de reducción de dimensionalidad como se mencionó en el capítulo 5, y con ello más certeza de las conclusiones extraídas a partir de este seguimiento. Esto se sintetiza como

3. *El software desarrollado se aplica para extraer conclusiones de una secuencia de vídeo realizando un análisis y procesamiento de los frames en bloque*

No existen restricciones sobre la cantidad de datos mínima o máxima sobre la cual puede aplicarse el esquema de reducción de dimensionalidad desarrollado en el capítulo 5. No obstante, debe tenerse en cuenta que uno de los aspectos que brinda confiabilidad en la aplicación de cualquiera de estas técnicas es la cantidad y densidad de datos con los que se cuenta. Considerando los aspectos discutidos en el capítulo 5 se plantea que

4. *Una medida de la confiabilidad de los resultados finales en la reducción de dimensionalidad, está dictada por la cantidad de secuencias de vídeo analizadas por sujeto, la cantidad de frames de los mismos y la tasa de muestreo. Entre mayores sean estas cantidades mucho más confiables serán los resultados obtenidos*

6.2. Productos Entregados

6.2.1. Artículos de Divulgación

1. **Fast Object Detection using Colour Segmentation.** Código ISSN: 2145-812X. Tercer Congreso Internacional de Ingeniería Mecatrónica. Bucaramanga 2011.
2. **Face Detection using Adaboost and Color Centroid Segmentation with Comprehensive Color Image Normalization.** Aceptado para revisión en Diciembre de 2010. Aún en espera de las indicaciones para su envío definitivo a la revista de Ingeniería y Ciencia de la Universidad EAFIT.
3. **A New Facial Point Detector using Active Appearance Models.** Código ISSN: 2145-812X. Tercer Congreso Internacional de Ingeniería Mecatrónica. Bucaramanga 2011.
4. **Intrinsic Dimensionality Estimation of Data using PCA and The Whitney Theorem.** En Construcción.
5. **Supervised Nonlinear Dimensionality Reduction over the Manifold of Facial Expressions with GNLM and CDA.** En Construcción.

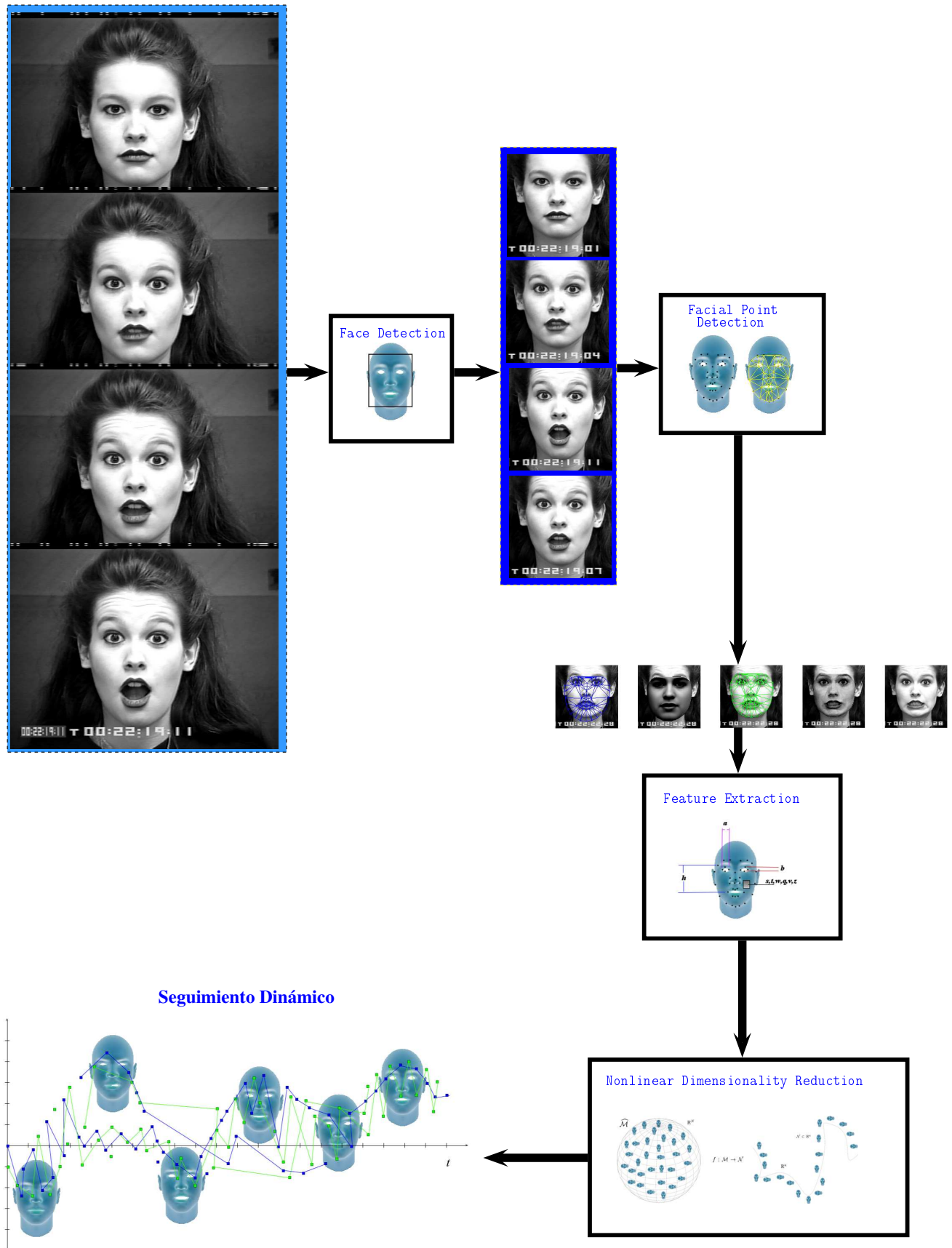


Figura 6.1: Esquema General de Seguimiento Dinámico

6.2.2. Códigos

En esta sección se brinda una descripción de los códigos desarrollados, así como de sus principales características y parámetros a fin de realizar el seguimiento dinámico de la información.

6.2.3. Detección de Rostro

Para hacer la detección de rostro en un conjunto de imágenes correspondientes a una secuencia de vídeo, se emplea el siguiente código

- ▲ `my_facedetect6.m` En el cual se ejecutan los algoritmos de segmentación cromática, y el programa realizado en C++ basado en *Adaboost* para la detección por textura.
- ▲ `y8.cpp` Código en C++ basado en *Adaboost* que se ejecuta dentro de `my_facedetect6.m`.

de esta manera es posible realizar la detección de rostro sobre la pila de *frames* que conforman la secuencia de vídeo. Entregando como salida los archivos de imagen con la detección realizada.

6.2.4. Localización de Puntos Faciales y Extracción de Características

Para la localización y extracción de puntos faciales se ejecuta el archivo `my_searchvideo.m`, en el cual a su vez se ejecutan los archivos:

- ▲ `my_SearchModel_2.m` Con el cual se realiza el proceso de búsqueda en la imagen objetivo por medio los *AAM*, y
- ▲ `extract_data.m` Con el cual se realiza la extracción de la información correspondiente a cada uno de los *frames*.

6.2.5. Reducción de Dimensionalidad

Para realizar la reducción de dimensionalidad se ejecuta el código `my_NDR.m`, en donde a su vez se ejecutan los siguientes archivos

- ▲ `my_corrdim.m` (Adaptación de Maaten [75]) Empleado para calcular la dimensión intrínseca de los datos por medio de correlación.
- ▲ `my_localPCA.m` Empleado para calcular la dimensión intrínseca de los datos usando **PCA**– ℓm .
- ▲ `my_AIC_MDL.m` Empleado para seleccionar el número de eigenvectores en la estimación de la dimensión intrínseca usando **PCA**.
- ▲ `my_KoptIsomap.m` Empleado para calcular el valor óptimo de K en la construcción del grafo empleando la regla de los K –vecinos, en técnicas en las que se aproximan las distancias geodésicas por medio de las distancias de grafo.
- ▲ `cda.m` Empleado para realizar la reducción de dimensionalidad usando **CDA**.
- ▲ `GNLM.m` Empleado para realizar la reducción de dimensionalidad usando **GNLM**.
- ▲ `IsomapII.m` (Desarrollado por Tenenbaum [84]) Empleado para realizar la reducción de dimensionalidad usando **Isomap**.

Construcción y Entrenamiento de los AAM

Para realizar la construcción de los AAM se requiere la ejecución de los archivos

- ▲ `my_alineacion.m` Empleado para realizar el análisis de Procrustes o alineación de forma, y construir el modelo estadístico.
- ▲ `my_takedata2.m` Usando para realizar la normalización de los niveles de gris de las imágenes y la construcción del modelo estadístico de textura.
- ▲ `my_combined_PCModel.m` Con el cual se realiza la construcción del modelo combinado de forma y textura.
- ▲ `my_PCA_g.m` Programa para realizar descomposición en eigenvalores y eigenvectores de la matriz de covarianza asociada a la textura, ya que su tamaño obliga a realizar una proyección auxiliar.
- ▲ `my_Rmatrix_obain.m` Con el cual se realiza el proceso de entrenamiento mediante la construcción de la matriz $\hat{\mathbb{R}}$.

6.2.6. Códigos Auxiliares

- ▲ `my_warpping.m` Empleado para realizar el *image warping* dos formas triangulares.
- ▲ `L2_distance.m` (Desarrollado por Tenenbaum [84]) Empleado para computar la matriz de distancias euclidianas entre pares de puntos.
- ▲ `dis_distance.m` (Desarrollado por Geng [87]) Usado para computar la matriz de disimilitud, usada en la reducción de dimensionalidad supervisada.

7

Conclusiones, Discusión y Trabajos Futuros

En este capítulo se consignan las conclusiones asociadas a la investigación y los resultados obtenidos. Se dan comentarios generales para luego emitir conceptos particulares propios de cada temática. En la sección 7.2 se consigna un comentario respecto a la dirección y enfoque de las investigaciones posteriores a corto plazo, teniendo en cuenta la perspectiva y la visión adquirida durante la realización de este proyecto.

Finalmente se presentan las recomendaciones para trabajos futuros, dando inicialmente un concepto general, para entrar luego en las particularidades de cada temática.

7.1. Conclusiones

En este trabajo fueron desarrolladas herramientas computacionales que permiten realizar el seguimiento dinámico de las expresiones faciales en una secuencia de vídeo, teniendo en cuenta los desarrollos más relevantes en la materia. Estas herramientas están enmarcadas en un esquema en el cual se busca el aprovechamiento de la riqueza en la información dinámica, aún poco explorada.

Dicho paquete de herramientas consta de un detector de rostro, un localizador de puntos faciales, un extractor de *features* y un reductor de dimensionalidad que permitirán extraer los flujos de información para su posterior uso, ya sea en clasificación o síntesis.

7.1.1. Detección de Rostro

1. El esquema de detección facial planteado aprovechó las fortalezas de *Adaboost*, buscando una localización más acertada del rostro con la ayuda de segmentación cromática basada en *CCS* y *CCIN*. Si bien el problema de segmentación por color se enfoca en la distribución cromática de la piel, este mismo esquema puede emplearse para la detección de otros objetos.
2. La técnica empleada para la corrección de iluminación brinda una alternativa robusta para lidiar con las variaciones en el espacio de color, de la distribución de la piel. Sin embargo debe siempre buscarse un escenario de buena calidad para la toma de las secuencias de vídeo, en donde existan fondos de colores bien diferenciados respecto de esta distribución y que no produzcan modificaciones severas en los espacios de representación.

7.1.2. Localización de Puntos Faciales

1. El uso de los *AAM* en la localización de puntos faciales brinda una alternativa sólida para este fin, ya que es una teoría consolidada en la cual se conocen bien sus alcances y limitaciones, que están asociados a los requerimientos de correlación entre las imágenes objetivo y las de entrenamiento.

7.1.3. Extracción de Características

1. El esquema para realizar la extracción de características evita los efectos de tratar datos en grandes dimensiones, pues no se consideran directamente las imágenes sino parámetros extraídos a partir de las mismas empleando propuestas consolidadas para dicha extracción. Adicionalmente con este esquema se tiene una medida de las simetrías en los movimientos faciales, y la separación de los datos en variedades diferentes brinda un análisis más amplio y exhaustivo de las regiones facial superior e inferior.
2. Teniendo en cuenta que las asimetrías en las expresiones pueden producir una variación de la dimensión intrínseca estimada, se extrae la información sin dar una generalización de la dimensión intrínseca de la variedad sobre la cual recaen los datos. Esto evita pérdida de información o inclusión de dimensiones adicionales innecesarias para realizar un posterior procesamiento.
3. Si se optara por representar la variedad descriptora por medio de los niveles de gris de la zona facial extraída, se tendría dicha variedad encajada o incrustada en \mathbb{R}^{RS} siendo $R \times S$ el tamaño de la imagen. El esquema propuesto en esta investigación para la extracción y representación de los *features*, supera enormemente esta primera opción de representación pues las variedades están encajadas en espacios euclidianos de dimensiones fijas, y no se ven afectadas por los efectos nocivos del aumento de la dimensionalidad a medida que $RS \rightarrow \infty$.

7.1.4. Reducción de Dimensionalidad

1. Las técnicas seleccionadas por sus propiedades para la reducción de dimensionalidad se basan en el empleo de las distancias de grafo como aproximaciones de las geodésicas. En el capítulo 5 se indicó que dicha calidad de aproximación depende de la cantidad y densidad de los datos tratados, razón por la cual debe buscarse en lo posible analizar una cantidad de expresiones amplia, en secuencias de vídeo con una cantidad de *frames* que sea lo mayor posible junto con una alta tasa de muestreo. Sumado a esto se encuentra que es más conveniente considerar las distintas expresiones de un sujeto sobre una misma variedad a considerar cada expresión individual sobre una variedad diferente.
2. Se realizó la propuesta de las técnicas **PCA** – ℓm y **PCA** – ω como alternativas para realizar la estimación intrínseca de la dimensión, aprovechando los conceptos y desarrollos teóricos asociados a variedades diferenciales. Los resultados obtenidos por medio de éstas son comparables a los que se obtienen empleando dimensión de correlación, y se espera que sienten un punto de discusión en una temática aún en desarrollo en donde no existen ni métodos ni soluciones definitivas para ningún problema.
3. Teniendo en cuenta que *s-Isomap* tienen su mejor comportamiento sobre variedades desarrollables, los resultados contenidos en

$$\widehat{\mathbf{F}}_{\nu, \hat{n}}^{i, \xi=s-\text{Isomap}}$$

pueden representar versiones deformadas de los resultados asociados a

$$\widehat{\mathbf{F}}_{\nu, \hat{n}}^{i, \xi=s-\text{CDA}}, \quad \text{y} \quad \widehat{\mathbf{F}}_{\nu, \hat{n}}^{i, \xi=s-\text{GNLM}}$$

pero con valiosos aportes para lograr la discriminación entre las diferentes clases que recaen sobre la variedad.

4. Lee [58] prueba que los métodos de reducción de dimensionalidad pueden usarse con total confianza para datos que se encuentren encajados en espacios de dimensión N , con $N \leq 5$. En el esquema de extracción de *features* que se planteó en el capítulo 4, se garantiza en lo que se refiere a la dimensión del espacio de representación original, que se tiene un grado total de confianza para el procesamiento de las variedades $\widehat{\mathcal{M}}_{FIF}, \widehat{\mathcal{M}}_{FIS}$ que están encajadas en espacios de dimensiones 3 y 4 respectivamente. Mientras que para las variedades $\widehat{\mathcal{M}}_{FST}, \widehat{\mathcal{M}}_{FIT}$ el grado de confianza es alto pues están en espacios de dimensión 6 que es apenas una unidad mayor al umbral de confianza. Para el procesamiento de la variedad $\widehat{\mathcal{M}}_{FSF}$, encajada en \mathbb{R}^9 el grado de confianza aún puede considerarse cercano al valor óptimo en comparación con procesar la totalidad de la información en un solo *bloque*.

Con esto se confirma aún más la ventaja de esta separación de la información, en comparación con el uso directo de la información de textura a partir de la imagen, pues en este último enfoque la dimensión RS del espacio de representación sería $RS \gg 5$ y por lo tanto la confianza sería muy baja.

5. Por medio del uso de la matriz de disimilaridad propuesta por Geng [87] se lograron mejores resultados que los obtenidos sin el empleo de la misma, con las técnicas de reducción de dimensionalidad **CDA**, **GNLM** e **Isomap**, ya que el uso de ésta permite *separar* la información de distintas clases que recaen sobre una misma variedad.

El uso de esta matriz con **Isomap** fué propuesto por Geng [87], y en este trabajo se analizó y se concluyó que era posible, debido a su planteamiento, aplicarla en **CDA** y **GNLM**, con resultados potencialmente mejores que los obtenidos con **Isomap** pues estas dos últimas técnicas manejan la no linealidad tanto con aproximaciones de distancias geodésicas como con la minimización de una función de estrés.

7.2. Discusión

Sin duda alguna el procesamiento de la información extraída de las expresiones faciales, depende y se ve fuertemente afectada por los métodos de reducción de dimensionalidad y las técnicas empleadas para realizar la estimación intrínseca de la variedad. Esto implicaría que las investigaciones realizadas a futuro deben centrar la mayor parte de sus esfuerzos en realizar contribuciones en estas temáticas pues su repercusión será directa en el análisis dinámico de patrones faciales. Aunque esto podría desviar un poco la atención de la finalidad esencial, se lograrían contribuciones no solo en esta temática sino en el análisis de información en general.

7.3. Recomendaciones sobre Trabajos Futuros

Ya que en este trabajo se realiza el seguimiento dinámico de las expresiones faciales, se propone realizar una investigación en la cual se estudien los procesos de clasificación y síntesis, considerando la información extraída $\widehat{\mathbf{F}}_{\nu, \widehat{n}}^{i, \xi}$ con el esquema desarrollado en este trabajo, a fin de encontrar la herramienta de clasificación que aproveche mejor la riqueza de esta información.

Aunque en esta investigación los tiempos de cómputo no jugaron un papel esencial, se sugiere el desarrollo de un proyecto en el cual se realice una implementación en C++ de la totalidad de las herramientas que hacen parte del esquema.

7.3.1. Detección de Rostro

1. Buscando ampliar los límites de aplicación de los desarrollos logrados, se propone realizar una investigación en donde se aborden imágenes con oclusiones y grandes rotaciones craneales,

siguiendo los trabajos de Ichikawa [11] y Huang [12] en donde se emplea también *Adaboost* para lidiar con estos problemas.

7.3.2. Localización de Puntos Faciales

1. Buscando disminuir el grado de correlación entre las imágenes objetivo y las imágenes de entrenamiento empleadas en los *AAM*, se recomienda integrar los *AAM* junto con la propuesta de Valstar [40]. Esta integración debe hacerse una vez sean conocidos los detalles de ese desarrollo, para delimitar con precisión el escenario de aplicación.

7.3.3. Extracción de Características y Reducción de Dimensionalidad

1. Si bien el planteamiento de los *features* para describir el comportamiento de las expresiones faciales es el resultado de una investigación profunda realizada por Tian [55], se sugiere estudiar y diseñar el planteamiento de un conjunto de *features* que mantenga las características esenciales de esta propuesta, pero que recaiga sobre una variedad de geometría conocida, a fin de aplicar con mucha más precisión los métodos de reducción de dimensionalidad.
2. Se sugiere realizar una investigación a fin de llegar al planteamiento de criterios para la selección óptima de los parámetros en el uso de técnicas como **LLE**, **HLLE** y **LTSA** en la reducción de dimensionalidad y **MLE** para realizar la estimación de la dimensión intrínseca, las cuales poseen fuertes cimientos teóricos.
3. A pesar de que la propuesta de usar **PCA**– ℓm y **PCA**– ω para la estimación de la dimensión intrínseca aprovecha conceptos de variedades diferenciales y asocia resultados sobre la selección del número de componentes, ésta **no** se plantea como una solución definitiva en un campo donde aún los desarrollos se basan en hipótesis razonables pero no siempre comprobables. Se sugiere, realizar una investigación en donde se realicen pruebas más allá de las consignadas en este trabajo, que si bien siguen exactamente las pruebas de otros trabajos, no muestran un comportamiento con conjuntos de datos propios de otras disciplinas.
4. Teniendo en cuenta las ventajas de los desarrollos realizados en torno a las series de tiempo, se sugiere explorar el uso de estas herramientas para modelar la información de las expresiones faciales una vez realizada la reducción de dimensionalidad, a fin de obtener beneficios tanto para clasificación como para síntesis.
5. Si bien la base de Datos Cohn-Kanade [7] brinda una excelente alternativa para contrastar resultados con otros autores, se sugiere como un proyecto a largo plazo la creación de una base de datos de secuencias de expresiones faciales que posean mejores condiciones de iluminación, mejor calidad de imágenes, fondos con colores lejanos a la distribución de la piel y una mejor tasa de *frames* por segundo, a fin de mejorar la información disponible para estas investigaciones.

7.4. Comparación con el Trabajo de Otros Autores

En el capítulo 5, por medio de las pruebas realizadas sobre $\widehat{\mathcal{M}}_{FSF}$ se evidenció y se comprobó la hipótesis de la existencia de expresiones bien diferenciadas entre sí para un individuo dado. La visualización de estos resultados es comparable a la realizada por Ping en [57], en donde con un esquema de extracción de características diferente se logró establecer por medio de visualización, la diferencia entre distintas expresiones en espacios de representación euclidianos de 2 y/o 3 dimensiones.

Bibliografía

- [1] Stan Z. Li and Anil K. Jain. *Handbook of Face Recognition*. Springer, 2005.
- [2] Stan Z. Li and Anil K. Jain. *Handbook of Face Recognition*. Springer, 2011.
- [3] Daniel Goleman. *La Inteligencia Emocional*. Vergara, 2004.
- [4] Paul Ekman, Harrieh Oster. Expresiones faciales de la emoción. *Annual Review of Psychology*, 30:527–554, 1979.
- [5] Paul Ekman, Wallace V. Friesen. Constants across cultures in the face and emotion. *Journal of Personality and Social Psychology*, 17:124–129, 1971.
- [6] Erika L. Rosenberg Paul Ekman. *What the Face Reveals*. Oxford University Press, 1997.
- [7] Cohn J. F. y Tian Y Kanade, T. Comprehensive database for facial expression analysis. *Proceedings of the Fourth IEEE International Conference on Automatic Face and Gesture Recognition (FG'00), Grenoble, France*, pages 46–53, 2000.
- [8] Nathan Faggian, Andrew Paplinski, and Tat-Jun Chin. Face recognition from video using active appearance model segmentation. In *Proceedings of the 18th International Conference on Pattern Recognition - Volume 01*, ICPR '06, pages 287–290, Washington, DC, USA, 2006. IEEE Computer Society.
- [9] A. Hadid, J.Y. Heikkila, O. Silven, and M. Pietikainen. Face and eye detection for person authentication in mobile phones. In *Distributed Smart Cameras, 2007. ICDSC '07. First ACM/IEEE International Conference on*, pages 101 –108, sept. 2007.
- [10] M. Jones P. Viola. Rapid object detection using a boosted cascade of simple features. *Proceedings of the IEEE Computer Society conference on Computer Vision and Pattern Recognition*, 20, 2001.
- [11] K. Ichikawa, T. Mita, and O. Hori. Component-based robust face detection using adaboost and decision tree. In *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*, pages 413 –420, april 2006.
- [12] Chang Huang, Haizhou Ai, Yuan Li, and Shihong Lao. High-performance rotation invariant multiview face detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(4):671 –686, april 2007.
- [13] Hongliang Li, King Ngi Ngan, and Qiang Liu. Faceseg: automatic face segmentation for real-time video. *IEEE Transactions on Multimedia*, 11(1):77–88, 2009.
- [14] Yan-Wen Wu and Xue-Yi Ai. An improvement of face detection using adaboost with color information. In *Proceedings of the 2008 ISECS International Colloquium on Computing, Communication, Control, and Management - Volume 01*, pages 317–321, Washington, DC, USA, 2008. IEEE Computer Society.

- [15] Peng Deng and Mingtao Pei. Multi-view face detection based on adaboost and skin color. In *Proceedings of the 2008 First International Conference on Intelligent Networks and Intelligent Systems*, pages 457–460, Washington, DC, USA, 2008. IEEE Computer Society.
- [16] Rein-Lien Hsu, M. Abdel-Mottaleb, and A.K. Jain. Face detection in color images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):696–706, may 2002.
- [17] M. Beniak, J. Pavlovicova, and M. Oravec. Automatic face detection based on chrominance components analysis. In *Systems, Signals and Image Processing, 2008. IWSSIP 2008. 15th International Conference on*, pages 475–478, june 2008.
- [18] A. Mohamed, Ying Weng, Jianmin Jiang, and S. Ipson. Face detection based neural networks using robust skin color segmentation. In *Systems, Signals and Devices, 2008. IEEE SSD 2008. 5th International Multi-Conference on*, pages 1–5, july 2008.
- [19] Lihong Zhao, Xiaolin Sun, and Xinhe Xu. Face detection based on facial features. In *Signal Processing, 2006 8th International Conference on*, volume 3, nov. 2006.
- [20] Qiong Liu and Guang zheng Peng. A robust skin color based face detection algorithm. In *Informatics in Control, Automation and Robotics (CAR), 2010 2nd International Asia Conference on*, volume 2, pages 525–528, march 2010.
- [21] Hwei-Jen Lin, Shu-Yi Wang, Shwu-Huey Yen, and Yang-Ta Kao. Face detection based on skin color segmentation and neural network. In *Neural Networks and Brain, 2005. ICNN B '05. International Conference on*, volume 2, pages 1144–1149, oct. 2005.
- [22] D. Chai and K.N. Ngan. Face segmentation using skin-color map in videophone applications. *Circuits and Systems for Video Technology, IEEE Transactions on*, 9(4):551–564, jun 1999.
- [23] Xue wu Zhang, Ling yan Liang, Dun qin Duan, and Wei liang Xia. A novel method of face detection based on fusing ycbcr and his color space. In *Automation and Logistics, 2009. ICAL '09. IEEE International Conference on*, pages 831–835, aug. 2009.
- [24] Sung-Hoon Kim, Hyon-Soo Lee, and Hyung-Ho Kim. Robust extraction of face candidate through segmentation and conditional merging in skin area. In *Intelligent Computing and Intelligent Systems, 2009. ICIS 2009. IEEE International Conference on*, volume 1, pages 547–551, nov. 2009.
- [25] Marc Ebner. *Color Constancy*. Wiley, 2007.
- [26] Kah Phooi Seng King Hann Lim, Li-Minn Ang. New face segmentation technique insusceptible to illumination. *Proceedings of ECTI-CON*, pages 449–452, 2008.
- [27] G.D. Finlayson, B. Schiele, J.L. Crowley. Comprehensive colour image normalization. *ECCV98 Fifth Wuropean Conference on Computer Vision*, 1:475–490, 1998.
- [28] y J.Zhang Q. Zhang, S.C. Kamata. Face detection and tracking in color images using color centroids segmentation. *International Conference on Robotics and Biomimetics, Bankok, Thailand*, pages 1008–1013, feb 2009.
- [29] P.J. Phillips, H. Wechsler, J. Huang, P. Rauss. The feret database and evaluation procedure for face recognition algorithms. *Image and Vision Computing J*, 16:295–306, 1998.
- [30] S.A. Rizvi P.J. Rauss P.J Phillips, H. Moon. The feret evaluation methodology for face recognition algorithms. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 22:1090–1104, 2000.
- [31] T. Sim, S. Baker, M. Bsat. The cmu pose, illumination, and expression (pie) database of human faces. *Carnegie Mellon University*, pages 295–306, 2000.
- [32] H. Masnadi-Shirazi. Adaboost face detection. *Department of Electrical and Computer Engineering. University of California, San Diego UCSD*, pages 1–6, 2007.

- [33] Longbin Chen. A novel facial feature localization method using probabilistic-like output *.
- [34] Tie Yun and Ling Guan. Automatic fiducial points detection for facial expressions using scale invariant feature. In *Multimedia Signal Processing, 2009. MMSP '09. IEEE International Workshop on*, pages 1–6, oct. 2009.
- [35] Yongli Hu, Baocai Yin, and Dehui Kong. A new facial feature extraction method based on linear combination model. In *Web Intelligence, 2003. WI 2003. Proceedings. IEEE/WIC International Conference on*, pages 520–523, oct. 2003.
- [36] Shaojun Zhu and Jieyu Zhao. Facial feature points extraction. In *Image and Graphics, 2009. ICIG '09. Fifth International Conference on*, pages 195–199, sept. 2009.
- [37] Ying Li, J.H. Lai, and P.C. Yuen. Multi-template asm method for feature points detection of facial image with diverse expressions. In *Automatic Face and Gesture Recognition, 2006. FGR 2006. 7th International Conference on*, pages 435–440, april 2006.
- [38] D. Vukadinovic and M. Pantic. Fully automatic facial feature point detection using gabor feature based boosted classifiers. In *Systems, Man and Cybernetics, 2005 IEEE International Conference on*, volume 2, pages 1692–1698 Vol. 2, oct. 2005.
- [39] Tatsuo Kozakaya, Tomoyuki Shibata, Mayumi Yuasa, and Osamu Yamaguchi. Facial feature localization using weighted vector concentration approach. *Image Vision Comput.*, 28:772–780, May 2010.
- [40] M. Valstar, B. Martinez, X. Binefa, and M. Pantic. Facial point detection using boosted regression and graph models. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2729–2736, june 2010.
- [41] Mikkel Bille Stegmann. *Active Appearance Models: Theory, Extensions and Cases. Master Thesis*. Technical University of Denmark, 2000.
- [42] T.F. Cootes and C.J.Taylor. *Statistical Models of Appearance for Computer Vision*. University of Manchester, 2000.
- [43] Tim.F. Cootes and Chris.J.Taylor. *Statistical Models of Appearance for Computer Vision*. University of Manchester, 2004.
- [44] Ian L. Dryden and K. V. Mardia. *Statistical Shape Analysis*. Wiley, 1998.
- [45] Rhodri Davies, Carole Twining, and Chris Taylor. *Statistical Models of Shape, optimization and evaluation*. Springer, 2008.
- [46] I.T. Jolliffe. *Principal Component Analysis*. Springer, 2002.
- [47] Ra Psarrou A, Shaogang Gong B, and Michael Walter A. Eigenfaces for recognition. In *In J of Cognitive Neuroscience v3 nr1*, pages 71–86, 1991.
- [48] Yu Y.L. Zhuji. Face recognition with eigenfaces. In *Industrial Technology, 1994. Proceedings of the IEEE International Conference on*, pages 434–438, 1994.
- [49] C.J. Taylor G.J. Edwards and T.F. Cootes. Interpreting face images using active appearance models. pages 1–6, 1998.
- [50] T.F. Cootes G.J. Edwards and C.J. Taylor. Advances in active appearance models. pages 137–142, 1999.
- [51] Xuelong Li Dacheng Tao Xinbo Gao, Ya Su. A review of active appearance models. *IEEE Transactions on Systems, Man, and cybernetics-Part C: Applications and Reviews*, pages 145–158, 2010.

- [52] C.J. Taylor T.F. Cootes, G.J. Edwards. Active appearance models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 681–685, 2001.
- [53] Jerome Friedman Trevor Hastie, Robert Tibshirani. *The Elements of Statistical Learning*. Springer, 2009.
- [54] Rafael C. Gonzalez and Richard E. Woods. *Digital Image Processing (3rd Edition)*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 2006.
- [55] Y.-I. Tian, T. Kanade, and J.F. Cohn. Recognizing action units for facial expression analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 23(2):97–115, feb 2001.
- [56] Ilse Rayvse. *Facial Analysis and Synthesis*. PhD thesis, FACULTEIT INGENIEURSWETENSCHAPPEN, Vakgroep Elektronica and Informatica (ETRO) Audio-Visuele Signaalverwerking (AVSP), 2006.
- [57] Wang Fengna Rayvse Ilse Fan Ping, Jiang Dongmei and Sahli Hichem. Manifold analysis for subject independent dynamic emotion recognition in video sequences. *Fifth International Conference on Image and Graphics*, pages 896–901, 2009.
- [58] John A. Lee, Michel Verleysen. *Nonlinear Dimensionality Reduction*. Springer, 2007.
- [59] Carrie Grimes. *New Methods in Nonlinear Dimensionality Reduction, Ph.D Thesis*. Stanford University, 2003.
- [60] Matthias Hein and Jean-Yves Audibert. Intrinsic dimensionality estimation of submanifolds in rd. In *Proceedings of the 22nd international conference on Machine learning, ICML '05*, pages 289–296, New York, NY, USA, 2005. ACM.
- [61] David L. Donoho, Carrie Grimes. Hessian eigenmaps: New locally linear embedding techniques for high-dimensional data. *Proc. of National Academy of Sciences*, 100:5591–5596, 2003.
- [62] Mikhail Belkin, Partha Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15:1373–1396, 2002.
- [63] Damien Francois. *High-dimensional data analysis: optimal metrics and feature selection, Docteur Thèse*. Université catholique de Louvain, 2006.
- [64] David L. Donoho. Aide-memoire. high-dimensional data analysis: The curses and blessings of dimensionality, 2000.
- [65] Pierre Demartines. *Analyse de Données par Réseaux de Neurones Auto-organisés, Docteur Thèse*. Institut National Polytechnique de Grenoble, 1992.
- [66] W.S. Massey. *Algebraic Topology: An Introduction*. Springer, 1967.
- [67] Michael Spivak. *A Comprehensive Introduction to Differential Geometry*. Publish or Perish, 2005.
- [68] K.J. Falconer. *Techniques in Fractal Geometry*. Wiley, 1997.
- [69] TmDsSauer Kathleen T.Alligood and James A. Yorke. *Chaos: An introduction to Dynamical Systems*. Springer, 1997.
- [70] A.R. Pears. *Dimension Theory of General Spaces*. Cambridge University Press, 1975.
- [71] John M. Lee. *Introduction to Smooth Manifolds*. Springer, 2006.
- [72] James C. Robinson. *Dimensions, Embeddings, and attractors*. Cambridge University Press, 2011.
- [73] K.J. Falconer. *The Geometry of Fractal Sets*. Cambridge University Press, 1985.

- [74] Balázs Kégl. Intrinsic dimension estimation using packing numbers. *Neural Computation*, 15:681–688, 2003.
- [75] L.J.P. van der Maaten. An introduction to dimensionality reduction using matlab, 2007.
- [76] James Theiler, Turab Lookman. Statistical error in a chord estimator of correlation dimension: The rule of five. *Int.J.Bif. Chaos*, 3:765–771, 1993.
- [77] Shun-Ichi Amari Andrzej Cichocki. *Adaptive Blind Signal and Image Processing*. Wiley, 2002.
- [78] Elizaveta Levina. Maximum likelihood estimation of intrinsic dimension. *Advances in Neural Information Processing Systems*, 17:777–784, 2005.
- [79] Michael A. Cox Trevor F. Cox. *Multidimensional Scaling*. Chapman and Hall-CRC, 2001.
- [80] Sammon, J. W. A nonlinear mapping for data structure analysis. *IEEE Trans. Comput.*, 18:401–409, 1969.
- [81] Juha Vesanto. Som toolbox 2.0-sammon. <http://www.cis.hut.fi/projects/somtoolbox/>, 2000.
- [82] Pierre Demartines and Jeanny Hérault. Curvilinear component analysis: a self-organizing neural network for nonlinear mapping of data sets. *IEEE Transactions on Neural Networks*, XX:1–10, 1996.
- [83] Juha Vesanto. Som toolbox 2.0-cca. <http://www.cis.hut.fi/projects/somtoolbox/>, 2000.
- [84] J. B. Tenenbaum, V. Silva, and J. C. Langford. A Global Geometric Framework for Nonlinear Dimensionality Reduction. *Science*, 290(5500):2319–2323, 2000.
- [85] Mira Bernstein and Vin De Silva and John C. Langford and Joshua B. Tenenbaum. Graph approximations to geodesics on embedded manifolds. *Science*, pages 1–26, 2000.
- [86] Chao Shao, Houkuan Huang. Selection of the optimal parameter value for the isomap algorithm. *Springer-Verlag*, pages 396–404, 2005.
- [87] Xin Geng, De chuan Zhan, and Zhi hua Zhou. Supervised nonlinear dimensionality reduction for visualization and classification. *IEEE Transactions on systems, man, and cybernetics-part B: cybernetics*, pages 1098–1107, 2005.
- [88] John Aldo Lee, Amaury Lendasse, Nicolas Conckers, Michel Verleysen. A robust nonlinear projection method. *ESANN'2000 proceedings-European symposium on Artificial Neural Networks Bruges (Belgium)*, pages 13–20, 2000.
- [89] John Aldo Lee, Amaury Lendasse, Michel Verleysen. Curvilinear distance analysis versus isomap. *ESANN proceedings-European Symposium on Artificial Neural Networks*, pages 185–192, 2002.
- [90] Alexander J. Simola Bernhard Scholkopt. *Learning with Kernels*. The MIT Press, 2002.
- [91] Sam T. Roweis and Lawrence K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000.
- [92] Lawrence K. Saul and Sam T. Roweis. An introduction to locally linear embedding. Technical report, 2000.
- [93] Manfredo Perdigao Do Carmo. *Riemannian Geometry*. Birkhauser, 1992.
- [94] John A. Lee and Michel Verleysen. Nonlinear projection with the isotop method. In *in: J.R. Dorronsoro (Ed.), Artificial Neural Networks, Lecture Notes in Computer Science*, pages 933–938. Springer, 2002.

- [95] John Aldo Lee, Cédric Archambeau, and Michel Verleysen. Locally linear embedding versus isotop. In *Proceedings of ESANN 2003, 11th European Symposium on Artificial Neural Networks*, pages 527–534, 2003.
- [96] Zhenyue Zhang and Hongyuan Zha. Principal manifolds and nonlinear dimension reduction via local tangent space alignment. *SIAM Journal of Scientific Computing*, 26:313–338, 2002.
- [97] Anders Brun, Carl fredrik Westin, Magnus Herberthson, and Hans Knutsson. H.: Fast manifold learning based on riemannian normal coordinates. In *Proc. 14th Scandinavian Conf. on Image Analysis*, 2005.