

**VERIFICACIÓN DE CORRESPONDENCIA EN HUELLAS DACTILARES
APLICANDO TÉCNICAS DE PROCESAMIENTO Y ANÁLISIS DIGITAL DE
IMÁGENES PARA LA DISMINUCIÓN DEL TIEMPO DE COTEJO**

JORGE ALBERTO ADRADA MOLANO

Universidad del Cauca

Facultad de Ciencias Naturales, Exactas y de la Educación

Programa de Ingeniería Física

Grupo de Investigación de Óptica y Láser

Popayán – Cauca

2010

**VERIFICACIÓN DE CORRESPONDENCIA EN HUELLAS DACTILARES
APLICANDO TÉCNICAS DE PROCESAMIENTO Y ANÁLISIS DIGITAL DE
IMÁGENES PARA LA DISMINUCIÓN DEL TIEMPO DE COTEJO**

JORGE ALBERTO ADRADA MOLANO

**Trabajo de investigación para optar al título de
Ingeniero Físico**

Director

LEONAIRO PENCUE FIERRO

Ingeniero Físico

Universidad del Cauca

Facultad de Ciencias Naturales, Exactas y de la Educación

Programa de Ingeniería Física

Grupo de Investigación de Óptica y Láser

Popayán – Cauca

2010

Nota de aceptación

Ing. SANDRA MILENA ROA MARTÍNEZ

Evaluadora del Proyecto

Ing. PABLO JAVIER SALAZAR VALENCIA

Evaluador del Proyecto

Ing. LEONAIRO PENCUE FIERRO

Director del proyecto de investigación

Fecha de sustentación: 3 marzo de 2010

© Copyright por
JORGE ALBERTO ADRADA MOLANO
2009

A Dios, por sus constantes bendiciones.

AGRADECIMIENTOS

Primero que todo quiero agradecer a Dios por ser mi apoyo incondicional en todo lo que hago, infinitas gracias le doy a Él por darme una familia maravillosa.

A mi Padre Roosevelt Adrada, mi Madre Alexandra Molano y Hermano Juank Adrada les agradezco mucho por todo el amor brindado, por sus enseñanzas, por hacerme reír, por darme amor, amistad, por los valores que me enseñaron y por todas las cosas que hemos vivido, porque con la ayuda de ellos he logrado llegar a donde estoy.

Agradezco muy especialmente a mi tío Felipe Adrada y su familia, quien me acogió en su hogar por largo tiempo, me brindaron su afecto, amistad y el empuje para no rendirme en el estudio.

A toda mi familia en general tíos, tías, primas, primos, abuela y abuelo, gracias por darme su cariño y escucharme en todo momento.

Agradezco a Karem por brindarme su amor y apoyo en todo momento, por aguantarme, seguir mis locuras y ser un gran apoyo en lo que hago...gracias por todo.

A mis compañeros que los considero unos verdaderos amigos, con ellos aprendí muchas cosas de la vida y de la física, gracias, gracia...gracias.

Gracias a Leonairo Pencue quien fue el director de trabajo de grado y al Grupo de Óptica y Láser de la Universidad del Cauca, a F. Buchelli por su introducción a OpenCV, gracias a ellos logré introducirme en el campo de la visión artificial, y a todos los profesores del Departamento de Física de la Universidad del Cauca, por enseñarme todo lo que se.

Gracias al C.T.I. por creer en mis capacidades y permitirme trabajar con ellos, en especial al grupo de lofoscopia Nilvia, Nora y Fredy, quienes me brindaron su apoyo en todo momento.

Jorge Alberto Adrada Molano

Popayán, 19 de noviembre de 2009, 5:36 pm.

RESUMEN

La biometría es ampliamente usada para la identificación y reconocimiento de personas, se basa en medidas de patrones únicos de cada persona, tales como la huella dactilar, el rostro, la geometría de las manos, el iris, el ADN, entre otros. No obstante el análisis de las huellas dactilares es el más utilizado y aceptado como medida biométrica, debido a la facilidad para obtener y almacenar las impresiones dactilares. La creciente demanda de dactilogramas, tomadas en su mayoría por el departamento de policía, el registro civil y la división pericial, han hecho crecer enormemente los archivos de almacenamiento de impresiones dactilares, por lo cual un cotejo manual tomaría meses en llegar a una búsqueda exitosa, debido a esto se hace necesario implementar sistemas automáticos de comparación de huellas. El medio automático más popular para la búsqueda de identificación o reconocimiento dactilar es el Sistema de Identificación Automática de Huellas Dactilares (*AFIS, por sus siglas en ingles*) el cual es un software que permiten comparar la similitud entre una huella y una base de datos, utilizando algoritmos avanzados de procesamiento digital de imágenes, que permiten extraer características de las huellas en las peores condiciones[1, 2].

En este trabajo se describe el cotejo de un par de huellas, por medio de un software de verificación que compara una huella contra otra, el cual es el primer paso para el diseño de sistemas de identificación donde se compara una huella contra n-huellas de una base de datos.

Para ello se han explorado en la literatura, distintos algoritmos para solucionar este problema, se decidió para el mejoramiento de las huellas el uso del filtro de Gabor, el cual es el más reportado y ofrece buenos rendimientos para la eliminación del ruido, seguido de todos los procesos para la aplicación de este filtro como son: la normalización del histograma de grises, la estimación de la dirección de las crestas y su frecuencia local. Por otra parte está el problema de extraer características a las huellas, en esta etapa se eligió las mismas características que usan los dactiloscopistas para la comparación de huellas, es decir la detección de minucias tales como terminaciones y bifurcaciones, la cual se obtiene del adelgazamiento de la imagen filtrada, con una post-etapa de eliminación de falsas minucias. Finalmente se llega a la comparación de huellas mediante sus minucias, en este punto se usan dos algoritmos una llamada comparación con alineación previa y la otra comparación mediante grafos, ambas son buenas y ofrecen buenos resultados que más adelante se verán.

TABLA DE CONTENIDO

AGRADECIMIENTOS	VI
RESUMEN.....	VII
LISTA DE TABLAS	XI
LISTA DE FIGURAS	XII
1. INTRODUCCIÓN.....	1
1.1. Problema	1
1.2. Antecedentes	2
1.3. Justificación.....	3
1.4. Objetivos	4
1.4.1. Objetivo General.....	4
1.4.2. Objetivos Específicos	4
2. MARCO TEÓRICO	5
2.1. BIOMETRÍA.....	5
2.1.1. Introducción	5
2.1.2. Sistemas Biométricos.....	6
2.1.2.1. Huellas Dactilares.....	7
2.1.2.2. Rostro.....	7
2.1.2.3. Geometría de las Manos.....	8
2.1.2.4. Iris y Escaneo Retinal	8
2.1.2.5. Voz.....	9
2.1.2.6. Comparación de las Tecnologías Biométricas	9
2.1.2.7. Aplicaciones	11
2.2. HUELLAS DACTILARES.....	11
2.2.1. Formación de la Huella Dactilar.....	11
2.2.2. Clasificación de las Huellas Dactilares.....	12
2.2.3. Medidas de Errores en los Sistemas de Verificación.....	14
2.3. ADQUISICIÓN Y PROCESAMIENTO	16
2.3.1. Adquisición de Huellas Dactilares	16

2.3.2.	MEJORAMIENTO.....	20
2.3.2.1.	Trabajos Relacionados.....	20
2.3.2.2.	Algoritmo de Mejoramiento.....	21
2.3.2.3.	Segmentación	22
2.3.2.4.	Mejora del Histograma	22
2.3.2.5.	Estimación de la Orientación	23
2.3.2.6.	Matriz de Confiabilidad	24
2.3.2.7.	Estimación de la Frecuencia.....	25
2.3.2.8.	Filtrado de Gabor.....	26
2.3.3.	EXTRACCIÓN DE CARACTERÍSTICAS	27
2.3.3.1.	Trabajos Relacionados.....	27
2.3.3.2.	Algoritmo de Extracción de Minucias	28
2.3.3.3.	Binarización	29
2.3.3.4.	Adelgazamiento.....	29
2.3.3.5.	Detección de Minucias.....	30
2.3.3.6.	Eliminación de Falsas Minucias.....	31
2.3.4.	COMPARACIÓN DE HUELLAS	32
2.3.4.1.	Trabajos Relacionados.....	32
2.3.4.2.	Algoritmo de Comparación de Huellas	34
2.3.4.3.	Comparación de Minucias con Alineamiento Previo	34
2.3.4.4.	Comparación de Minucias por Grafos.....	37
3.	RESULTADOS EXPERIMENTALES	39
3.1.	CAPTURA DE IMAGEN.....	39
3.1.1.	Base de datos de la FVC2004	39
3.1.2.	Base de datos del autor	41
3.2.	ALGORITMO	43
3.2.1.	Medidas del tiempo de ejecución	44
3.3.	MEJORAMIENTO.....	45
3.3.1.	Segmentación	45
3.3.2.	Mejoramiento del Histograma.....	47
3.3.3.	Estimación de la Orientación	50

3.3.3.1.	Máscara de Confiabilidad	53
3.3.4.	Estimación de la Frecuencia.....	56
3.3.5.	Filtrado	58
3.3.6.	Conclusiones Parciales del Mejoramiento	67
3.4.	EXTRACCIÓN DE MINUCIAS	68
3.4.1.	Binarización	68
3.4.2.	Adelgazamiento.....	69
3.4.3.	Detección de Minucias.....	71
3.4.4.	Eliminación de Falsas Minucias	72
3.4.5.	Tiempo total de inscripción de la huella (<i>enroll</i>)	73
3.4.6.	Conclusiones Parciales de la Extracción de Minucias.....	74
3.5.	COMPARACIÓN DE LAS HUELLAS	74
3.5.1.	Comparación por Alineamiento Previo	74
3.5.1.1.	Tiempo total de Comparación por alineamiento previo	77
3.5.2.	Comparación por Grafos	77
3.5.2.1.	Tiempo total de Comparación por Grafos	82
3.5.3.	Conclusiones Parciales de la Comparación.....	82
4.	CONCLUSIONES.....	83
4.1.	Trabajos Futuros.....	84
	REFERENCIAS	85
	Anexo A	88
	Anexo B	92
	Anexo C	94

LISTA DE TABLAS

Tabla 1. Comparación entre diversas tecnologías biométricas.	10
Tabla 2. Principales usos de la tecnología biométrica.	11
Tabla 3. Tipos de sensores de la FVC2004.....	40
Tabla 4. Tiempos de comparación y cotejo de la FVC2004.	40
Tabla 5. Resultados de la FVC2004 para la DB1_A.	79

LISTA DE FIGURAS

Figura 1. Diagrama de flujo de sistemas de verificación e identificación.	6
Figura 2. Reporte de mercadeo en biometría.	10
Figura 3. Forma de una huella dactilar en el dedo sano.....	12
Figura 4. Tipos de dactilogramas.....	13
Figura 5. Tipos de minucias dentro de la huella.	14
Figura 6. Típicas graficas de FAR y FRR, y la ROC.	16
Figura 7. Equipo para la toma de huellas dactilares de forma off-line.....	17
Figura 8. Métodos off-line para tomar huellas.	17
Figura 9. Clases de sensores electrónicos comerciales.	19
Figura 10. Huella tomada a distintas resoluciones.	20
Figura 11. Diagrama de flujo para el mejoramiento de las huellas dactilares.....	22
Figura 12. Procedimiento para el cálculo de la frecuencia local en la imagen.	25
Figura 13. Intensidad de las crestas en dirección perpendicular a ellas.....	25
Figura 14. Expresión del filtro de Gabor en 2D y 3D.....	27
Figura 15. Diagrama de flujo para la extracción de minucias.	28
Figura 16. Vecindades del pixel p_0	29
Figura 17. Consideraciones para detectar el borde de una imagen.....	30
Figura 18. Condiciones para detectar terminaciones.	30
Figura 19. Condiciones para detectar bifurcaciones.	31
Figura 20. Estructuras comunes de falsas minucias.	32
Figura 21. Problema de comparación de minucias alineadas.....	35
Figura 22. Caso en el que dos minucias comparten una minucia en común.	36
Figura 23. Procesos de ajuste de la huella S respecto a C.....	36
Figura 24. Representación de K -plet.....	37
Figura 25. Representación de los ángulos en un K -plet.....	37
Figura 26. Tipos de imágenes de la FVC2004.	39
Figura 27. Huellas de la base de datos del autor.	41

Figura 28. Reconstrucción de huellas con resolución inadecuada.	42
Figura 29. Rutina para comparar dos huellas.	44
Figura 30. Ejemplos de segmentación de huellas,.....	45
Figura 31. Imágenes segmentadas antes y después de la corrección.	46
Figura 32. Normalización y ecualización del histograma.	48
Figura 33. Imágenes del sensor térmico.....	49
Figura 34. Estimación de la orientación.....	51
Figura 35. Estimación de la mala orientación.	52
Figura 36. Matriz de orientación original.	53
Figura 37. Matriz de confiabilidad.	54
Figura 38. Máscara de confiabilidad para las imágenes de la Figura 35.	55
Figura 39. Estimación de la frecuencia.	58
Figura 40. Filtro de Gabor con los parámetros N , $T=31$ y $DE=5$, $w=16$	61
Figura 41. Reconstrucción variando la orientación.	62
Figura 42. Reconstrucción variando orientación y frecuencia.	63
Figura 43. Comparación del filtro de Gabor con otros.	65
Figura 44. Reconstrucción de la huella variando ψ	66
Figura 45. Binarización de la imagen.	68
Figura 46. Adelgazamiento de figuras geométricas simples.	69
Figura 47. Imagen Adelgazada con conectividad 8 y 4.	70
Figura 48. Extracción de minucias.	71
Figura 49. Eliminación de falsas minucias.....	72
Figura 50. Eliminación de minucias variando el tamaño del cuadro.....	73
Figura 51. Huellas escaneadas a distintas resoluciones.	76
Figura 52. Resultados de la FAR, FRR para la DB1_A de la FVC2004.	78
Figura 53. Imágenes de la DB1_A de mala calidad.	79
Figura 54. Resultados de la FAR, FRR con $ERR = 2.3\%$	80
Figura 55. Resultados de la FAR, FRR con $ERR = 9.7\%$	81
Figura 56. Imágenes con mediana y mala calidad.	81

Figura 57. Interface grafica del MID.	88
Figura 58. Despliegue del MID para una comparación de dos huellas.	90

Capítulo 1

1. INTRODUCCIÓN

1.1. Problema

El Cuerpo Técnico de Investigación (C.T.I.) de la Fiscalía seccional Cauca tiene en sus áreas de trabajo el equipo de lofoscopia, que se encarga principalmente del estudio de las crestas dactilares que se encuentran en la falange distal de los dedos, utilizan ésta información para identificar, individualizar a una persona y verificar su identidad, dado que las huellas dactilares se encuentran fundadas en el principio de mismidad.

Actualmente la comparación de huellas se hace visualmente, en la mayoría de los casos con la ayuda de comparadores ópticos y lupas, los cuales amplifican las imágenes, pero no realizan ningún mejoramiento a esta.

El problema de este método es que demanda mucho tiempo y esfuerzo visual para lograr una buena comparación entre ellas, toma alrededor de 5 a 15 minutos cotejar huellas dactilares de buena calidad y hasta 4 horas huellas de mala calidad, debido a problemas como el sobre-entintado o sub-entintado de los dedos presentes en el momento hacer una reseña, además la fuerza con que presionan los dedos sobre el papel, el nivel de sudoración de los dedos de la persona, la calidad de la tinta y el papel de registro, entre otras; afectan las impresiones dactilares.

Por ello se propuso solucionar este problema utilizando técnicas de procesamiento y análisis de imágenes.

1.2. Antecedentes [2, 3]

El estudio de las huellas dactilares empezó en 1684, cuando el Inglés Nehemiah Grew publicó su primer artículo científico acerca de un estudio sistemático en crestas, valles y porosidades de las huellas dactilares. Poco a poco se observó que las huellas dactilares servirían para identificar a una persona de otra, es así como en 1809 Thomas Bewick comenzó a utilizar su huella dactilar como su marca de registro, aun sin conocer que las huellas dactilares son prácticamente irrepetibles, lo cual fue sugerido por Henry Fauld en 1880, basándose en observaciones empíricas encontró que cada huella dactilar es única.

Otra investigación importante fue realizada por Sir Francis Galton, quien en 1888 introdujo el cotejo de huellas dactilares haciendo comparaciones entre minucias. Más adelante en 1899 Edward Henry introdujo su propio sistema para clasificación de huellas dactilares, aunque este es muy complejo de usar, podía reducir la búsqueda de grandes bases de datos.

Muchos continuaron investigando nuevas formas de clasificación y búsqueda de huellas, al punto tal que a principios del siglo veinte el reconocimiento de huellas dactilares se había formalizado y aceptado como un método de identificación. Varias técnicas de reconocimiento dactilar, incluyendo la adquisición o levantamiento de huellas latentes, la clasificación y el cotejo de huellas fueron desarrollados. Rápidamente las bases de datos de todas partes del mundo empezaron a crecer rápidamente, entonces la búsqueda manual de impresiones dactilares comenzó a ser un problema, debido a que se deben almacenar una impresión dactilar por cada dedo. Por ello en los años 60 se empezaron a desarrollar sistemas automáticos de identificación de huellas dactilares (AFIS), en 1972, los tempranos experimentos con reconocimiento óptico de patrones de huellas dactilares parecen haberse dirigido John Fitzmaurice de Barid-Atomics y Joseph Wegstein y Raymond Moore del Departamento Nacional de Estándares de U. S.. También en 1979 las investigaciones del FBI permitieron probar el primer prototipo de AFIS. Alrededor de 1986, el AFIS era lo suficientemente sofisticado y barato, para justificar su adquisición. Este sistema almacenaba la impresión dactilar como una relación de datos entre minucias, similar al sistema de Galton. El éxito del AFIS en búsqueda y cotejo de huellas era tan exitoso que la mayoría de las agencias de ley en todo el mundo empezaron a adquirir un AFIS. Este medio ha mejorado la productividad de operaciones de organismos de ejecución de la ley ya que la búsqueda de huellas dactilares se hace muy rápido, además reduce considerablemente el coste de contratar y entrenar a muchos peritos dactiloscopistas.

Actualmente el campo de acción del análisis de huellas dactilares, es parte de la biometría, que es la ciencia encargada de establecer medidas en los humanos, tales como el estudio del ADN, iris, voz, forma de las manos, venas de las manos, olor, entre otras; estas nuevas tecnologías serán estudiadas en capítulos posteriores.

1.3. Justificación

En éste trabajo de grado se desarrolla un software capaz de establecer la correspondencia entre huellas, esto lleva a explorar y determinar el uso de distintos algoritmos para el mejoramiento de imágenes, la extracción de características de las huellas y la probabilidad de correspondencia entre un par de ellas, haciendo uso de las particulares de esta. Los procesos en este trabajo descritos son válidos tanto para problemas de identificación y verificación de huellas dactilares, teniendo en cuenta que el problema de identificación puede ser implementado como una ejecución secuencial de N cotejos entre dos huellas como se hace en los sistemas de verificación [2], el cual es el tema de estudio en éste trabajo, no obstante el sistema puede realiza un cotejo de huellas lo bastante rápido, para ser implementado en programas AFIS.

Su implementación ahorra tiempo al proceso de comparar huellas dactilares, debido a que los dactiloscopistas puede llevarles alrededor de 10 a 15 minutos, cotejar huellas de buena calidad; y hasta 4 horas huellas de mala calidad, sin embargo, a la mayoría de programas de reconocimiento automático les toma pocos segundos [4], además permite disminuir el descarte de impresiones dactilares de pobre calidad, las cuales se presentan comúnmente como imágenes con bajo contraste y alto ruido de fondo, lo cual implica un gran esfuerzo visual a los investigadores.

Sin embargo éste software puede enriquecer las imágenes, dando un buen contraste aplicando procesos de ecualización local de la imagen, disminuye el ruido de fondo mediante el uso de filtros selectivos, además, la detección de bifurcaciones y terminaciones ayuda a la rápida localización de la mayoría de minucias, que necesitan los investigadores para el cotejo huellas, es decir 10 puntos característicos iguales en ambas huella. Todo esto con el fin de aumentar la confiabilidad y la disminución del tiempo de los cotejos, realizados por los lofoscopistas en el CTI seccional Cauca.

1.4. Objetivos

1.4.1. Objetivo General

Diseñar e implementar un software que muestre la impresión dactilar mejorada, realice la detección de minucias dactilares y estime el porcentaje de similitud entre éstas, para obtener la correspondencia entre ellas, mediante el procesamiento y análisis digital de imágenes, con el propósito general de disminuir el tiempo de cotejo de los lofoscopistas en el CTI.

1.4.2. Objetivos Específicos

- Establecer las configuraciones óptimas para la captura y digitalización de las imágenes.
- Determinar las condiciones apropiadas de afinación del filtro de Gabor, para obtener un buen realce de las crestas de la huella.
- Localizar las bifurcaciones y terminaciones en las crestas dactilares, con su respectivo valor de orientación.
- Analizar distintos métodos de cotejo, basado en minucias y correlación de huellas dactilares, para determinar la correspondencia entre ellas.
- Desarrollar una interface fácil e intuitiva de manejar.
- Establecer la tasa de falsa aceptación y la de falso rechazo para evaluar el desempeño del sistema.

Capítulo 2

2. MARCO TEÓRICO

2.1. BIOMETRÍA

2.1.1. Introducción [5]

La biometría proviene de las palabras bio “vida” y metría “medida”, es decir la ciencia que estudia las medidas características de una persona.

Estas medidas deben ser encontradas en todas las personas, aunque el simple hecho de medir la altura de una persona es una medida biométrica, sin embargo no tendría un uso práctico porque muchas personas tienen la misma altura. Actualmente los estudios en biometría buscan establecer la identidad de una persona, basado en sus características físicas, químicas o de comportamiento. La importancia de esto se puede ver en sus aplicaciones, las cuales involucran por ejemplo la identificación de los individuos de un país, el control de acceso a áreas privadas como plantas nucleares, bancos, edificios gubernamentales, acceso a información por internet, hasta el simple hecho de entrar a una casa residencial. Se puede entender también que las medidas biométricas nos hacen una llave única e irrepetible.

Tradicionalmente los métodos para determinar la identidad de una persona se basa en claves, tarjetas, llaves, etc. Las cuales pueden ser fácilmente pérdidas, olvidadas, y muy factible la suplantación por otra persona, es aquí donde la biometría juega su papel importante.

Existen diversos tipos de medidas biométricas como el reconocimiento de las huellas, el ADN, el iris, la voz, etc. Las cuales poseen todas las personas, porque son partes de su cuerpo, en la actualidad se usan sistemas automáticos para la comparación de patrones biométricos, que son rápidos, alta confiabilidad y difíciles de suplantar. Examinemos entonces los sistemas biométricos.

2.1.2. Sistemas Biométricos

Un sistema biométrico es básicamente un sistema de reconocimiento de patrones, que reconoce la identidad de una persona por sus características físicas ó de comportamientos. Dependiendo de su aplicación se puede clasificar en *sistemas de verificación* o *sistemas de identificación* [2], la primera realiza la comparación entre dos patrones biométricos, uno de la persona y el otro guardado previamente, el resultado es el rechazo o la aceptación que los dos patrones sean iguales; el segundo realiza la comparación entre n patrones biométricos, uno de la persona y el otro de una base de datos de los patrones biométricos, el resultado es la determinación de *a quién* pertenece el patrón.

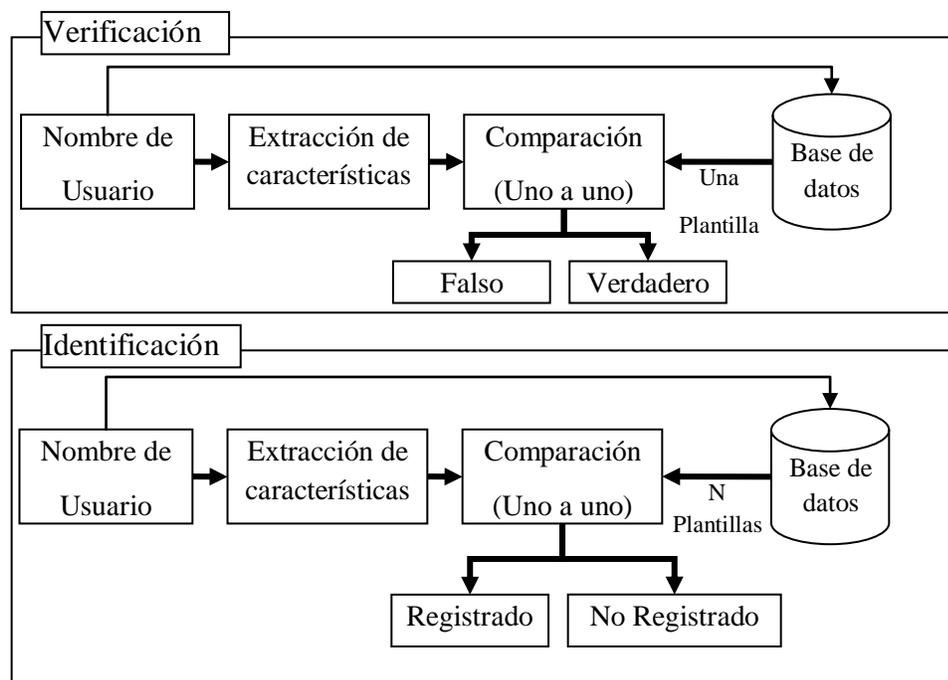


Figura 1. Diagrama de flujo de sistemas de verificación e identificación.

Fuente: Diagrama de flujo de Maltoni[2]

Para entender mejor cómo funcionan los sistemas de verificación e identificación, observe la Figura 1, el esquema fue tomado de [2], en el primer diagrama un usuario intenta ingresar con sus datos, entonces el sistema busca el nombre de usuario de la persona y compara el patrón biométrico, esto lo realiza uno sola vez, el resultado es la aceptación o el rechazo de la persona, en la práctica se utiliza para tener control sobre el acceso de personas a áreas privadas.

En el segundo diagrama un usuario intenta ingresar con sus datos, entonces el sistema busca el patrón biométrico de la persona y lo compara con la base de datos tenga, hasta encontrar el registro de la persona, en caso de no encontrarla se dirá que el usuario no está registrado en el sistema, uno de los usos principales de este sistema es para determinar

quienes entran a determinadas áreas, en el caso de los sistemas periciales (*como el C.T.I*) es útil para determinar la identidad de una huella latente.

En general ambos casos siempre se examina un par de patrones y establecen si son o no iguales, la pregunta que surge es, ¿Qué clases de medidas biométricos hay?. Para ello veamos las más comunes son:

1. Huellas Dactilares.
2. ADN.
3. Rostro.
4. Geometría de las manos.
5. Termogramas infrarrojos de las Venas de las Manos, Manos y Rostro.
6. Iris.
7. Firma.
8. Voz.

De las cuales vamos a tomar, vamos a ver un poco más las huellas, el rostro, el iris, la voz y las manos.

2.1.2.1. Huellas Dactilares

Las huellas dactilares se forman en el epidermis de la piel de los dedos, está formado por un patrón intercalado de crestas y valles. La formación de esta huella al igual que cualquier parte del cuerpo depende de las características genéticas y medio ambientales.

La toma de huellas dactilares es algo muy simple de hacer, solo utilizando tinta y papel; sin embargo hoy en día se cuenta con sensores electrónicos para el registro de la huella, la cual almacena en forma digital las huellas, esta imagen es utilizada por algoritmos que utilizan técnicas basadas en minucias, crestas y correlación para comparar la huella con otro patrón, la principal ventaja de esta medida, es la fácil adquisición de la huella, los tiempos de comparación son cortos y el equipo utilizado es muy económico. Actualmente se encuentran muchas aplicaciones en esta área que involucran control de acceso a zonas restringidas, aplicaciones civiles (aplicadas al registro de identificación personal), y forenses (que sirven para determinar la identidad de una persona). Existen muchas empresas que ofrecen sistemas AFIS los cuales se pueden ajustar para ser robustos y/o precisos dependiendo en que se apliquen.

2.1.2.2. Rostro

La forma de nuestro rostro es uno de las formas que usualmente utilizamos para identificar una persona de otra, sin embargo a nivel de algoritmo, el reconocimiento de rostros no es

intrusivo (*ilustrativo*). Es un reto desarrollar técnicas de reconocimiento de rostros que puedan tolerar los efectos de las expresiones faciales, el envejecimiento, y las variaciones de pose del rostro respecto a la cámara. Sin embargo los algoritmos trabajan bien cuando se tiene un rostro en condiciones adecuadas, en ella se buscan proporciones entre los ojos, cejas, ancho de la boca, la nariz y el mentón, también se usa el análisis del rostro como una combinación ponderada de un número de caras canónicas. En la actualidad existen programas comerciales que pueden comparar un rostro con una base de rostros, pero tienen limitaciones en la iluminación y la pose del rostro [2, 5].

2.1.2.3. Geometría de las Manos

Se han encontrado individualidad en la forma de las manos, en ellas se busca medidas que incluyen la forma, tamaño de la palma, largo y ancho de los dedos, y en algunos casos incluyen en las proporciones de esta. Los algoritmos desarrollados son más simples que para el resto de las medidas biométricas, ya que no tienen en cuenta anomalías como rasguños en la mano, color de piel, sudor, etc. Debido a que se usa principalmente la silueta, comercialmente se encuentran cientos alrededor del mundo, la desventaja principal que del reconocimiento de las manos es la escala, ya que la forma de la mano de una persona cambia de acuerdo al crecimiento, además los dispositivos en general tienen que ser muy grandes para tomar toda la mano, aunque algunos sistemas utilizan solo la información de los dedos, lo cual hace al sensor más pequeño [2, 5].

2.1.2.4. Iris y Escaneo Retinal

La textura del ojo es determinada por la morfo-genética caótica durante el desarrollo del embrión, por ello inclusive entre hermanos gemelos se pueden encontrar grandes diferencias, ya que cada uno de ellos crece en condiciones levemente diferentes, la textura del ojo continua cambiando hasta los dos años, después de ello se estabiliza, sin embargo el pigmento sigue cambiando durante toda la vida. La captura del iris se hace mediante cámaras de buena resolución y con adaptación óptica, por ello al principio fueron costosas, pero su costo ha reducido considerablemente. Estos dispositivos son extremadamente precisos y rápidos, además es posible detectar lentes de contacto con una impresión falsa del iris.

La retina vascular del ojo es rica en su estructura y se puede considerar como única en cada persona. Actualmente es uno de los sistemas biométricos más seguros que existen, ya que no es fácil cambiar o replicar la retina vascular, debido a que puede contener hasta 400

puntos característicos. Los precios son muy exequibles al público en general, se pueden comprar por 500 (1'000.000 pesos) dólares en el mercado [2, 5].

2.1.2.5. Voz

Siempre que una persona nos habla por teléfono, nosotros podemos determinar a quién pertenece esa voz. Las características de la individualidad de la voz dependen de la forma y tamaño de las cuerdas vocales, boca, cavidad nasal y labios, es por ello que es aceptado como una medida biométrica en la mayoría de las sociedades. Sin embargo no puede ser utilizado para identificar a una persona de gran base de datos de identidades, además la voz se degrada con el tiempo, depende del estado emocional, la salud (*resfriados*), etc., y también se encuentran algunas personas con una extraordinaria habilidad para imitar la voz de otros, como es el caso de la mayoría de cómicos [2, 5].

2.1.2.6. Comparación de las Tecnologías Biométricas

Para la comparación de las tecnologías se van a tomar diferentes criterios de calificación, los cuales son:

1. Universalidad: indica que tan común es encontrar la medida biométrica.
2. Singularidad: indica que tan único o diferenciable es una medida biométrica con respecto a otra.
3. Permanencia: indica el tiempo de permanencia de la medida biométrica en la persona sin sufrir cambios.
4. Colectividad: la facilidad para adquirir, almacenar y medir el patrón biométrico.
5. Rendimiento: indica que tan preciso, rápido y robusto es el sistema en el manejo de la medida biométrica.
6. Aceptabilidad: indica la aprobación de la tecnología biométrica en el público.
7. Fiabilidad: indica que tan fácil es engañar al sistema biométrico.

En la Tabla 1, hay un resumen de las ventajas y desventajas de las principales tecnologías biométricas, se puede ver que las huellas dactilares tienen un buen balance, cabe destacar que el rendimiento es muy deseable por las empresas, que suelen utilizar una tecnología biométrica para controlar el acceso a personas. También el desempeño general del reconocimiento del ADN y el iris es bueno y en algunos casos es superior al de las huellas dactilares, debido a la cantidad de puntos característicos que se analizan.

Tabla 1. Comparación entre diversas tecnologías biométricas.
Fuente: libro de *Handbook of Fingerprint Recognition 2003* [2].

Sistema biométrico	Universalidad	Singularidad	Permanencia	Colectividad	Rendimiento	Aceptabilidad	Fiabilidad
ADN	Alto	Alto	Alto	Bajo	Alto	Bajo	Bajo
Rostro	Alto	Bajo	Medio	Alto	Bajo	Alto	Alto
Geometría de manos	Medio	Medio	Medio	Alto	Medio	Medio	Medio
Termograma facial	Alto	Alto	Bajo	Alto	Medio	Alto	Bajo
Venas de Manos	Medio	Medio	Medio	Medio	Medio	Medio	Bajo
Iris	Alto	Alto	Alto	Medio	Alto	Bajo	Bajo
Retina	Alto	Alto	Medio	Bajo	Alto	Bajo	Bajo
Firma	Bajo	Bajo	Bajo	Alto	Bajo	Alto	Alto
Voz	Medio	Bajo	Bajo	Medio	Bajo	Alto	Alto
Huellas Dactilares	Medio	Alto	Alto	Medio	Alto	Medio	Medio

Sin embargo en la Figura 2, se puede observar que la demanda de sistemas biométricos automáticos, es liderada por el reconocimiento de huellas dactilares, principalmente por entidades gubernamentales que utilizan los sistemas AFIS, con un 38,3% en sistemas AFIS y un 28,4% en identificación civil, las entidades de identificación civil (como la *Registraduría Nacional de Colombia*), y empresas que usan para el control a áreas restringidas. Otro sistema biométrico importante es el reconocimiento del rostro con 11,6%.

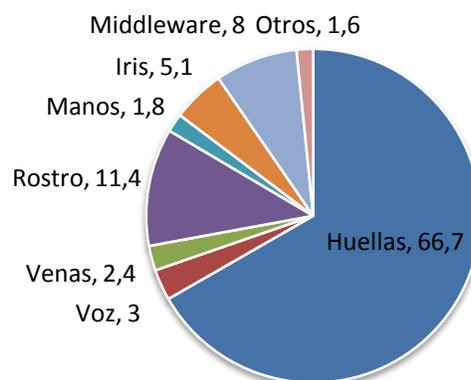


Figura 2. Reporte de mercadeo en biometría.
Fuente: libro de *Biometrics Market and Industry Report* [6].

2.1.2.7. Aplicaciones

Existen diversas aplicaciones para los sistemas biométricos, ya se han nombrado algunas antes. La necesidad de conocer *quién*, entra o sale de determinado lugar, es autorizado para entrar, y quien soy yo, son algunos de los problemas que se están solucionando con los sistemas automáticos de reconocimiento biométrico, Jain en [5] nos indica quienes son los principales usuarios de éstos sistemas, estos se pueden resumir en la Tabla 2.

Tabla 2. Principales usos de la tecnología biométrica.
Fuente: libro de Biometrics Market and Industry Report [6].

Forense	Gubernamental	Comercial
Identificación de Cadáveres	Tarjeta de Identificación	ATM (<i>modos de transferencia asíncrona</i>)
Investigación Criminal	Licencias de Conducción	Control de Acceso, clave de PC
Determinación de parentesco	Asistencia Social	Celulares
Personas perdidas	Inmigración, registro del voto electoral.	Comercio Electrónico, Internet, Bancos, tarjetas inteligentes

2.2. HUELLAS DACTILARES

En esta sección se va a estudiar a fondo las huellas dactilares, debido a que es el problema que se va a solucionar mediante un sistema automático de reconocimiento de huellas. Los temas a tratar son: formación de la huella, adquisición de huellas, clasificación de huellas.

2.2.1. Formación de la Huella Dactilar

Las huellas dactilares se forman en el epidermis de la piel de los dedos, está formado por un patrón intercalado de crestas y valles (surcos), como se ve en la Figura 3 . La formación de esta huella al igual que cualquier parte del cuerpo depende de las características genéticas y medio ambientales.



Figura 3. Forma de una huella dactilar en el dedo sano.

Fuente: http://commons.wikimedia.org/wiki/File:Fingerprint_detail_on_male_finger.jpg

El ADN contiene las instrucciones generales para la formación de la huella, alrededor del 7 mes de gestación del feto, las huellas toman su forma final y continuará sin cambios en toda su vida, sin embargo la forma específica de las crestas y minucias está influenciado por otros factores como la posición del bebe, la cantidad de liquido amniótico, la temperatura, presión, etc., éstas razones hacen que las huellas dactilares sean diferentes inclusive entre hermanos gemelos [2, 7].

La diferencia del ADN de un par de gemelos no puede ser hallada, también la distinguibilidad de sus rostros, las formas de sus manos, voces y cuerpos no se puede ó es difícil de determinar por sistemas automáticos de reconocimiento de éstos patrones. Sin embargo la distribución de las minucias en el dedo es diferente, aunque comparte el mismo tipo de dactilograma (*ver, sección 2.2.2*), ancho y separación de las crestas.

Otro atractivo del reconocimiento de las huellas dactilares se atribuye a que su formación no cambia, si el dedo sufre alguna herida leve, las crestas se reconstruyen igual que su forma original, existirá un cambio cuando la herida es grave, como cortaduras profundas, dermatitis severa, quemadura de alto grado, etc., para el caso de la dermatitis severa el registro de las huellas no es conveniente debió al desgaste, sus huellas son borradas, en el caso de las cortaduras las cicatrices formadas pasarán a ser parte de su identidad, por último para el caso de las quemaduras la huella se formará si su quemadura no afecta el epidermis de la piel.

2.2.2. Clasificación de las Huellas Dactilares [8]

El sistema de clasificación y comparación de huellas, está conforme al *Decreto 1216 de Julio 04 de 1935*, donde se ratificó y unificó al sistema dactiloscópico Henry Canadiense. El método de categorización que implementó Sir Edward Henry fue hecho en 1900, el redefinió al sistema de clasificación de Galton, el cual había clasificado las huellas en 3 principales clases: arcos, presillas y espirales. Henry sin embargo amplió esta

clasificación y fue optado por varios países en el mundo, de hecho casi todos los esquemas de clasificación alrededor del mundo son variantes del sistema Henry.

De acuerdo a este sistema las huellas se clasifican en 8 dactilogramas (es decir en las formas que toman las crestas), donde se hace la distinción de los nombres en español (colombiano) e inglés (internacional), ya que la mayoría de AFIS usan estos nombres, en la Figura 4 se pueden ver los posibles dactilogramas, ellos se basan principalmente en la distribución de deltas y núcleos, los cual son representados por círculos y triángulos, respectivamente, y como ellos se distribuyen; de todos ellos la más difícil de encontrar y categorizar es la accidental, por la forma inusual de sus crestas que se pueden confundir con las presillas, además pueden contener más de 2 núcleos o deltas, los cuales son raros de encontrar.

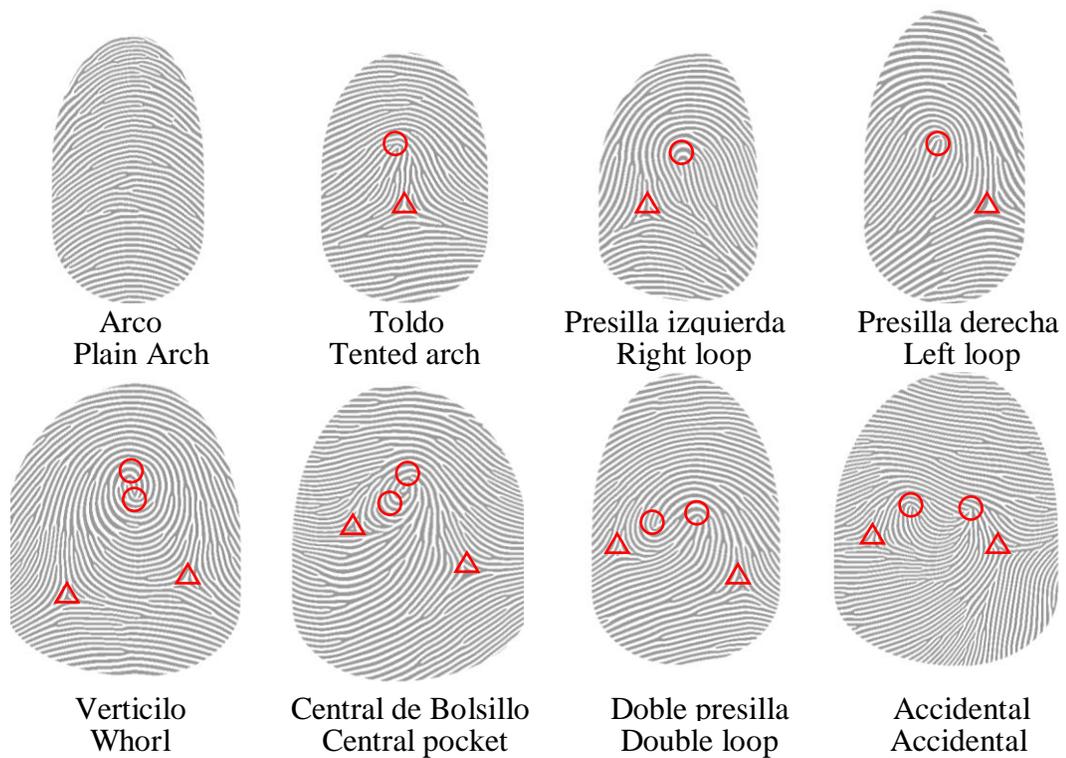


Figura 4. Tipos de dactilogramas

Fuente: Las imágenes fueron creadas por el programa SFinGe.

Cuando se realiza un “cotejo”, que es la comparación entre dos huellas, se establece primero la igualdad del dactilograma, si estos coinciden se buscan las minucias en la huella. En los AFIS ó cualquier tipo de sistema automático de búsqueda de minucias, se manejan principalmente dos tipos: las bifurcaciones y las terminaciones, de aquí se desprenden todas las formas de minucias que se usan en el sistema Henry, estas se pueden ver en la Figura 5, aunque en realidad se pueden encontrar otro tipo de minucias muy particulares, por ejemplo una bifurcación de donde se desprendan tres crestas, estas formas se encuentran

principalmente en el núcleo o el delta de la huella, y se encuentran aun más en huellas de tipo accidental.

Una de las minucias interesantes y que puede cambiar a lo largo de la vida, son las cortaduras, porque dependen de las condiciones a las que se exponen la yema de los dedos, aunque la cicatriz puede perdurar pueden aparecer nuevas cicatrices.



Figura 5. Tipos de minucias dentro de la huella.

2.2.3. Medidas de Errores en los Sistemas de Verificación

Para la evaluación del rendimiento de un sistema de verificación de huellas, y en general para sistemas biométricos, puede ser formulado como sugiere [2].

Llamemos a una huella de un nuevo usuario como S (*Sospechosa*) y la huella almacenada en los registros como C (*Conocida*), entonces la hipótesis nula y alternadas son:

H_0 : $C \neq S$, la huella de entrada NO es la misma que la huella almacenada.

H_1 : $C = S$, la huella de entrada SI es la misma que la huella almacenada.

Las decisiones asociadas a estos dos casos son:

D_0 : $C \neq S$, la persona es un impostor.

D_1 : $C = S$, la persona es genuina.

Entonces una hipótesis que pruebe la formulación inherente contiene dos tipos de errores: tipo I: *Falsa Aceptación*, cuando D_1 es decidido cuando H_0 es verdad, y tipo II: *Falso Rechazo*, cuando D_0 es decidido cuando H_1 es verdad, estos dos errores son conocidos también como:

- FAR (*False Accept Rate*) ó FMR (*False Match Rate*), la tasa de falsa aceptación se presenta por ejemplo: cuando un usuario NO autorizado es aceptado por el sistema como un usuario SI autorizado. Se calcula como se ve en la ecuación (1), donde N_{FA} es el número de falsas aceptaciones y T_{FA} es el número total de falsas aceptaciones:

$$FAR = \frac{N_{FA}}{T_{FA}} \quad (1)$$

- FRR (*False Reject Rate*) ó FNMR (*False Non-Match Rate*): la tasa de falso rechazo se presenta por ejemplo: cuando un usuario SI autorizado es rechazado por el sistema como un usuario NO autorizado. Se calcula como se ve en la ecuación (2), donde N_{FR} es el número de falsos rechazos y T_{FR} es el número total de falsos rechazos:

$$FRR = \frac{N_{FR}}{T_{FR}} \quad (2)$$

Las graficas de la FAR y FRR, ver la Figura 6, brindan información sobre como ajustar el sistema para distintas aplicaciones como: aplicaciones civiles, identificación y control de acceso. En [9] se sugiere que valores bajos de FAR y altos de FRR el sistema se vuelve más seguro, y puede ser aplicado a sistemas robustos de control de seguridad, aunque esto implica una mayor tasa de falso rechazo, y una persona cuando intente ingresar necesitará varios intentos para determinar si es verdadero o falso usuario.

Cuando se invierten estos valores, es decir FAR bajos y FRR altos se puede usar para la identificación de criminal, donde se arrojan posibles huellas que pertenezcan a una huella de entrada, el estado intermedio de este es utilizado para aplicaciones civiles, donde la alta seguridad no sea necesaria, las implicaciones son distintas a las anteriores, ya que ahora una persona puede tener más facilidad para acceder al sistema, porque hay más probabilidad que el sistema reconozca una persona no autorizada como autorizada.

Los valores de la FAR y FRR disminuyen y aumentan (*respectivamente*) a medida que el umbral cambia. Si se grafica la FRR vs la FAR se construye una curva denominada ROC (*Receiver Operating Characteristic*), que se usa para evaluar el desempeño del sistema y poderlo comparar con otros ROC de otros sistemas, con el umbral se puede ajustar qué tipo de aplicación se necesite, es decir la sensibilidad del programa para que reconozca una huella con otra.

También se pueden comparar usando la ERR (*Equal Error Rate*), el cual es un punto de intercepción entre la FAR y la FRR, en este punto se encuentran el mínimo de la falsa

aceptación y rechazó. Estas son las medidas básicas para caracterizar un sistema de reconocimiento, aunque existen otras medidas que pueden ser encontrados en [2] y [10].

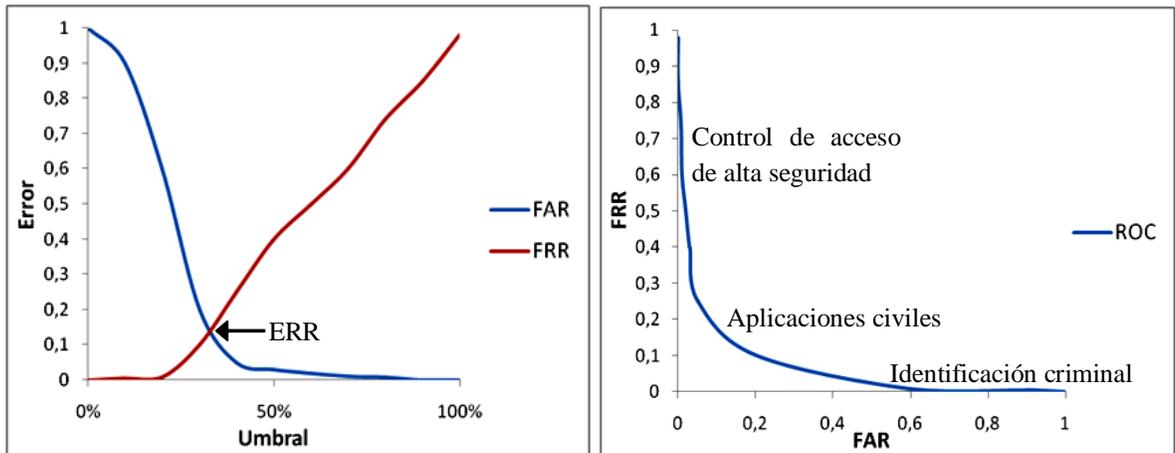


Figura 6. Típicas graficas de FAR y FRR, y la ROC.

2.3. ADQUISICIÓN Y PROCESAMIENTO

2.3.1. Adquisición de Huellas Dactilares

Existen dos tipos de clasificaciones para la adquisición de huellas según [2], las cuales son “off-line” y “live-scan”.

La adquisición off-line ó fuera de línea es la que se hace comúnmente con tinta y papel, también se presenta cuando se hacen levantamiento de huellas latentes (*huellas de escena del crimen*), y cualquier forma que no involucre dispositivos electrónicos. Por mucho tiempo y en nuestros días se ha usado este tipo de adquisición, para guardar registros de nuestra identidad, validación de documentos como cheques, actas, compraventas, etc., el equipo necesario para una buena adquisición de la impresión dactilar consta de planchuela, rodillo, tinta, tarjeta de registro, porta-tarjetas, guates, gasa y aseptidina ó alcohol; como se ve en la Figura 7.

adquisición live-scan, que se pueden clasificar en: sensores ópticos, de estado sólido y ultrasónicos, ellos pueden ser del tipo toque (*touch*), cuando se introduce todo el dedo en el aparato; ó de deslizamiento (*sweep*), cuando se desliza por una ranura del dispositivo, esto se encuentra en [2] y se puede resumir según el principio físico de funcionamiento:

Reflexión interna total frustrada (óptico)

Es uno de los más comunes dispositivos, que usa una luz emitida por LEDs para iluminar de forma oblicua un prisma, cuando la luz se encuentra con una cresta la luz es reflejada al sensor, si es un valle la luz no se refleja.

Capacitivo (estado sólido)

Estos están contruidos como una matriz de micro-capacitores embebidos en un chip, cuando una cresta toca uno de estos capacitores su capacitancia cambia y con ello el potencial eléctrico, con ello se reconstruye la imagen, el funcionamiento similar al funcionamiento del teclado del PC.

Térmico (estado sólido)

Estos sensores están hechos de un material piro-eléctrico, que genera una corriente basado en las diferencias de temperatura entre los valles y surcos de la huella, la imagen aparece cuando se hace contacto y desaparece rápidamente porque la temperatura se estabiliza, es por ello que estos deben ser de tipo sweep

Piezoeléctrico (estado sólido)

Estos son sensores de presión, que cambian su voltaje cuando una fuerza es aplicado sobre él. Su funcionamiento es parecido al capacitivo, con la desventaja que no es suficientemente sensible, y produce imágenes binarias.

Ultrasónico (sonido)

Su funcionamiento se basa en el efecto dopler (como la ecografía), tiene un transmisor que emite un onda hacia la huella, y un receptor que capta las diferencias de tiempo de viaje de la onda, siendo mayores en los surcos. Tienen una alta sensibilidad y son muy resistentes a suciedad, y el aceite que tienen los dedos, sin embargo aun son muy costosos.

Para darse una idea de cómo es la forma de estos dispositivos, se puede observar en la Figura 9 los principales dispositivos comerciales, siendo el óptico el más utilizado por su costo bajo, y el térmico por su tamaño reducido el cual se encuentra en muchos dispositivos portátiles.

				
Tipo	Óptico	Capacitivo	termico	Piezoelectrico
Fuente	www.biometrika.it	www.atmel.com	www.st.com	www.bm-f.com

Figura 9. Clases de sensores electrónicos comerciales.

Todos estos dispositivos entregan una imagen digital, que se caracteriza principalmente por su resolución, la cual se relaciona con área de escaneo y el número de píxeles, existen otros parámetros para la caracterización como el rango dinámico, la precisión geométrica y la calidad de la imagen.

La resolución juega un papel importante a la hora de tomar una huella, porque cuanto mayor sea la resolución más detalles podremos observar, la medida de esto es el dpi (dots per inch), es decir la cantidad de puntos por área que se puede cuantificar, las investigaciones del FBI y el NITS sugieren que 500 dpi es una medida adecuada para escanear una huella, sin embargo existen empresas que venden sensores con resoluciones de 300 dpi, con el fin de ocupar menos memoria para las imágenes. Existe una relación entre el tamaño de la imagen, la resolución y el área censada en milímetros como:

$$\text{Tamaño} = [r (f/25.4), r (c/25.4)]$$

Donde f y c son el número de filas y columnas. Esta fórmula nos sugiere que cuanto mayor es la resolución del dispositivo, tendremos imágenes más grandes y con más espacio en memoria; tomemos por ejemplo un área promedio de la huella de $20 \times 16 \text{ mm}^2$ y una resolución de 500 dpi, entonces la imagen generada tendrá [394, 315] píxeles, si aumentamos a 1000 la resolución, la imagen será de [787, 630] píxeles, esta resolución es utilizada cuando se quieren ver detalles como los poros las crestas, el cual es un nuevo tema de investigación que no se tratará en este trabajo. Una imagen tomada a distintas resoluciones puede empeorar o mejorar la calidad de la imagen, como se ve en la Figura 10, donde A es a 650 dpi, B a 500 dpi, C a 400 dpi, D a 300 dpi y E a 150 dpi. Las dimensiones son proporcionales, se puede observar como las minucias de la huella se han perdido ó no se pueden determinar con certeza para bajas resoluciones, y para muy altas los poros son muy notorios, por ello la resolución de 500 dpi es la mejor ya que no deja ver muchos poros y las crestas se pueden diferenciar muy bien.

Una vez obtenida la imagen, se debe guardar en un formato digital el cual preserve la estructura de la huella para posteriores usos. Los formatos digitales más populares para guardar imágenes son JPEG, PNG, TIFF, GIF, BMP, entre otros, algunos de estos métodos causan pérdida de información u ocupan demasiados bits cuando se guardan, de todos ellos el formato con buenas balance entre calidad y tamaño de archivo es el JPEG, sin embargo

el FBI con ayuda del NITS preocupados por el creciente aumento en las tarjetas de registro dactilar, que en 1995 alcanzaban los 200 millones, desarrollaron un formato digital llamado *cuantización escalar de onda pequeña (Wavelet Scalar Quantization)*, el cual ofrece mejores más compresión y menos pérdida de información que el JPEG.

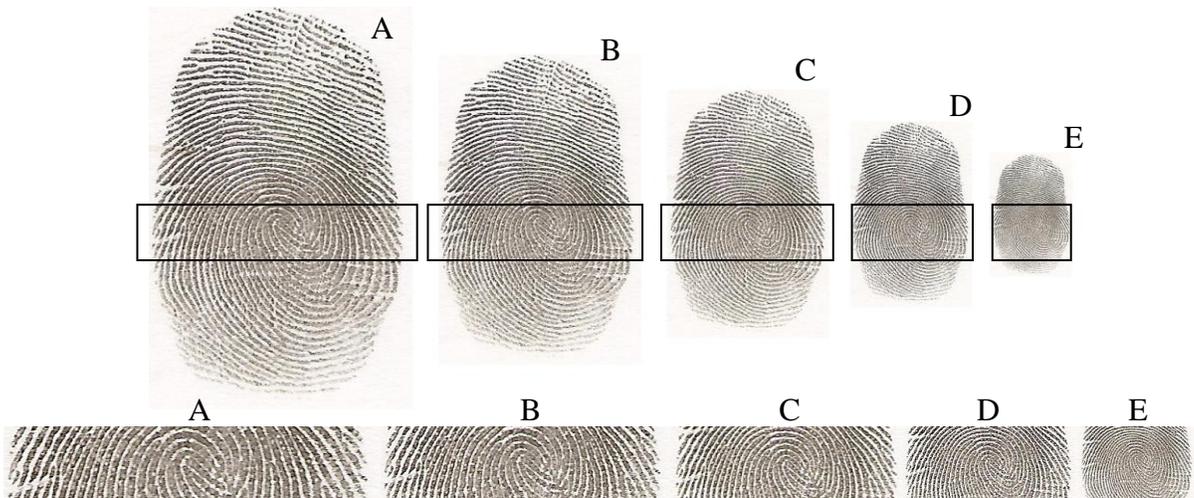


Figura 10. Huella tomada a distintas resoluciones.

2.3.2. MEJORAMIENTO

2.3.2.1. Trabajos Relacionados

El mejoramiento es el primer problema al que se enfrentan los sistemas de reconocimiento de huellas, se realiza debido a que la mayoría de las imágenes tienen ruido, para el caso de las huellas, el ruido se presenta por varios factores, en la adquisición *off-line*, se da por la mala reseña de la huella, la calidad de la tinta, el papel, la presión, la humedad y la colaboración del reseñado. Para el caso *live-scan* el ruido se da por la no linealidad de los sensores, la resolución, suciedad en los sensores, ruido electromagnético, etc., cualquiera sea la razón, las imágenes deben pasar por un filtrado digital para limpiar la huella.

La mayoría de algoritmos utilizan la medida de la varianza para segmentar la imagen [11], porque es fácil y rápido de implementar y los resultados son muy buenos; otros autores utilizan por ejemplo los puntos de esquina característicos de Harry [12] y usando un mapa de coherencia [13], con ello se busca disminuir el tiempo de computo eliminando zonas de la imagen que no pertenezcan a la huella.

Una vez segmentado se puede aplicar uno de los primeros procesos para el mejoramiento, se suele utilizar la normalización [11], dilatación del contraste y la ecualización [14], todos ellos buscan preparara la imagen para posteriores tratamiento.

Para el mejoramiento se utilizan filtros adaptativos en dirección y frecuencia de las crestas, por ello primero se debe estimar la orientación, los métodos utilizados se basan en gradientes [7, 11, 15], en los cuales se proponen varios métodos para mejorar la estimación de la orientación, como los trabajos de [16], muy pocos se basan en otros métodos, en [2] se pueden encontrar otras referencias.

La estimación de la frecuencia inicialmente no se tenía en cuenta, simplemente se usaba una frecuencia igual para todas las huellas [7, 17-19], no obstante las crestas no tienen siempre la misma distancia entre una y otra, Hong [20] fue uno de los primeros en proponer la medida de la frecuencia local en las crestas, y es usado ahora por otros autores con sus propias consideraciones [11, 21].

Por último la etapa del filtrado de la huella, inicialmente y hasta ahora se usa el filtro de Gabor [20-24], debido a su alto rendimiento e inmunidad al ruido, sin embargo se han propuesto una gran variedad de filtros para el mejoramiento de la imagen, entre ellos están los filtros anisotrópicos [25], el filtro de mediana direccional [26], el procesamiento por histéresis diferencial [27], combinación entre filtros isotrópicos y anisotrópicos [28], también encontramos el filtrado en el espacio de Fourier [29]. En todos estos filtros mencionados usan de alguna forma la estimación de la frecuencia y la orientación, para obtener buenos resultados en la reconstrucción de la huella.

2.3.2.2. Algoritmo de Mejoramiento

Para realizar esto muchos autores sugieren el diagrama de flujo de la Figura 11 [2, 7, 11, 15, 20, 21], el esquema puede llevar algunos procesos internos que no se mencionan. Con base en ello se realiza el algoritmo de mejoramiento con los siguientes pasos:

1. Imagen de entrada.
2. Segmentación.
3. Normalización.
4. Estimación de la orientación.
5. Estimación de la frecuencia local.
6. Filtrado.



Figura 11. Diagrama de flujo para el mejoramiento de las huellas dactilares.

2.3.2.3. Segmentación

Este proceso consiste en separar el fondo y segmentar el área de interés, la cual es el área donde están contenidas las crestas y valles de la huella dactilar, el fondo es simplemente la superficie que esta fuera del borde de la imagen de interés.

Para ello se analizan los valores de la varianza local de la imagen $I(i, j)$, se divide la imagen $I(i, j)$ en bloques k de tamaño $N \times N$, se analiza la varianza de cada bloque y se fija un umbral global, si el valor es menor que este umbral se establece como fondo de la imagen (negro), de lo contrario se establece como imagen de interés (blanco), con esto se crea una imagen binaria la cual es usada como máscara para el resto de operaciones.

La varianza para cada bloque k de dimensiones $N \times N$ se define como la ecuación (3)

$$V(k) = \frac{1}{N^2} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} (I(i, j) - M(k))^2 \quad (3)$$

2.3.2.4. Mejora del Histograma

Para preparar la imagen para los procesos de estimar la orientación y frecuencia, se debe adecuar los niveles de gris de la imagen, de forma tal que las crestas y los surcos estén bien diferenciados.

Existen dos tipos de mejora del histograma que son utilizadas, la primera que hace la ecualización del histograma como se indica en [14] y la segunda llamada normalización mediante la media y la varianza [9].

El primer algoritmo utiliza la función de OpenCV *cvEqualizeHist*, este método realiza las siguientes operaciones:

1. Calcula el histograma H de la imagen de entrada $I(i, j)$.
2. Normaliza el histograma.
3. Computa la integral del histograma, tal que la nueva imagen $H'(i) = \text{suma}_{0 \leq j \leq i} H(j)$.
4. Transforma la imagen usando H' como una LUT, tal que la imagen de destino $E(i, j) = H'(I(i, j))$.

Esta imagen la llamaremos de aquí en adelante como imagen ecualizada $E(i, j)$, en los resultados se modifica la función *cvEqualizeHist* para que acepte el parámetro de la mascarar de segmentación la cual se llamara $E_{mask}(i, j)$.

El segundo algoritmo el cual se ha reportado en la mayoría de la literatura [9, 26, 28] usa la normalización del histograma, mediante la ecuación (4), es más simple que la anterior y rápida de implementar.

$$N(i, j) = \begin{cases} M_0 + \sqrt{\frac{V_0(I(i,j)-M)^2}{V}} & \text{si } I(i, j) > M \\ M_0 - \sqrt{\frac{V_0(I(i,j)-M)^2}{V}} & \text{otros casos} \end{cases} \quad (4)$$

Donde M_0 y V_0 es la media y varianza deseada, M y V es la medida de la media y la varianza de la imagen I , y $N(i, j)$ es la imagen normalizada. Este método no altera la estructura de las crestas, el propósito de esto es la reducción de la variación en los niveles de gris entre de las crestas y los surcos, con el fin de mejorar los procesos siguientes.

2.3.2.5. Estimación de la Orientación

Este es un paso fundamental para aplicar el filtrado, ya que éste parámetro sirve para afinar la orientación del filtro de Gabor con las crestas.

Para calcular la orientación de cada pixel de la imagen $I(i, j)$, el algoritmo se baso en el algoritmo de Hong [20], para ello se realiza las operaciones (5), (6), y (7), como se ven a continuación:

$$V_x(i,j) = 2\delta_x(i,j)\delta_y(i,j) \quad (5)$$

$$V_y(i,j) = \delta_x^2(i,j) - \delta_y^2(i,j) \quad (6)$$

$$\theta(i,j) = \frac{1}{2} \operatorname{atan2}\left(\frac{V_x(i,j)}{V_y(i,j)}\right) + \frac{\pi}{2} \quad (7)$$

Donde δ_x y δ_y es la convolución de la imagen con un kernel direccional, estos pueden ser matrices de Sobel, Prewitt, ó Marr Hildreth, este último se trata de un operador gaussiano el cual da un suavizado al gradiente y se aplica en este trabajo, aunque implica un coste computacional extra, pero la reconstrucción final es mejor que usando los otros dos. Finalmente se usa *atan2* el cual es la función arco-tangente comprendida entre $[-\pi, \pi]$, con lo cual se obtiene que los ángulos de cero grados sean horizontales y los de 90 verticales.

En forma resumida el algoritmo debe aplicarse como

1. Aplicar a la imagen de entrada (imagen normalizada), los gradientes direccionales.
2. Se calcula V_x y V_y , en cada proceso del cálculo de cada uno de ellos se aplican filtros gaussianos, en el pseudocódigo, presente en la sección 3.3.3, se detalla cómo hacerlo.
3. Finalmente se aplica la función arco-tangente para estimar los ángulos de cada pixel de la imagen.

2.3.2.6. Matriz de Confiabilidad

Durante el cálculo de la orientación se estima la *matriz de confiabilidad* ($C(i, j)$), basado en la coherencia en los gradientes δ_x y δ_y y los momentos de inercia, el algoritmo es basado en el trabajo de Peter [21]. Donde se calcula el momento de inercia sobre el eje axial de orientación (8) (el cual será la mínima inercia), y sobre la perpendicular a la axial (9) (el cual será la máxima inercia). Entonces la matriz de *confiabilidad* ($C(i, j)$) se encuentra con la ecuación (10).

$$I_{\min}(i, j) = \frac{\delta_x^2(i, j) + \delta_y^2(i, j)}{2} - \frac{\delta_x^2(i, j) - \delta_y^2(i, j)}{2} \frac{\delta_x(i, j)\delta_y(i, j)}{\Gamma} - \frac{\delta_x(i, j)\delta_y(i, j)}{2} \frac{\delta_x^2(i, j) - \delta_y^2(i, j)}{\Gamma} \quad (8)$$

$$I_{\max}(i, j) = \delta_x^2(i, j) + \delta_y^2(i, j) - I_{\min}(i, j) \quad (9)$$

$$C(i, j) = 1 - \frac{I_{\min}(i, j)}{I_{\max}(i, j) + \varepsilon} \quad (10)$$

Donde Γ :

$$\Gamma(i, j) = \sqrt{(\delta_x^2(i, j) - \delta_y^2(i, j))^2 - (\delta_x(i, j)\delta_y(i, j))^2} \quad (11)$$

Finalmente para obtener una máscara de confiabilidad, para todos los valores $C(i, j)$ tales que $\Gamma(i, j) > \varepsilon$, se computan $\Gamma(i, j) * C(i, j)$, finalmente se binariza fijando un umbral β y se obtiene una máscara binaria $Mc(i, j)$. En el pseudocódigo en la sección 3.3.3.1 se explica con

detalle cómo hacerlo y esta máscara es llamada *mask2*, y los resultados de la nueva máscara se pueden ver en la Figura 37.

2.3.2.7. Estimación de la Frecuencia

Existen algoritmos que usan una frecuencia estable para todas las huellas, basado en la resolución del sensor ó el espacio promedio entre crestas [7, 17-19], sin embargo los resultados muestran que mantener una frecuencia única no permite reconstruir adecuadamente las huellas, podemos ver esto en la Figura 43. Para el desarrollo de este algoritmo se usan las consideraciones de [11, 20].

Para determinar la frecuencia local, primero se divide la imagen normalizada $N(i, j)$ en bloques de $[w, w]$, después cada bloque es rotado en dirección ortogonal a la orientación del bloque y, lo ampliamos en el numero de columnas, con dimensiones $[w, 2w]$, como se ve en la Figura 12. El ángulo θ de rotación se halla tomando un promedio de las orientaciones del bloque $[w, w]$ y se adiciona 90 para dejarlo ortogonal.

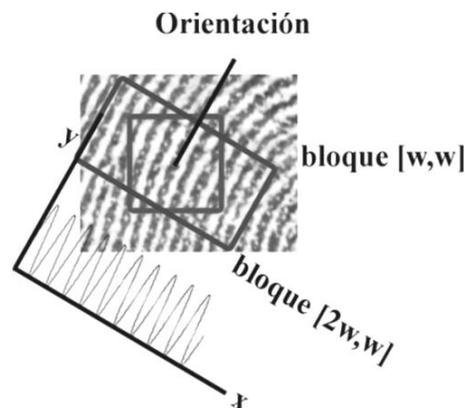


Figura 12. Procedimiento para el cálculo de la frecuencia local en la imagen.

Una vez rotado el bloque en $\theta + 90$, se puede sumar los valores de las columnas del bloque $[2w.w]$, en la cual se obtiene una distribución similar a una onda sinusoidal, como se ve en la Figura 13.

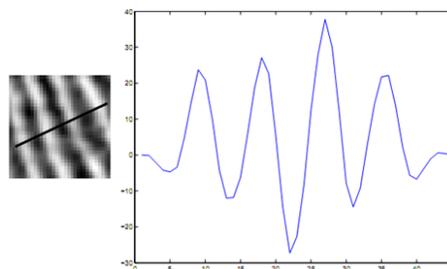


Figura 13. Intensidad de las crestas en dirección perpendicular a ellas.

Fuente: Imagen tomada de Thai [11].

Entonces se cuenta el número de crestas del bloque para hallar la frecuencia, finalmente se guarda cada bloque en una matriz $F(i, j)$ con la frecuencia local. Los resultados se pueden ver en la Figura 39.

2.3.2.8. Filtrado de Gabor

El paso más importante para la reconstrucción es el uso del filtro de Gabor, aunque este depende de las buenas estimaciones de las orientaciones y frecuencias.

La expresión matemática del para el filtro de Gabor es la ecuación (12), y sus parámetros x_θ e y_θ son las ecuaciones (13) y (14). Esta función se puede entender de la siguiente forma: *la función coseno genera una onda sinusoidal, este tren de ondas es enseguida multiplicado por una elipse gaussiana dada por la expresión exponencial, como se ve en la Figura 14.*

$$G(x_\theta, y_\theta, \omega) = \exp\left[-\frac{1}{2}\left(\frac{x_\theta^2}{\sigma_x^2} + \frac{y_\theta^2}{\sigma_y^2}\right)\right] \cos(2\pi\omega x_\theta) \quad (12)$$

$$x_\theta = x \cos(\theta) + y \sin(\theta) \quad (13)$$

$$y_\theta = -x \sin(\theta) + y \cos(\theta) \quad (14)$$

Donde ω es la frecuencia de la función, θ la orientación y la posición con $[x, y]$; los parámetros σ_x y σ_y son las desviaciones estándar de la envolvente gaussiana a en x e y .

Comúnmente se utilizaba los valores de σ_x y σ_y con valores fijos hallados empíricamente, como los utilizados en [9] de 4.0 y 4.0 respectivamente. Sin embargo en [11] se cambia esto, porque si se mantiene constante σ_x y σ_y se compromete a un ancho de banda constante, el cual no tendría en cuenta las variaciones de la frecuencia, por ello Thai [11] relacionó estos valores de desviación estándar con la frecuencia como $\sigma_x = k_x \omega^{-1}$ y $\sigma_y = k_y \omega^{-1}$. Donde k_x y k_y son unas constantes y ω es la frecuencia.

Con esto el filtrado es más flexible al ancho de las crestas, además con la desviación estándar se relaciona también las dimensiones de la matriz del filtro de Gabor, usado como kernel de convolución como: $w_x = \psi \sigma_x$ y $w_y = \psi \sigma_y$, donde ψ es una constante, en realidad este es el ancho total de la matriz de convolución, pero su rango debe ser $[-0.5\psi\sigma_y, 0.5\psi\sigma_y]$ y $[-0.5\psi\sigma_x, 0.5\psi\sigma_x]$.

Teniendo en cuenta que la orientación esta almacenada en cada pixel, entonces la convolución con la matriz de Gabor $G(i, j)$, debe ser afinado para cada pixel con $F(i, j)$ y $\theta(i, j)$ y multiplicarse con la imagen normalizada $N(i, j)$, como se ve en la ecuación (15).

$$M(i, j) = \sum_{u=-\frac{w_x}{2}}^{\frac{w_x}{2}} \sum_{v=-\frac{w_y}{2}}^{\frac{w_y}{2}} G(u, v; \theta(i, j), F(i, j)) N(i - u, j - v) \quad (15)$$

Donde $M(i, j)$ es la imagen mejorada, en esta ecuación no se ha normalizado los valores del filtrado, más adelante se verá la ventaja de hacer esto.

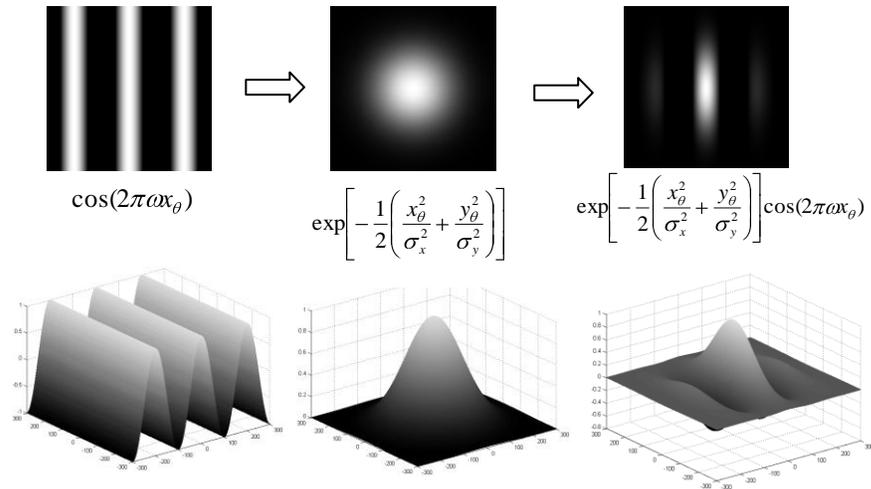


Figura 14. Expresión del filtro de Gabor en 2D y 3D.

2.3.3. EXTRACCIÓN DE CARACTERÍSTICAS

2.3.3.1. Trabajos Relacionados

Para comparar dos huellas, primero de deben extraer particularidades en cada huella y después cotejarlas. Algunos métodos utilizan la respuesta del filtro de Gabor en distintas direcciones [7, 17], están los que usan las características de la fase de la imagen [30], y la las minucias de la huella [7, 11, 15, 20], éste último se basa en como los dactiloscopistas comparan las huellas y se va a utilizar en este trabajo, por ello en este apartado se hablará de trabajos relacionados con las minucias, en la sección 2.3.4.1 se muestran otro tipo de extracción de características usadas para el cotejo entre huellas.

La extracción de minucias consiste básicamente en encontrar terminaciones y bifurcaciones en la huella, para hacer esto la mayoría de algoritmos se basan en usar imágenes binarias, donde aplican procesos de adelgazamiento de las crestas, comparación con plantillas y finalmente se eliminación de falsas detecciones.

Sin embargo, Maltoni [2] fue uno de los primeros en proponer la extracción de minucias directamente en escala de grises, en el cual se evita la binarización y el adelgazamiento de las crestas, el motivo de hacer esto es porque en los procesos de binarización se puede perder información, y los de adelgazamiento que tienen un tiempo de consumo alto (dependiendo del grosor de las crestas) causan una adición de falsas minucias.

En [12] y [25] se realiza la extracción de minucias directamente en la imagen binaria, mediante el método de codificación del contorno de la huella, donde las minucias se detectan cuando hay una curva significativa.

Una vez detectados estos puntos característicos, se debe eliminar las falsas minucias, que se dan principalmente por el ruido de la huella y/o la mala reconstrucción de la huella, estas estructuras de falsas minucias fueron clasificados en [2], y se pueden ver en la Figura 20.

2.3.3.2. Algoritmo de Extracción de Minucias

El enfoque de este trabajo para extrae las minucias es usando la imagen adelgazada, por lo cual se necesitan los procesos siguientes:

1. Binarización.
2. Adelgazamiento.
3. Detección de Minucias.
4. Eliminación de Falsas Minucias.

Siguiendo con el diagrama de flujo de la Figura 11, se adiciona la etapa de la extracción de de minucias como se ve en la Figura 15.

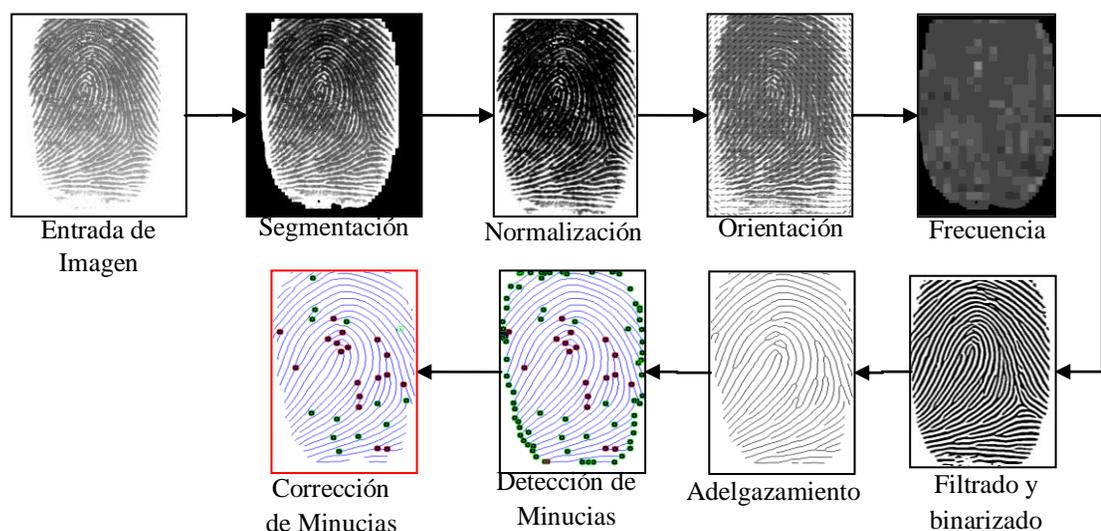


Figura 15. Diagrama de flujo para la extracción de minucias.

2.3.3.3. Binarización

El proceso de binarización consiste en convertir una imagen en escala de grises en dos bits (colores blanco y negro) 1 y 0, se hace estableciendo un umbral fijo para las imágenes, si un valor en la imagen gris es mayor al umbral será 1, si es menor será 0, un ejemplo de imágenes binarias se puede ver en la Figura 45, esto se necesita para la etapa de adelgazamiento.

2.3.3.4. Adelgazamiento

Adelgazar la imagen lleva a un proceso simple de extracción de minucias, para ello se realiza detectando los bordes de la imagen, y eliminándolos sucesivamente, teniendo en cuenta que algunas consideraciones para no eliminar los pixeles internos. La reducción está basado en el algoritmo de ROS [31], en este algoritmo se detectan de forma sucesiva los pixeles del borde superior, inferior, derecho e izquierdo, teniendo en cuenta las consideraciones de la Figura 17, y la información de los 8 pixeles vecinos, para formar líneas delgadas (*del grosor de 1 pixel*).

Para la detección de los bordes, primero debemos determinar el tipo de conectividad entre pixeles, para ello se utiliza la Figura 16, donde se muestran los pixeles vecinos de P0 de un bloque de 3x3, si se analizan los pixeles p1, p2, p3, p4, p5, p6, p7 y p8 se dice que es de conectividad 8, por el contrario si se examinan los pixeles p1, p3, p5 y p7 ó p2, p4, p6 y p8 se dice que es de conectividad 4.

P8 (x-1, y-1)	P1 (x-1, y)	P2 (x-1, y+1)
P7 (x, y-1)	P0 (x, y)	P3 (x, y+1)
P6 (x+1, y-1)	P5 (x+1, y)	P4 (x+1, y+1)

Figura 16. Vecindades del pixel p0.

El segundo paso es tener en cuenta la distribución de los pixeles en el bloque, si se cumple alguna de las condiciones de la Figura 17, se establecerá que el pixel central p0 es parte del borde.

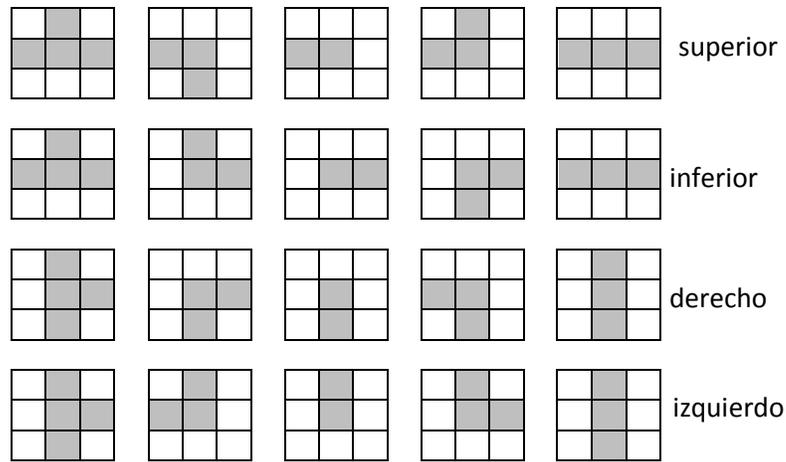


Figura 17. Consideraciones para detectar el borde de una imagen.

En síntesis, el algoritmo trabaja de la siguiente forma:

1. La suma de los pixeles vecinos sea igual a mayor a uno.
2. Para localizar los pixeles del borde superior, inferior, derecho e izquierdo se deben tener en cuenta las condiciones de la Figura 17.
3. Realizar un ciclo completo para cada borde y borrar el pixel detectado hasta que no se detecte ningún otro pixel.

2.3.3.5. Detección de Minucias

Después del adelgazamiento las minucias pueden ser detectadas, utilizando mascarar de comparación, en este caso se detectan las terminaciones y bifurcaciones. La primera se presenta cuando una línea pierde continuidad y, la segunda cuando dos o más líneas se unen en punto, con ellas se pueden representar todas las minucias que buscan los dactiloscopista en los cotejos, tales como ojales, puntos aislados, horquillas, etc.

Utilizando una ventana de comparación de la Figura 18, se recorre toda la imagen pixel a pixel, si la suma de los pixeles vecinos es igual a uno, P0 indicará la posición de la terminación, otras condiciones para que sea una terminación se resumen en la Figura 18.

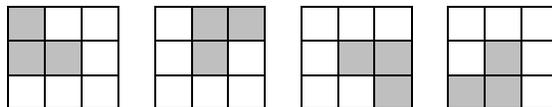


Figura 18. Condiciones para detectar terminaciones.

Para el caso de las bifurcaciones las consideraciones se resumen en la Figura 19, al igual que con las terminaciones, el recorrido es pixel a pixel, cuando un pixel cumpla con alguna de estas consideraciones, se dirá que el pixel p_0 es una bifurcación.

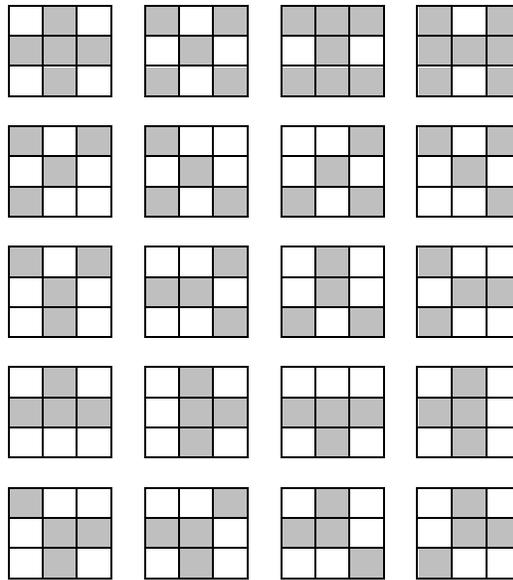


Figura 19. Condiciones para detectar bifurcaciones.

2.3.3.6. Eliminación de Falsas Minucias

Cuando se hallan las minucias, es muy común encontrar falsas minucias, debido a sobre-detecciones, se pueden hallar distintas formas que se atribuyen a falsas minucias, estas se resumen en la Figura 20:

- A. Cuando una línea pierde continuidad cerca a ella, dos terminaciones serán detectadas, con ángulos desfasados ~ 180 grados.
- B. Cuando dos líneas pierden continuidad cerca a ella, cuatro terminaciones serán detectadas, con ángulos desfasados ~ 180 grados.
- C. Cuando una línea que debería llevar un flujo “natural” de dos líneas paralelas, se corta y se une repentinamente con una línea cercana a ella, y el ángulo entre la línea y la bifurcación están desfasado en ~ 90 grados.
- D. Cuando hay dos bifurcaciones muy cercanas, puede ser un caso especial de doble bifurcación, pero si las líneas que forman las bifurcaciones están bastante separadas se trata de dos líneas paralelas. Sus ángulos son desfasados ~ 180 grados.
- E. Cuando una bifurcación unida por una línea a una terminación están muy cerca, se deben eliminar las dos minucias y dejar una sola línea.
- F. Cuando dos líneas paralelas son atravesadas por una línea, y se forman dos bifurcaciones con ángulos desfasados paralelos, se debe eliminar la línea que las une.

- G. Cuando en una bifurcación las dos líneas que se unen antes son atravesadas por otra línea, forma dos bifurcaciones con algunos paralelos, se debe eliminar esta línea, y dejar una solo bifurcación.
- H. Similar al caso F pero con dos líneas.

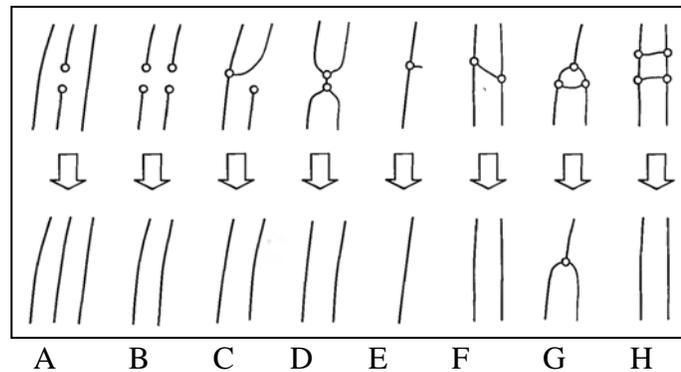


Figura 20. Estructuras comunes de falsas minucias.
Fuente: Imagen tomada de Handbook of Fingerprint Recognition [2].

2.3.4. COMPARACIÓN DE HUELLAS

2.3.4.1. Trabajos Relacionados

Se han elaborado muchos algoritmos para la comparación de huellas, utilizando técnicas que involucran hasta redes neuronales, el proceso consiste básicamente en comparar las características extraídas en cada huella y establecer su similitud.

Estas técnicas según [2] pueden clasificarse en 3 grandes grupos el cotejo de huellas, los que utilizan la correlación, las crestas y las minucias. Aunque ahora se está empezando a usar sistemas híbridos que combinan estas particulares.

La primer técnica usa la superposición de las imágenes y la correlación entre cada pixel es computado, para hacer esto se puede usar la fase-única de las componentes en la transformada de Fourier de la imagen [30], los resultados son muy buenos en imágenes de baja calidad, sin embargo el tiempo de procesamiento es muy alto comparado con otras técnicas, debido a las etapas de alineamiento¹ y el mismo proceso de correlación; para mejorar esto se utiliza las transformadas de Fourier-Mellin, que logran alinear la huella rápidamente, y el uso de la correlación de fase-única con banda limitada [32]. También se ha propuesto el desarrollo de prototipos opto-electrónicos, donde se proyectan la imagen en

¹ La alineación involucra la traslación, rotación y el escalado.

una pantalla de cristal liquido, y mediante arreglo de lentes y luz laser se procesa la huella en tiempos realmente cortos [33]. En general este tipo de procesos utilizan toda la información de la huella, y la tasa de reconocimiento de huellas es alta, las limitaciones que tiene son el tiempo de procesamiento, que se pueden mejorar con dispositivos opto-electrónicos, pero su costo es muy alto.

La segunda técnica utiliza el patrón de las crestas para compararlas, en ellas se puede encontrar características como la orientación y frecuencia local, forma de las crestas e información de la textura; la extracción es mucho más confiable que las minucias, pero la distinguibilidad es muy baja. Se suele utilizar la respuesta del filtro de Gabor en distintas direcciones para crear un *FingerCode*, que después es comparado mediante la distancia euclidiana entre los códigos [7], la ventaja de esta técnica es que no es necesario reconstruir toda la huella, no hay procesos adelgazamiento, detección de minucias, simplemente la estimación local, la detección del núcleo y la rotación. También se encuentra el uso de los 7 momentos Hu para comparar la forma de la huella, esta comparación tiene la ventaja que es invariante a la posición, escala y rotación [14], en ella se encuentran trabajos que utilizan los momentos Hu en la imagen mejorada con el filtro de Gabor, y redes neuronales para la comparación de características [34], en general estos algoritmos no requieren muchos procesos para extraer las características, por ello son bastante rápidos, además de ser confiables. Una nueva técnica busca el alineamiento de las crestas de la huella, y compara cuántas de ellas son iguales, usando la transformada de Hough [35] a las crestas, sin embargo no presenta muy buenos resultados y el tiempo de comparación es de alrededor de 12 segundos.

El último método es el más popular de todos, se fundamenta en como los dactiloscopistas comparan las huellas, es decir buscando minucias y comparan su posición relativa. A nivel de algoritmos, se han propuestos varias alternativas para la comparación de minucias, el primer paso en todos es la extracción de minucias en la huella como se mencionó en el apartado 2.3.3.2. Con el triplete de la posición y ángulo de las minucias, se puede determinar la similitud entre dos huellas usando el área superpuesta formado por las minucias [36], con el triplete se usa la transformada de Hough para hallar todas las posibles combinaciones de posición, escala y rotación, aunque la complejidad del cálculo puede tomar un tiempo considerable [37], un nuevo método compara las minucias similar al método humano, busca una minucia en particular y sus vecinos próximos, se hace uso de grafos para determinar los vecinos de las minucias, y un algoritmo de comparación dinámica[15]. Este último método se usa en este trabajo.

2.3.4.2. Algoritmo de Comparación de Huellas

Se trabajó la comparación de huellas mediante las coordenadas de las minucias, se describen dos técnicas para la comparación de huellas. Los procesos que se van a estudiar son:

1. Comparación mediante Minucias usando un alineamiento previo
2. Comparación mediante Minucias usando Grafos

El diagrama de flujo para la comparación se describió en la Figura 1.

2.3.4.3. Comparación de Minucias con Alineamiento Previo

La comparación se hace buscando las similitudes entre las minucias, para entender esto llamemos S a la huella de entrada (Sospechosa), y C a la huella almacenada en el registro (Conocida). En cada huella se extraen las características de la posición $[x,y]$ y el ángulo $[\theta]$ de cada minucia, es decir bifurcación ó terminación; entonces para S y C tenemos:

$$S = \{m_1, m_2, m_3, \dots, m_m\}, \text{ donde } m_i = \{x_i, y_i, \theta_i, tipo_i\} \quad i = 1..m \quad (16)$$

$$C = \{m'_1, m'_2, m'_3, \dots, m'_n\}, \text{ donde } m'_j = \{x'_j, y'_j, \theta'_j, tipo'_j\} \quad j = 1..n \quad (17)$$

Donde m y n denota el número de minucias de en S y C , respectivamente.

Entonces una minucia en S será considerada como igual a una minucia en C si la distancia espacial (Δp) es menor que un radio de tolerancia r_0 , y la diferencia de dirección ($\Delta \theta$) es menor que un ángulo de tolerancia θ_0 como se ve en las siguientes ecuaciones:

$$\Delta p(m_i, m_j) = \sqrt{(x_i - x'_j)^2 + (y_i - y'_j)^2} \leq r_0 \quad (18)$$

$$\Delta \theta(m_i, m'_j) = \min(|\theta_i - \theta'_j|, 360^\circ - |\theta_i - \theta'_j|) \leq \theta_0 \quad (19)$$

La ecuación (19) toma el mínimo entre $|\theta_i - \theta'_j|$ y $360^\circ - |\theta_i - \theta'_j|$ por la circularidad de los ángulos, ya que la diferencia entre 350° y 5° es 15° . Se usa el ángulo y radio de tolerancia, debido a que es muy difícil encontrar las mismas coordenadas en dos huellas, debido a los errores en la extracción de minucias y la toma de impresión dactilar, a esto se le llama *algoritmo de comparación elástico*.

Para hacer esta comparación efectiva, es necesaria la alienación relativa de las huellas, es decir superponer a S sobre C , o viceversa. Las operaciones básicas para ello son la translación $[x,y]$, rotación $[\theta]$ y escala $[t]$; y algunos algoritmos más robustos hacen correcciones geométricas de deformaciones [18], con transformaciones afines. La escala puede ser determinada por la resolución del sensor, ó por el ancho de las crestas de S y C .

Para la alineación llamemos S'' a las minucias alineadas de S con respecto a C , tal como se muestra en la ecuación (20), donde es necesaria la matriz de transformación de (21).

$$S''_{\Delta x, \Delta y, \theta}(m'_j = \{x'_j, y'_j, \theta'_j\}) = m''_j = \{x''_j, y''_j, \theta'_j + \theta\} \quad (20)$$

$$\begin{bmatrix} x''_j \\ y''_j \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x'_j \\ y'_j \end{bmatrix} + t \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (21)$$

Donde Δx y Δy se atribuye al desplazamiento (*translación*) de S a C .

Llamemos a Mc una función que indica que indica si una minucia en m''_j es igual a m_i , acorde con las ecuaciones (18) y (19), tal que:

$$Mc(m''_j, m_i) = \begin{cases} 1 & \text{si } \Delta s(m''_j, m_i) \leq r_0 \text{ y } \Delta \theta(m''_j, m_i) \leq \theta_0 \\ 0 & \text{en otros casos} \end{cases} \quad (22)$$

Entonces el problema de cotejo de huellas puede ser formulado como:

$$Maximización_{\Delta x, \Delta y, \theta, P} = \sum_{i=1}^m Mc(S''_{\Delta x, \Delta y, \theta}(m'_{P(i)}), m_i) \quad (23)$$

Donde $P(i)$ es la función que determina el emparejamiento entre las minucias de S y C , es decir es la función que indica cuales son el par de minucias iguales en cada huella, y que servirán como referencia para la alineación, estos puntos son introducidos en la ecuación (21) para transformar todos los puntos de una huella.

La solución de (23) es sencilla cuando se tiene correctamente alineadas las huellas, visualmente esto se puede ver como en la Figura 21, donde las minucias de S'' alineadas con C , se encierran en una “caja” con el radio de tolerancia r_0 , entonces si una minucia de C esta dentro de esta “caja” se analiza el ángulo, si es similar las dos minucias son iguales; en la figura se han señalado con línea continua los círculos donde las minucias son iguales, por ejemplo m_2 y m_1'' , y con línea punteada las que no lo son como m_4 y m_3'' ó m_1 .

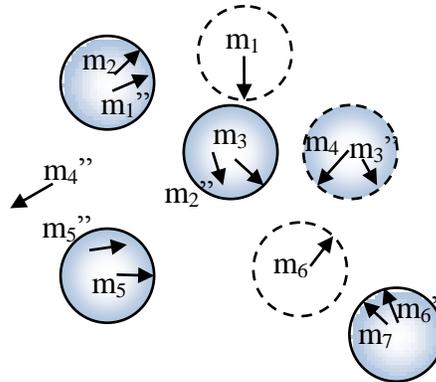


Figura 21. Problema de comparación de minucias alineadas.

Sin embargo puede ocurrir que dos “cajas” de S'' compartan una minucia ó más en común de C , como en la Figura 22 , por ejemplo si m_1 es emparejado como m_2'' , m_2 no tendría ninguna pareja, pero si m_1 es emparejado con m_1'' , entonces m_2 tendrá a m_2'' , esto maximizaría el número de minucias como aparece en la ecuación (23).

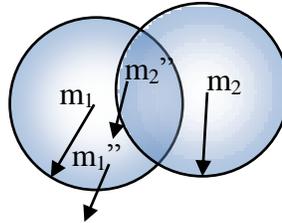


Figura 22. Caso en el que dos minucias comparten una minucia en común.

Para hallar el alineamiento previo y la solución de la ecuación (21), se deben establecer θ , t , Δx y Δy , para ello es necesario encontrar dos minucias iguales en cada huella, las cuales llamaremos P_1 y P_2 a las minucias de la huella C , y P_3 y P_4 a las minucias de la huella S . Las diferentes etapas del ajuste se observan en la Figura 23.

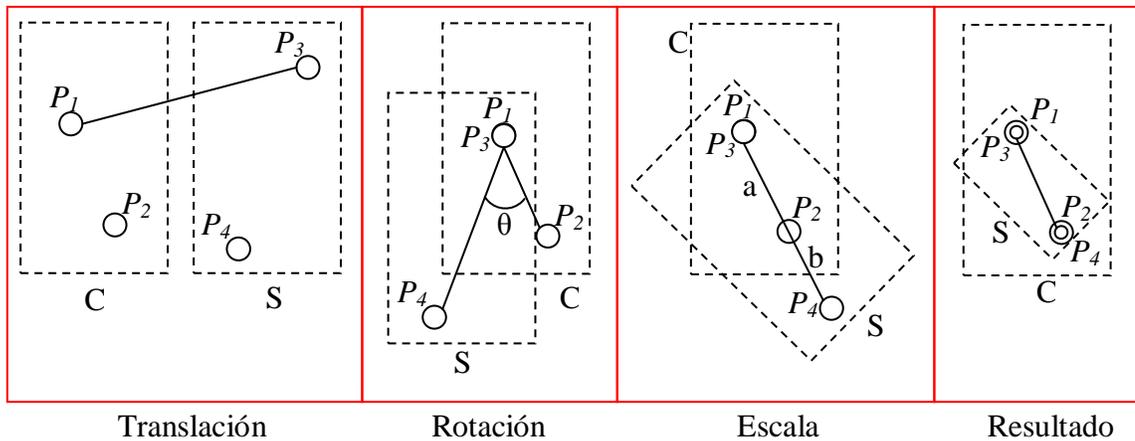


Figura 23. Procesos de ajuste de la huella S respecto a C .

Las etapas del proceso de alineación son:

1. Traducción: se hace coincidir los puntos P_1 y P_3 :
2. Rotación: se forma un ángulo θ entre los puntos $[P_1, P_2, P_4]$, entonces se hace rotar S con centro en P_1 en la misma dirección de P_2 .
3. Escala: se miden las longitudes de ambas imágenes, para la imagen C la longitud a es la distancia entre P_1 y P_2 , para la imagen S la longitud b es la distancia entre P_3 y P_4 , y se cambia el tamaño a la mayor con respecto a la menor, o viceversa.
4. El resultado es la imagen S ajustada respecto a C . Con estos valores se puede hacer la transformación de los demás puntos de la huella y se procede a calcular las minucias iguales en cada huella.

2.3.4.4. Comparación de Minucias por Grafos

Debido a que la alineación por la alineamiento previo se hace teniendo en cuenta que los cuatro puntos se elijen manualmente, se hace el uso de grafos para alinear las huellas automáticamente y comparar las minucias, este método fue propuesto por Chikkerur [15], y el algoritmo trabaja de la siguiente forma.

Se define una representación para capturar la información estructural local de las minucias llamada *K-plet*. Esta representación es invariante a la translación y rotación, siempre y cuando se defina su propio sistema local de coordenadas. Los *K-plet* consiste de una minucia centra m y K otras minucias $m(1), m(2), \dots, m(k)$ escogidas de sus vecinos locales, como se ve en la Figura 24.

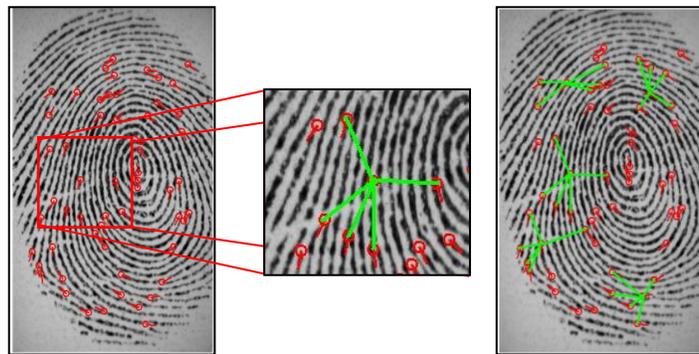


Figura 24. Representación de *K-plet*,

Fuente: imagen tomada del trabajo de tesis de Sharat Chikkerur [15].

Cada minucia es definida en términos de sus coordenadas $[\varphi(i, j), \theta(i, j), r(a, b)]$ donde $r(a, b)$ representa la distancia euclidiana entre la minucia $m(a)$ y $m(b)$. $\theta(i, j)$ es la orientación relativa de la minucia $m(j)$ respecto a la minucia central $m(i)$. $\varphi(i, j)$ representa el ángulo entre la arista que conectando las dos minucias y el ángulo de orientación de la minucia, este será el sistema de coordenadas, y todos los otros ángulos serán referenciados a esta minucia central; como se ve en la Figura 25.

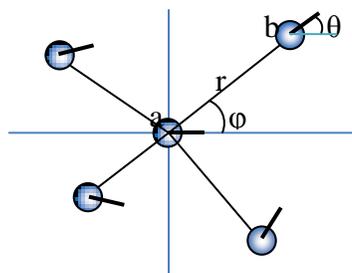


Figura 25. Representación de los ángulos en un *K-plet*.

Para escoger los vecinos de la minucia central, se escoge los vecinos más cercanos para los cuatro cuadrantes, se establece el r más grande y si en el círculo formado por este se encuentra una minucia no asociada, se asociará a la minucia central; por ello no necesariamente existirán 4 minucias asociadas a la minucia central.

Después esto se codifica la relación estructural de los K -plet en forma de grafos. Un grafo es un conjunto, no vacío, de objetos llamados vértices (*o nodos*) y una selección de pares de vértices, llamados aristas que pueden ser orientados o no. Típicamente, un grafo se representa mediante una serie de puntos (*los vértices*) conectados por líneas (*las aristas*). Entonces cada minucia es representada por un vértice (v) y las minucias asociadas (*vecinas*) son representadas por una arista direccionada (u, v). Cada vértice (u) es lleno con atributos de $[x(u), y(u), \theta(u), t(u)]$, que representan la posición, ángulo y tipo de minucia, mientras que cada arista tienen atributos de $[r(u, v), \phi(u, v), \theta(u, v)]$.

Para hacer la comparación de las huellas, se utiliza un enfoque de *programación dinámica*, basada en el algoritmo de alineación de cadenas (*string*), básicamente si se tienen una cadena $S=[acbcd]$ y $T=[cabd]$ entonces las dos cadenas pueden ser alineadas como S' y T' , de esta forma:

$$S' = ac_bcd$$

$$T' = _cabd_$$

La solución de este tipo requeriría un tiempo exponencial, por ello es necesario la *programación dinámica* para solucionar éste problema, el cual usa relaciones de recurrencia. Después se relaciona los elementos iguales en posición de ambas cadenas, estas corresponden a las minucias iguales en cada huella, con ello el último paso es determinar un valor de la comparación de las huellas, por ello se elige un puntaje ó *score* generado a partir de la cantidad de minucias de cada huella y las minucias comparadas, mediante la ecuación (24).

$$s = m^2 (M_S M_C)^{-1} \quad (24)$$

Donde m es la cantidad de minucias iguales en cada huella, M_S y M_C son la cantidad de minucias de la huella sospechosa y conocida, respectivamente.

Capítulo 3

3. RESULTADOS EXPERIMENTALES

Para entender mejor los resultados en general de las diferentes etapas, se usaron las diferentes imágenes de la base de datos de la FVC2004 [10], y algunas imágenes propias del autor. Además se incluye unas recomendaciones del trabajo en cada una de las secciones, y al final el pseudocódigo del algoritmo que aparece en el anexo C. El código fuente del trabajo está en el CD que se entrega a la facultad.

3.1. CAPTURA DE IMAGEN

Las imágenes fueron capturadas de diferentes formas, la primera de ellas se realizó con sensores electrónicos, aunque no fue hecha por el autor, ya que se descargó de la base de datos de la FVC2004, disponible en internet. La segunda forma se realizó por el método off-line, con ella se creó la base de datos del autor, las cuales se describen a continuación.

3.1.1. Base de datos de la FVC2004

Las imágenes de la FVC2004 están divididas como aparece en la Tabla 3, son huellas bastante difíciles de trabajar, un ejemplo de cada una de estas huellas se ve en la Figura 26, en ella de izquierda a derecha, las imágenes pertenecen a la base de datos DB1_B, DB2_B, DB3_B y DB4_B, respectivamente, los tamaños han sido redimensionados proporcionalmente para observar mejor el tamaño de salida del sensor.



Figura 26. Tipos de imágenes de la FVC2004.

También se tiene en cuenta el tiempo de proceso de la huella, tomando dos tiempos, uno para la extracción de características (*enrollment*), el cual lleva todos los procesos de la Figura 15, y el otro de comparación (*matching*), el cual lleva los procesos de comparación mediante minucias; se toma como referencia tiempo de ejecución de los algoritmos de la FVC2004, el cual se resume en la Tabla 4, ella está dividida en categoría “open”, que es una categoría de poca exigencia, y “light”, que es caracterizado por los bajos recursos computacionales y memoria usada. El tiempo de extracción es para una sola huella y el de comparación es de una huella con toda la base de datos.

Tabla 3. Tipos de sensores de la FVC2004

Nombre	Sensor	Tamaño	Resolución
DB1_B	Óptico (<i>CrossMatch</i>)	640 × 480	500 dpi
DB2_B	Óptico (<i>Digital Persona</i>)	328 × 364	500 dpi
DB3_B	Barrido Térmico (<i>Atmel FingerChip</i>)	300 × 480	512 dpi
DB4_B	Sintéticos (<i>SFinGe v3.0</i>)	288 × 384	500 dpi

Tabla 4. Tiempos de comparación y cotejo de la FVC2004.

Nombre	Empresa	Tipo y país	Tiempo de extracción (s)	Tiempo de comparación (s)
P101 <i>Open</i>	Bioscrypt Inc.	Industria Canadá	0.08	1.48
P047 <i>Open</i>	Sonda Ltd.	Industria Rusia	2.07	2.07
P071 <i>Open</i>	Institute of Automation, The Chinese Academy of Sciences	Académico China	0.35	0.67
P049 <i>Open</i>	Neurotechnologija Ltd.	Industria Lituania	0.34	0.38
P107 <i>Light</i>	Ji Hui	Independiente China	0.13	0.13
P009 <i>Light</i>	Suprema Inc.	Industria Corea	0.25	0.22
P101 <i>Light</i>	Bioscrypt Inc.	Industria Canadá	0.09	0.17
P108 <i>Light</i>	Beijing HanWang Technology Co., Ltd.	Industria China	0.23	0.23

3.1.2. Base de datos del autor

La base de datos creada por el autor consta de 160 imágenes adquiridas de forma *off-line*, 80 de las cuales se tomaron de los dedos pulgar, índice, anular, medio y meñique, y las otras 80 del índice y el pulgar, las primeras 80 huellas se tomaron utilizando tinta para impresiones dactilares y papel común, todas las impresiones son simultaneas, es decir tocar el papel sin hacer rotación del dedo (ver Figura 8), las impresiones no fueron tomadas por un experto, pero se tuvo en cuenta que se presionaran bien en el papel de forma adecuada, además se utilizó poca tinta por lo cual las imágenes en general se ven opacas; las otra 80 huellas se tomaron de manera inadecuada y sin utilizar tinta de huellas dactilares, en lugar de ello se utilizó oleo, además se busco personas que tuvieran cierto desgaste.

Todo ello con el fin de limitar al sistema de reconocimiento para que haga la comparación, porque si se obtiene buenos resultados con imágenes de media y mala calidad es de esperarse que con buena calidad los resultados sean mucho mejores, algunos ejemplos de estas huellas se pueden ver en la Figura 27, en ella de izquierda a derecha, la huella de un dedo pulgar y meñique de mediana calidad, y huellas del índice de mala calidad, por el desgaste y por la mala adquisición, los tamaños han sido redimensionados proporcionalmente.

Para la digitalización de las huellas fueron se utilizó un escáner óptico a 400 dpi, y los tamaños del área de la huella varían según sea el tamaño de la impresión, aunque la sugerencia del FBI es tomar la imagen con 500 dpi, se optó por una resolución menor para evaluar una ventaja del algoritmo, la cual gracias a la estimación de la frecuencia local se pueden tener imágenes con crestas un poco unidas, y reconstruirlas adecuadamente.



Figura 27. Huellas de la base de datos del autor.

También se tomaron imágenes con una resolución mucho mayor a 500 dpi, para observar como cambiaba el ancho de las crestas, además con estas imágenes se probó en el algoritmo reconstruía negativamente las huellas, ya que el ancho de las crestas confundía la

frecuencia real de la imagen, y para una misma cresta podían aparecer dos, de igual forma se digitalizaron a 300 dpi, en este caso la huella logró ser reconstruida adecuadamente, aunque una buena cantidad de crestas se unían, debido a la poca distancia en pixeles que tenían, los resultados se puede observar en la Figura 28, donde se toman la misma sección de una imagen para distintas resoluciones, A es la imagen original, B a 300 dpi, C a 400 dpi, D a 500 y E a 700 dpi, como se puede observar la imagen E tienen una mala reconstrucción, y claramente se ve que una cresta se duplica, D a tiene una buena reconstrucción, no obstante se siguen observando huecos en medio de las crestas, en C no aparecen ninguna cresta con huecos o duplicados, finalmente B tiene algunas crestas pegadas, pero su reconstrucción es buena, por ello se eligió escanear las huellas a 400 dpi, sin embargo en el C.T.I. existen imágenes con gran resolución, entonces la solución a esto fue la re-dimensión digital de la imagen, la discusión de esto se encuentra en la sección 3.5.1.

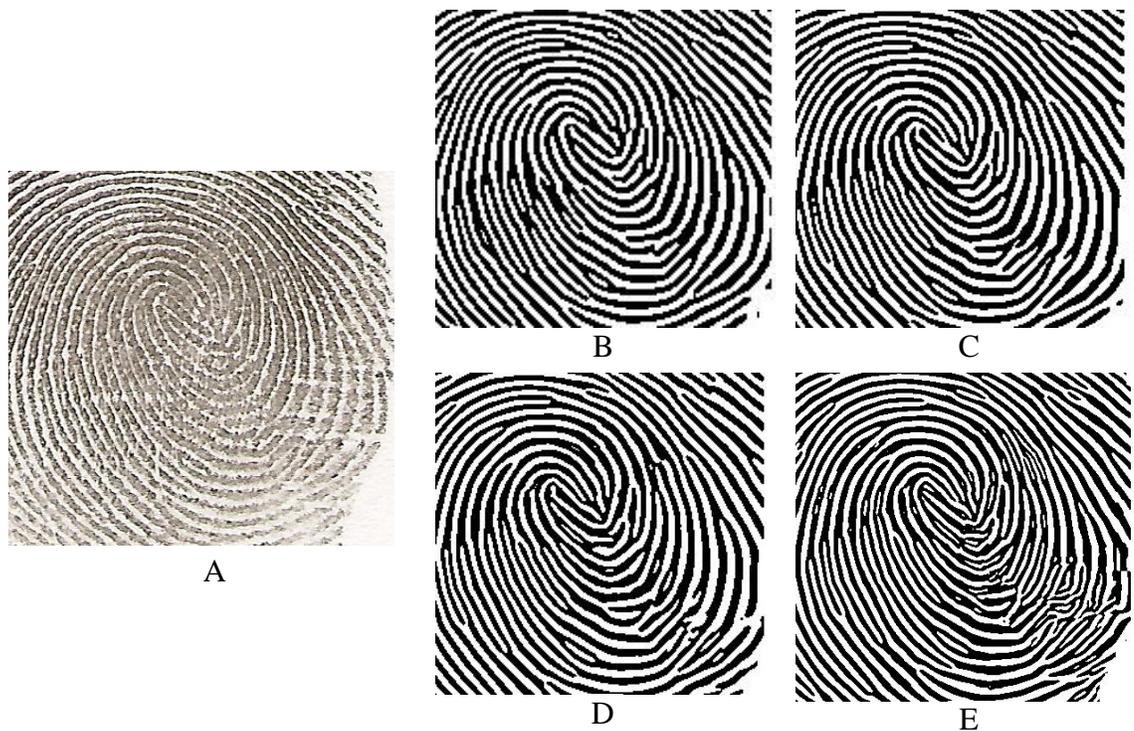


Figura 28. Reconstrucción de huellas con resolución inadecuada.

3.2. ALGORITMO

El algoritmo de comparación de huellas comprende dos etapas una de inscripción, es decir donde se extraen las características de la imagen, y otra de comparación, donde se ve la similitud entre las huellas, en el marco teórico se observa cómo se realiza estas dos, la primera se observó en la Figura 15, y la segunda en la Figura 1. Sin embargo el desarrollo completo para la comparación de un par de huellas se puede ver en la Figura 29. Para cada imagen se realiza el mismo procedimiento, al final se alinean y se comparan sus características.

Considerando que las imágenes ya se han adquirido, se realizan las siguientes etapas para cada huella:

- Adecuar los niveles de gris de la imagen con la normalización o la ecualización local.
- Segmentar la huella y localizar el rectángulo mínimo que encierre la máscara.
- Se aplica un ROI o recorte a la imagen normalizada, desde este punto la imagen normalizada cambia su tamaño, y las próximas imágenes se basarán en las dimensiones de esta imagen recortada, además se multiplica con la máscara de segmentación.
- Se calcula la orientación local de la imagen normalizada.
- Se estima la frecuencia local tomando la información de la orientación y la imagen normalizada.
- Se realiza el filtrado con el filtro de Gabor, en forma de convolución espacial, para ello se requiere la frecuencia para estimar el tamaño de los kernel de convolución, cada uno de ellos es rotado en 60 partes entre 0° y 180° , se necesita la orientación para aplicar la orientación en sentido paralelo a las crestas, finalmente se evita realizar la convolución en zonas fuera de la máscara de segmentación.
- Se adelgaza la imagen y se localizan las minucias, es decir terminaciones y bifurcaciones.
- Se eliminan falsas minucias, que surgen en el borde de las crestas junto con algunas minucias internas, se tiene en cuenta la matriz de confiabilidad para eliminar minucias de partes muy ruidosas en la imagen.
- Se hace la alineación de las minucias, donde se realizan transformaciones de rotación, traslación y escala.
- Se compara con un algoritmo elástico y se determina la cantidad de minucias iguales.

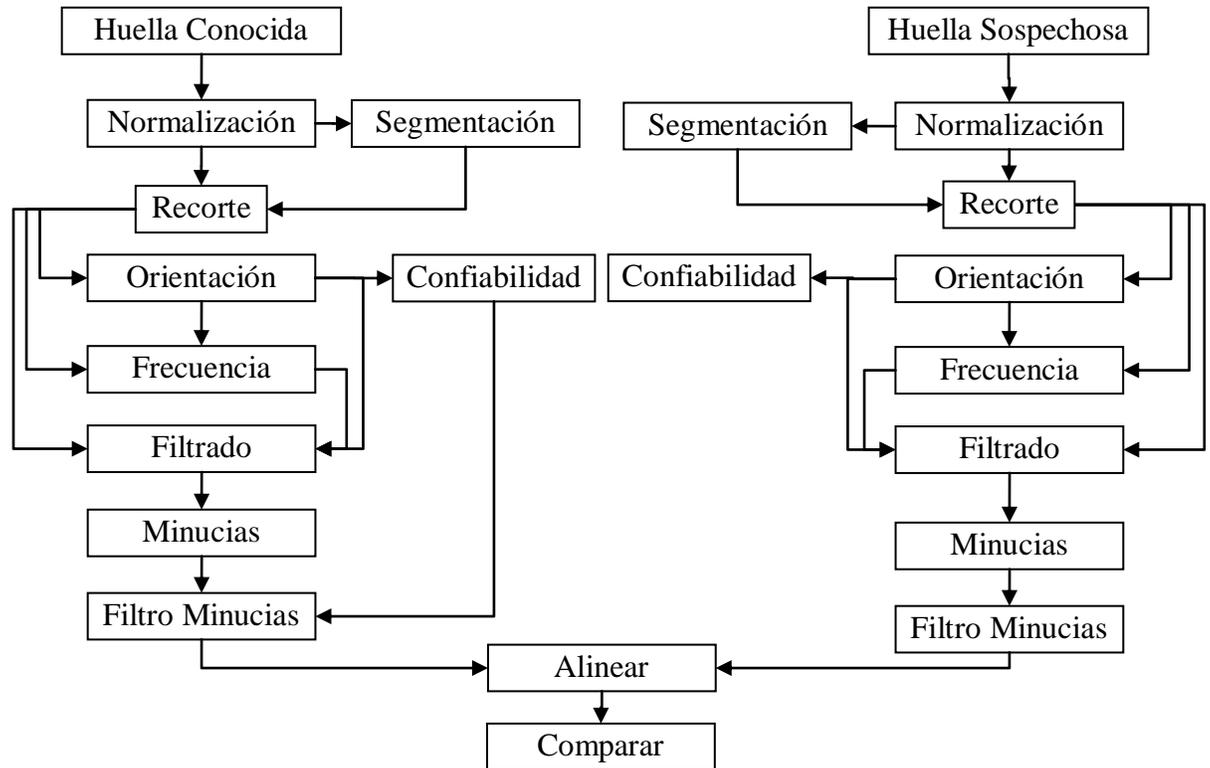


Figura 29. Rutina para comparar dos huellas.

3.2.1. Medidas del tiempo de ejecución

En todo el trabajo se prestará atención a los tiempos de cada proceso, y los procesos finales de extracción de características de la huella (*enroll*) y comparación (*match*), para ello todo los algoritmos son ejecutado bajo Windows XP Professional Service Pack 2 en Intel Core 2 Duo T5750 a 2.00 Ghz y 3 Gb de RAM, implementado en DevC++ y con las librerías de OpenCV. La forma de medir el tiempo se hace utilizando el algoritmo del anexo B.

Como la base de datos de la FVC2004 está disponible en internet, se dejará en el documento el nombre de la imagen original, con el fin que en un futuro se pueda comparar.

3.3. MEJORAMIENTO

3.3.1. Segmentación

Los valores utilizados en el trabajo corresponden a $N = 6$, y el umbral fue 15. Los primeros resultados dejaban algunas partes (negras) de la máscara dentro de la huella, y otras (blancas) fuera de la huella, es importante eliminar estas zonas porque cuando se va a filtrar la imagen aparecerán falsas minucias en estas, más aun estas zonas pueden pertenecer a minucias, se intento solucionar este problema aumentando N y variando el umbral, donde se eliminaban gran parte de estas mascararas, pero continuaban apareciendo, como se ve en la Figura 30, en ella se utilizan valores de $N = 6, 8, 12$ y 16 pixeles, a la izquierda la imagen original, a la derecha la imagen segmentada.

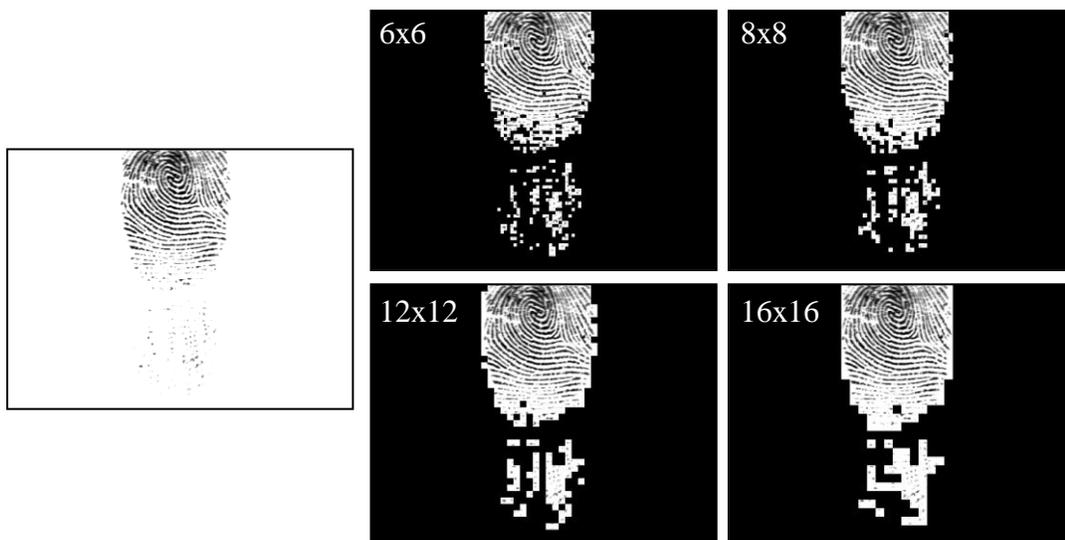


Figura 30. Ejemplos de segmentación de huellas,.

Otra solución a este problema fue utilizar la información de sus vecinos², usando un bloque de análisis como en la Figura 16. Asumiendo que 255 es blanco y 0 es negro, entonces para P_0 se decide lo siguiente:

1. Si la suma de pixeles P_1 a P_8 es mayor o igual a 4×255 , P_0 será igual a 255.
2. Si la suma de pixeles P_1 a P_8 es menor a 4×255 , P_0 será igual a 0.

Con estas consideraciones la máscara no presenta las mascararas indeseadas, algunos resultados se pueden ver en la Figura 31, donde A_1, B_1 y C_1 son las imágenes originales, A_2, B_2 y C_2 son las imágenes segmentadas sin la corrección, donde la imagen B_2 tiene muchas partes de la segmentación dentro de la huella, A_3, B_3 y C_3 son las imágenes

² Los vecinos son los 8 pixeles que están alrededor del pixel de análisis.

después de la corrección, en ellas se ve como se soluciona el problema, aunque se probó en muchas imágenes y se encontraron errores como los de la huella C3, donde parte de la máscara indeseada aparecen en la parte inferior, esto se podía eliminar aumentando el umbral, pero esto afectaba las máscaras en otras huellas, por lo que se dejó el umbral de 15. Por último están A4, B4 y C4 que es una imagen superpuesta con la original, esto se hace para observar que zonas están afectas por la segmentación. El tipo de imagen como C1 es común cuando se posa el dedo mal en el sensor, pero una toma de otra huella o recortando manualmente puede ser la solución en estos casos.

Además en este proceso se encierra en un rectángulo a la máscara, con el fin de recortar la imagen (ROI) y tener menos píxeles que procesar en procesos futuros, esto se ve claramente en la huella B y C, el propósito de esto es disminuir la cantidad de memoria reservada para cada imagen, de los procesos siguientes.

El tiempo promedio para las imágenes de la Figura 31 son: A=4.45, B=6.06, C=4.674 y D=11.52 ms, respectivamente. Las imágenes utilizadas son A=7_2_{DB4_A}, B=45_1_{DB3_A}, C=16_2_{DB2_A} y D=1_1_{DB1_A}.

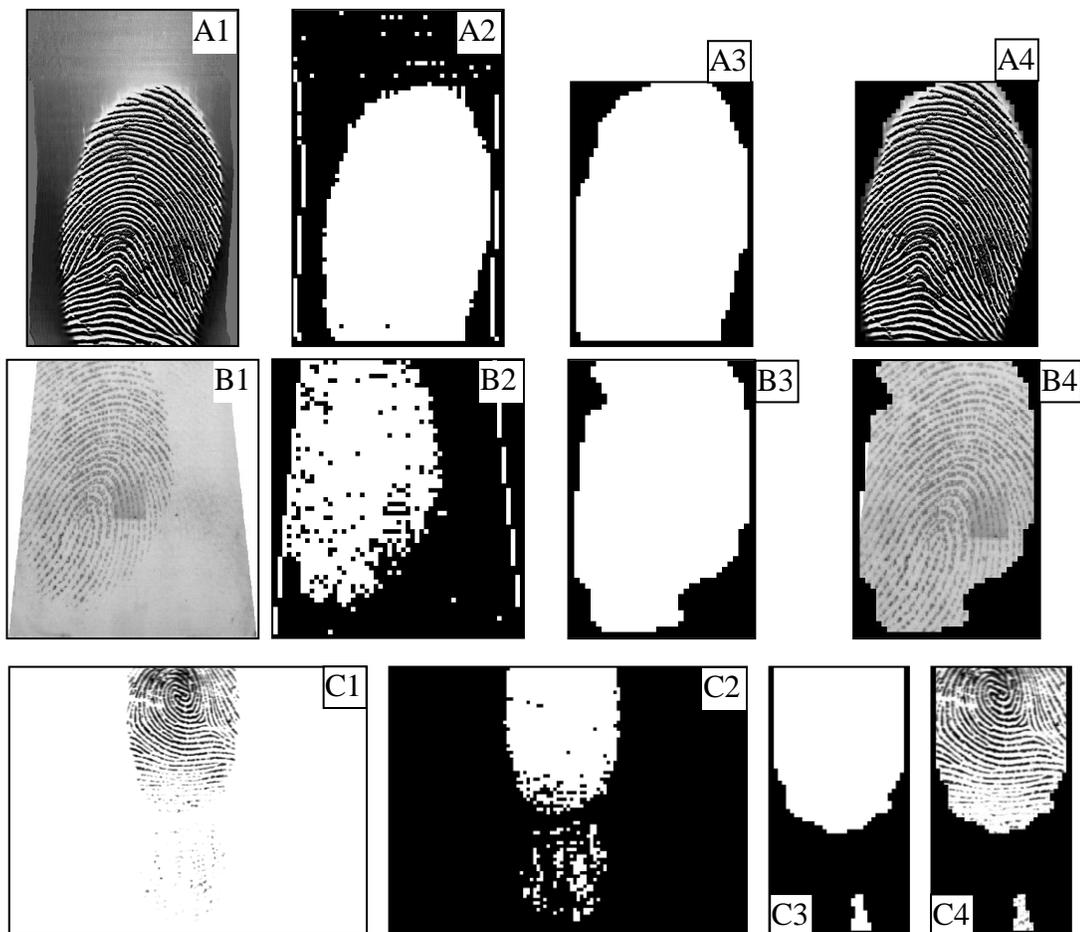


Figura 31. Imágenes segmentadas antes y después de la corrección.

Aplicar la segmentación es relativamente sencillo, se deben tener presente las siguientes recomendaciones:

- Antes de aplicar la segmentación se debe mejorar el histograma, con el fin de que la varianza entre las crestas y el fondo aumente, esto mejora mucho la detección, permitiendo mantener un umbral estático.
- La ecuación (3) se aplica a un cuadrado de la imagen, si se aplica siguiendo una trayectoria de arriba hacia abajo, los bordes derecho e inferior de la imagen no se podrán segmentar adecuadamente, debido a que pueden quedar un ancho menor a el cuadrado de análisis, para ello se puede realizar dos cosas, una es crear pixeles falsos que permita recorrer toda la imagen, o simplemente aplicar a un marco del ancho del bloque de segmentación.

3.3.2. Mejoramiento del Histograma

Se comparó el resultado de la ecualización del histograma utilizando la máscara de segmentación y sin ella, y la normalización como se sugiere en la literatura.

El problema de la normalización es que se deben elegir valores fijos para todas las imágenes, ello ocasiona que para ciertas imágenes el contraste sea buena y para otras sea malo, se utilizó los parámetros de [9] de $M_0=100$ y $V_0=100$ y los resultados son satisfactorios para todas las imágenes, sin embargo se utilizó la máscara de segmentación para las medidas de M y V , lo cual incrementó la diferencia entre surcos y crestas como se ve en la Figura 32 (A1, B1 y C1), en ella las imágenes A, B y C pertenecen a la base de datos de la FVC2004, todas las imágenes han sido recortadas por el sistema.

Por otra parte la ecualización normal (sin máscara), producen la unión de las terminaciones con las crestas, y un incremento en el ruido de la imagen, lo cual produce malas reconstrucciones de la huella en la etapa del filtrado, por ello este método no se sugiere utilizarlo para las huellas dactilares. Sin embargo se mejoró esto utilizando la ecualización con máscara, con el fin de cambiar la distribución del histograma de las crestas de la huella y no de toda la imagen en general, se aplica como una ecualización normal pero al cálculo del histograma se aplica la máscara de segmentación, el resultado fue una imagen similar a la normalización, como se ve en la Figura 32 imágenes A4, B4 y C4.

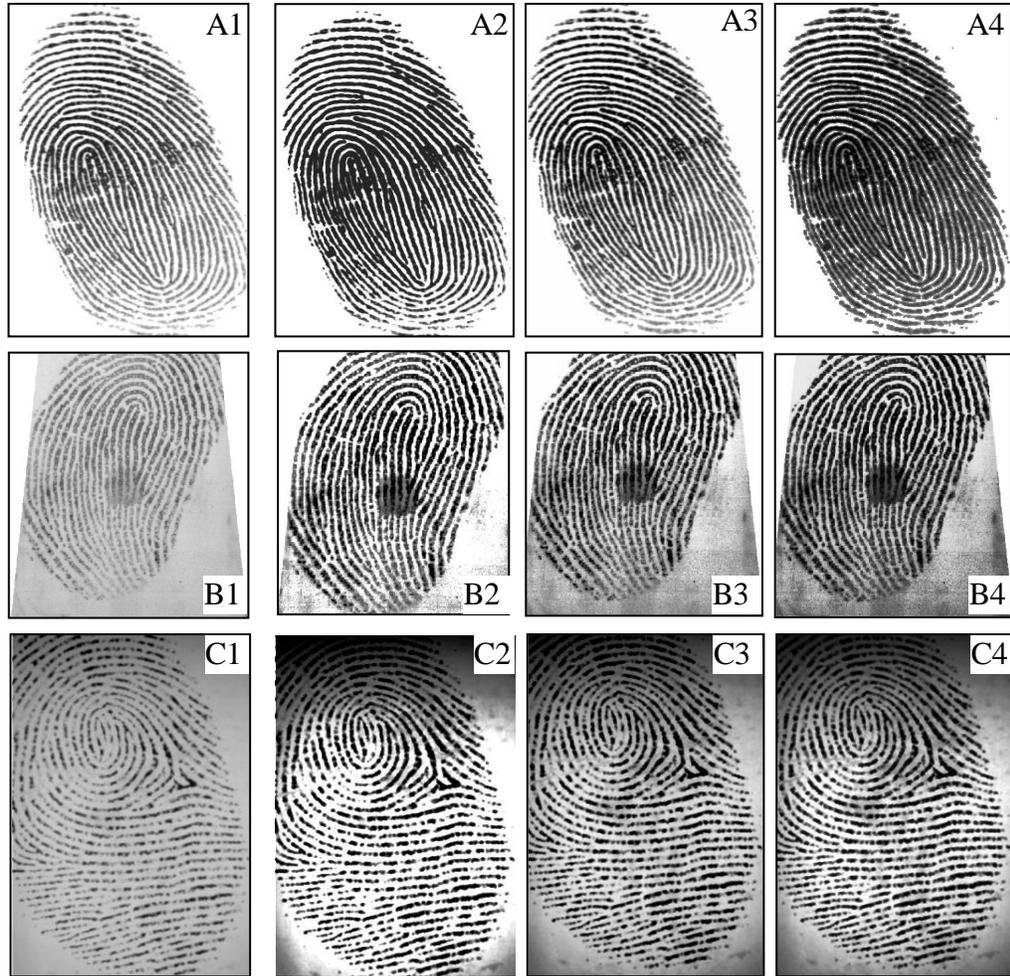


Figura 32. Normalización y ecualización del histograma.

Como resultado de los métodos utilizados, se eligió la normalización porque ofrece una mejor distribución del histograma en general para todas las imágenes, no obstante la ecualización ofrece mejores resultados para algunas imágenes, por ello se puede pensar en un futuro implementar un selector automático para escoger alguno de los dos métodos, con el fin de tener siempre la mejor distribución del histograma. Un hecho importante de elegir la normalización y no la ecualización local, es por el mejor desempeño en imágenes de la DB1_B y DB2_B, las cuales pertenecen al sensor óptico *CrossMatch* y *Digital Persona*, el cual es uno de los más utilizados a nivel comercial, por ello este trabajo se enfocó a este trabajo a estos dos sensores.

En la Figura 32, las imágenes utilizadas pertenecen a: A=71_6_{DB2_B}, B=87_6_{DB4_B}, C=57_4_{DB2_B}, y D es de la base de datos del autor con un tamaño de 225 x 404, los tiempos promedios de usando la ecualización son: 1.87, 1.75, 2.52 y 1.71 ms, respectivamente, con la normalización son: 1.79, 1.78, 2.50 y 1.68 ms.

Se presentaron problemas con las imágenes de la base de datos DB3_A, las cuales fueron adquiridas por un sensor térmico, que en general son oscuras con gran diferencias entre las crestas y los surcos como se ve en la Figura 33, donde A1 y B1 son las imágenes originales, A2 y B2 son mejoradas por la normalización, A3 y B3 por la ecualización local, y A4 y B4 por la ecualización general, cuando se aplican los métodos de mejoramiento del histograma, en la normalización causa un brillo general, aunque en B2 se puede notar cierto mejoramiento, pero las crestas y surcos no presentan mayor distinción entre ellas, lo mismo ocurre con la ecualización local, no obstante la ecualización general empeora la imagen y disminuye la distinción entre crestas y surcos.

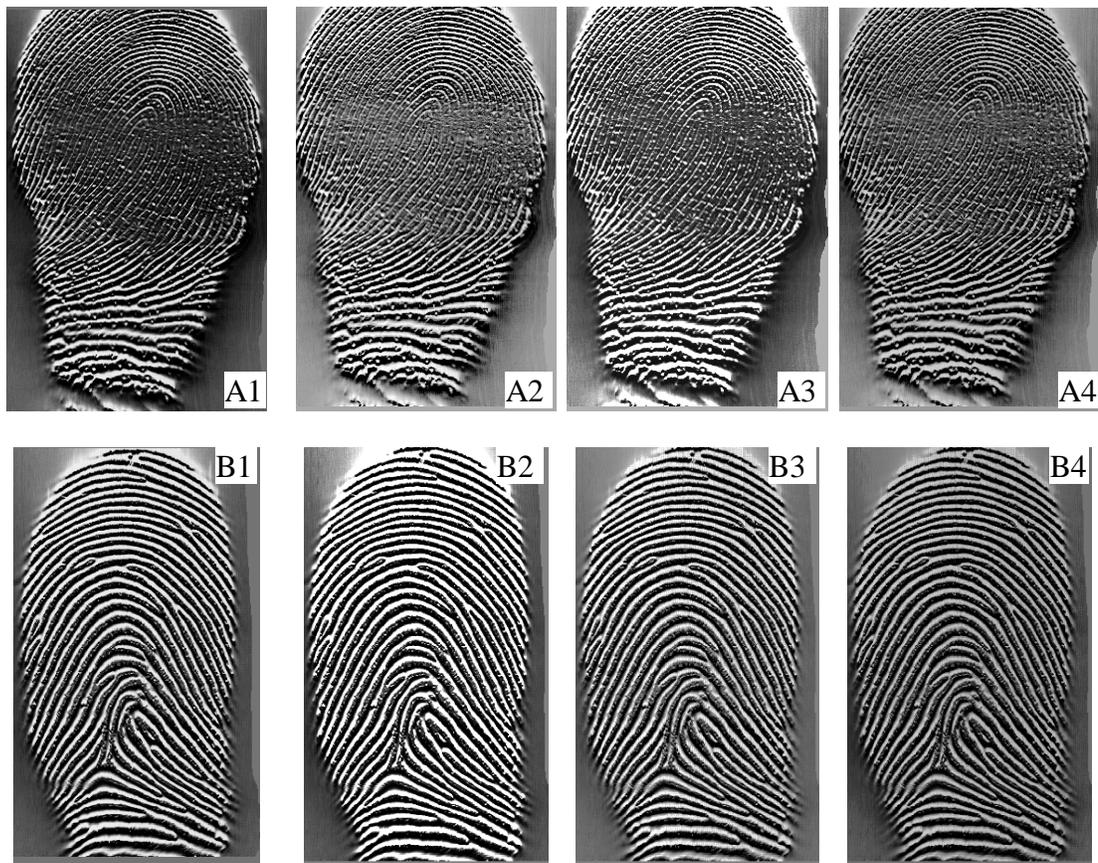


Figura 33. Imágenes del sensor térmico.

Para aplicar la normalización se deben tener en cuenta las siguientes recomendaciones:

- En la ecuación (4), la normalización se establece elevando al cuadrado y después tomar la raíz cuadrada, matemáticamente esto se puede resolver simplemente sacando el término $(I(i, j) - M)$ de la raíz, pero hay que tener presente que esto solo se puede hacer solo si $(I(i, j) - M)$ es mayor a cero, para el caso de valores negativos la expresión se debe cambiar a $-(I(i, j) - M)$, es por ello que la forma más fácil de sacar este término es utilizando el valor absoluto de tal forma que el término cambia a $|I(i, j) - M|$, con ello se gana un poco de tiempo en el procesamiento.

- Elegir valores de M y V que sean más o menos iguales, cuando uno de ellos es muy superior al otro se puede presentar que la imagen quede con un gris uniforme, es decir bajo contraste, o por el contrario un gran contraste que en algunas ocasiones tiende a unir crestas o desaparecerlas.

3.3.3. Estimación de la Orientación

La determinación de la dirección de las crestas depende principalmente de dos parámetros, el tamaño (T) y la desviación estándar (DE) del filtro gaussiano, cuando se utilizan grandes valores de T y DE la estimación se hace mas inmune al ruido; pero con malas reconstrucciones de las crestas que forman el núcleo y el delta. Por el contrario utilizando valores pequeños se obtiene una buena reconstrucción de la huella, pero no se hace inmune al ruido, esto también fue reportado en [19]. Con el fin de obtener calidad en la reconstrucción se eligen valores pequeños, porque obtener una buena estimación de la orientación (*con T y DE grandes*) en regiones con alto ruido no asegura que las minucias encontradas allí sean reales, además se afectaría en general las minucias de la huella y el costo computacional es mayor.

Se ajusto el filtro gaussiano con un $T=31$ y $DE=5$, parámetro establecido en [21]. En todas las imágenes la orientación puede ser muy bien estimada como se ve en la Figura 34, aunque existen caso como la imagen D donde algunas líneas cambian repentinamente el “flujo” de las cresta (*encerradas en el círculo*), sin embargo esta zona hace parte de la máscara de segmentación por lo que no se tendrá en cuenta en procesos posteriores. En todas las imágenes la orientación es estimada en cada pixel de la imagen, la forma de visualización de la Figura 34 se toma con bloques de 10x10 pixel, haciendo un promedio de las orientaciones en cada bloque.

Se intento ajustar el filtro gaussiano con parámetros menores a los mencionados, y los resultados no son buenos, porque el ruido empieza a confundir la dirección real de las crestas, y cuando se aplica el filtro de Gabor las crestas no se reconstruyen bien.

Las imágenes utilizadas en la Figura 34 son: A=71_6_{DB1_A}, B=27_6_{DB2_A}, C=49_5_{DB3_A}, D=49_5_{DB4_A}, E=49_5_{DB3_A} y F pertenece al autor con un tamaño 300 x 438 pixeles, los tiempos promedios de esta operación son: 46, 50, 55, 35, 40 y 45 ms, respectivamente.

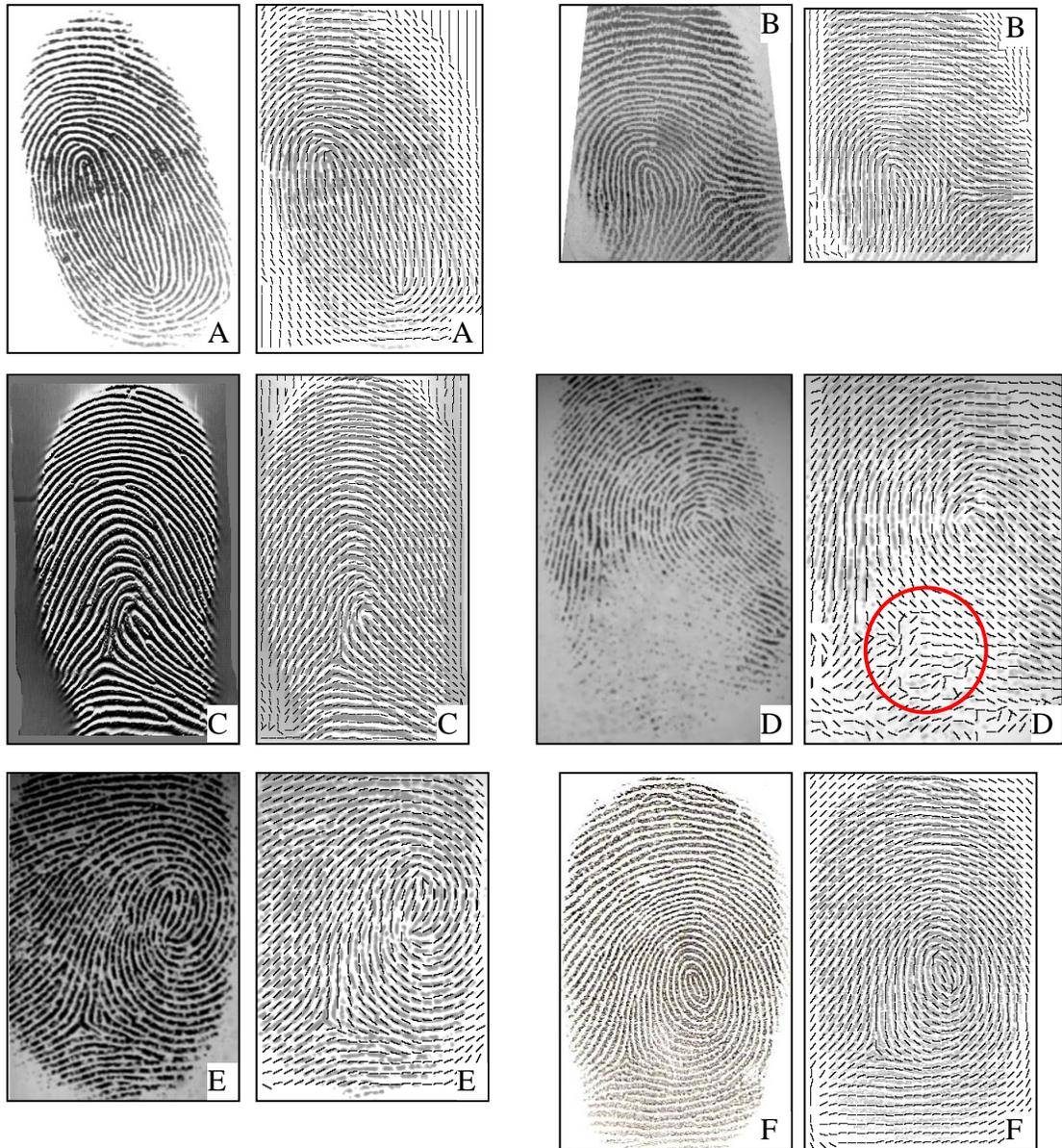


Figura 34. Estimación de la orientación.

Al aplicar los parámetros de arriba en una imagen con mucho ruido como la imagen A1 y B1 de la Figura 35, se observa que el flujo de crestas cambia repentinamente donde el ruido es muy alto (cerca del núcleo y en las cortaduras como se ve en la imagen B1), en estos dos casos la $T=31$ y $DE=5$ no son adecuados, como se ve en A2 y B2; existe una solución rápida para este problema y es cambiando los parámetros del filtro gaussiano a $T=71$ y $DE=20$, con esto se obtiene una buena reconstrucción de la orientación de las crestas (imagen C), más adelante en la sección 2.3.2.8 se verá las desventajas de este método, por ello otra solución a este problema se da en la siguiente sección 2.3.2.6.

Las imágenes utilizadas en la Figura 35 son: $A=94_3_{DB3_A}$, $B=3_8_{DB1_A}$, el tiempo de procesamiento cuando se utiliza $T=71$ y $DE=20$ son: 137.94 y 76.083 ms, comparado con $T=31$ y $DE=20$ de 54.57 y 35.99 ms, son mucho menores.

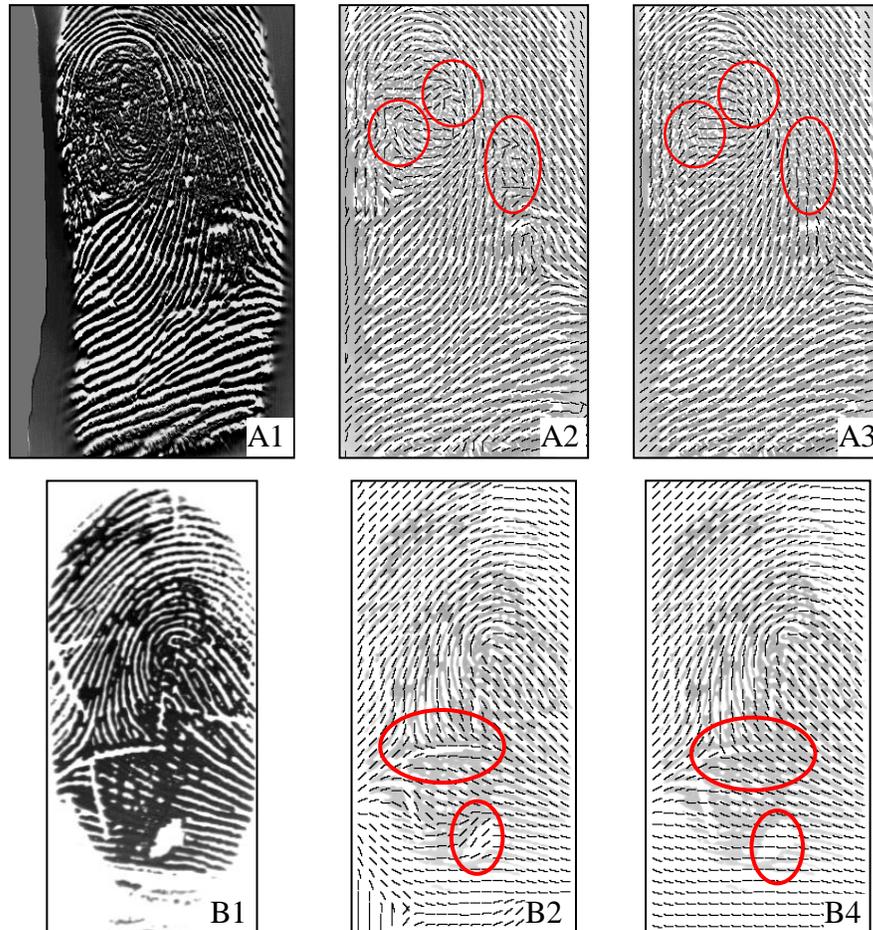


Figura 35. Estimación de la mala orientación.

Para entender mejor como se debe aplicar el suavizado en ésta etapa, llamemos entonces a la función que aplica el filtro gaussiano como *Gauss*, el cual esta afinado con un tamaño de 31 y desviación estándar de 5, entonces los procesos que se deben seguir son:

- Primero se estima el gradiente V_x de la formula (5) y se aplica dos veces *Gauss* (V_x).
- Se eleva al cuadrado de δ_x a δ_y , después a cada uno se aplica *Gauss* (δ_x^2) y *Gauss* (δ_y^2), se realiza la diferencia entre las dos expresiones, y a esta nuevamente se aplica un *Gauss* ($\delta_x^2 - \delta_y^2$), con ello se puede calcular V_y que aparecen en la ecuación (6), y se aplica *Gauss* (V_y).
- Con ello se puede aplicar ecuación (7), el cual determina la orientación final, a la cual se le debe corregir la orientación para que los ángulos de 0° sean paralelos a el eje axial, y los de 90° sean perpendiculares a el eje axial, ello se puede hacer que para

todos los valores de la matriz de orientación mayores a 90° restarles 90° , y para el resto sumarle 90° .

La imagen que resulta de la orientación es la que se ve en la

Figura 36, la cual está normalizada entre 0 y 255, donde las regiones negras son valores de 0° , las blancas de 180° y el resto son valores intermedios de estos ángulos.

Una de las aplicaciones que tiene esta imagen es para la determinación del núcleo y el delta, haciendo por ejemplo el uso del índice de Poncairé [13], el cual trata de encontrar la zona roja señalada en la

Figura 36, el cual es una región donde están presentes todas las orientaciones entre 0° y 180° , estos puntos se suelen utilizar para alinear la huellas, no obstante no se utilizó en el trabajo, porque como se observó en la Figura 4, existen huellas que no poseen núcleo o delta, entonces se estaría limitando al algoritmo a ciertas huellas, además es necesario que posean dos núcleos o deltas, o cualquier combinación de estos.

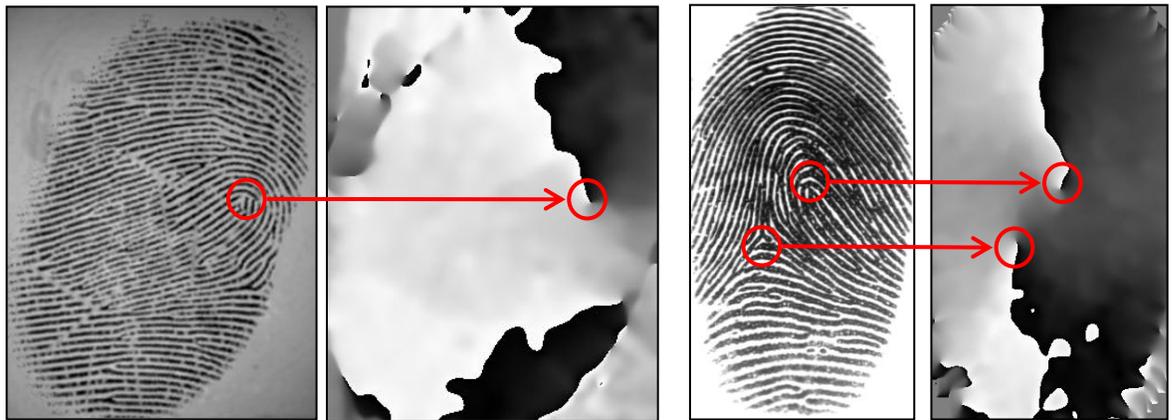


Figura 36. Matriz de orientación original.

3.3.3.1. Máscara de Confiabilidad

Los resultados de la máscara de confiabilidad, ayudan en gran medida a ocultar zonas donde la reconstrucción de la huella es mala, además esto ayuda a disminuir el coste computacional del filtrado. Los parámetros utilizados en el trabajo fueron $\varepsilon = 0.001$ y $\beta = 0.003$, esta matriz utiliza los valores de la estimación de la orientación pero con parámetros de $T=31$ y $DE = 3$, la máscara en las imágenes se puede ver en la Figura 37, donde se aplica para distintas imágenes con altos y bajos niveles de ruido, en algunas la máscara encierra los núcleos de las huellas como A, por otra parte en las huellas C con poco ruido es afectada muy poco como era de esperarse; en B y D las crestas apenas notorias son también

enmascaradas, porque así se logre reconstruir esta zona, no se va a tener certeza sobre si hay o no minucias allí.

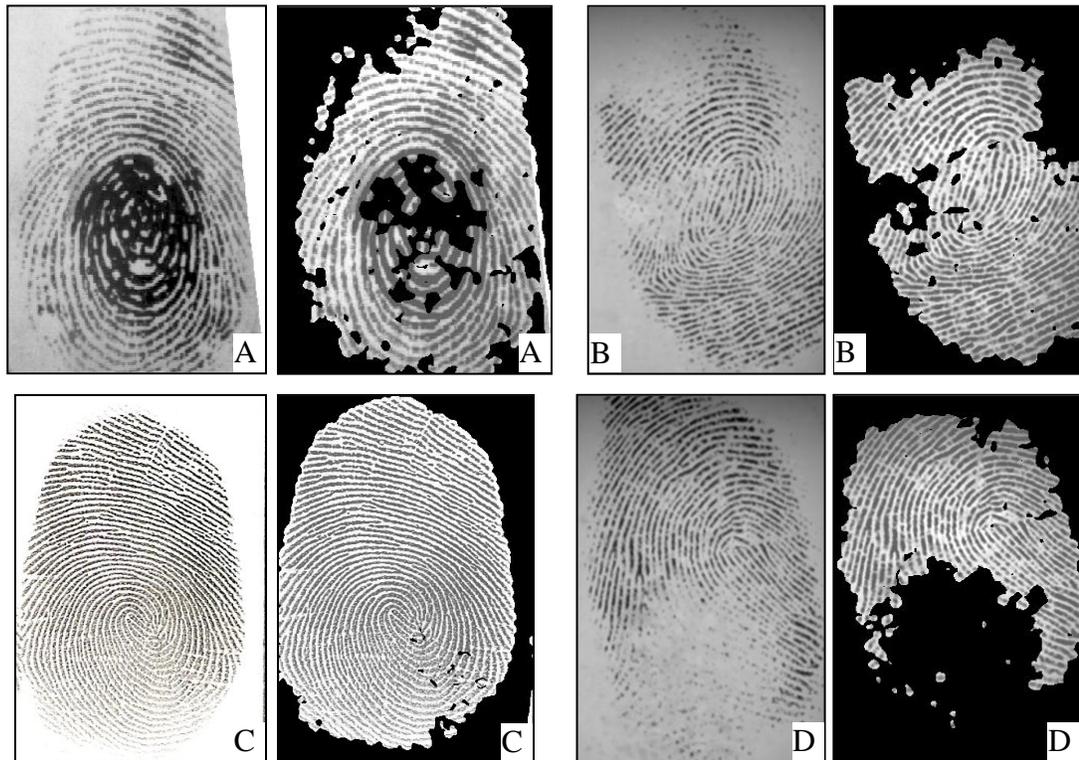


Figura 37. Matriz de confiabilidad.

Esta máscara se sitúa principalmente en zonas donde la huella tiene mala calidad, por ejemplo donde las crestas se unen por el sobre-entintado, donde el ruido es alto, etc., sin embargo también en los núcleos y deltas de la huella, pruebas previas sugieren que la reconstrucción en estas zonas (núcleos y deltas) es muy difícil de obtener si la imagen tiene mucho o poco ruido, porque las crestas toman unas formas muy inusuales, con altas curvaturas y producen la mayoría de falsas minucias (ver Figura 20), que en este caso no son falsas ya que corresponden a la huella.

Es por ello que el ajuste de estos parámetros es bastante difícil, ya que si se disminuye ε y β no se enmascaran los núcleos y deltas, pero tampoco las zonas corruptas de la huella. Por ello se dejan fijos los valores antes mencionados, los cuales quitan cierta cantidad de minucias que a la hora de un análisis no se podrían determinar qué tipo de minucia son.

Además el tiempo de este proceso es un poco lento, porque se debe volver a calcular la orientación de la imagen y con ella hacer las operaciones, no obstante la suma de este tiempo más el de la estimación de la orientación con $T=31$ y $DE=5$, es igual o menor a estimar la orientación con $T=70$ y $DE=20$.

Las imágenes utilizadas en la Figura 37 son: A=13_8_{DB2_A}, B=81_4_{DB4_A}, C es del autor con un tamaño de 320 x 483 pixel, D=38_7_{DB4_A}, E=24_7_{DB1_A}, F=4_5_{DB3_A}, el tiempo de procesamiento es: 47.03, 42.53, 61.59, 36.74, 42.53 y 51.24 ms, respectivamente.

Recordando las imágenes de la Figura 35, las cuales no se podían estimar la orientación (en ciertas zonas), cuando se aplica esta máscara el resultado se puede ver en la Figura 38, donde se observa cómo se cubre estas áreas evitando así más adelante una mala reconstrucción o falsas detecciones.

Las imágenes utilizadas en la Figura 38 son: A=94_3_{DB3_A}, B=3_8_{DB1_A}, el tiempo de procesamiento es: 50.77 y 33.56 ms, respectivamente.

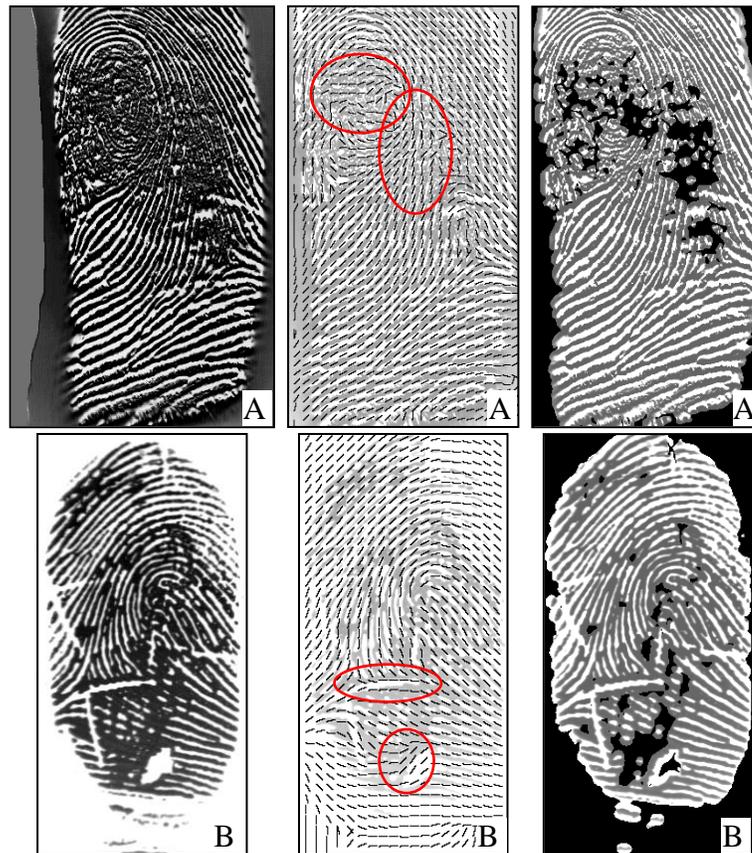


Figura 38. Máscara de confiabilidad para las imágenes de la Figura 35.

Determinar la máscara de confiabilidad es sencillo, solo se deben seguir las ecuaciones (8), (9), (10) y (11); los dos únicos parámetros importantes en ellos son ε y β , sin embargo estos son muy sensibles, es decir que un pequeño cambio produce un aumento notorio en la máscara, además si se quiere que la máscara afecte zonas muy específicas se debe tener un bajo suavizado en T y DE , porque cuando este aumenta existe una tendencia a cubrir partes de la huella que no tienen ruido.

3.3.4. Estimación de la Frecuencia

La frecuencia va muy ligada con una buena estimación de la orientación, debido a que el bloque para determinar la frecuencia debe ser perpendicular a la dirección de las crestas, para ello se fijo rango de frecuencias entre 0.19 y 0.07 Hz y un ancho de bloque $w = 16$ parámetros establecidos por [21], aunque el tamaño se cambio.

Los resultados iniciales dejaban algunas zonas en la dentro de la huella con partes negras (0), es decir que no se logro determinar la frecuencia porque estaba fuera del rango, esto ocurre debido al ruido, la mala estimación de la orientación ó zonas de alta curvatura. Ver Figura 39 columna A.

Para evitar esto se tomo el promedio de los pixeles vecinos que tengan definida su frecuencia, recordando la misma notación de la Figura 16, pero en este caso cada pixel es un bloque de 16×16 . Entonces el promedio es igual a la suma de los pixeles vecinos, dividido sobre el número de pixeles vecinos distintos de cero, por ejemplo si se tiene lo siguiente:

Caso donde todos los pixeles están llenos, el promedio será $= 1.08/8 = 0.135$

0.12	0.12	0.15
0.13	0	0.15
0.12	0.14	0.15

Cuando algunos pixeles son cero, el promedio será $= 0.82/6 = 0.136$, y no $0.82/8 = 0.103$.

0	0.12	0.15
0.13	0	0.15
0.12	0	0.15

Con esto se logra recuperar todas las zonas dentro de la huella, como se ve en la Figura 39 columna C, las recuperaciones se ven señaladas por círculos. La imágenes A y D son huellas con buena calidad donde no se logró establecer frecuencia adecuadamente, en A estas zonas corresponden a regiones cerca al núcleo, y en D es una zona dentro de la mancha negra que se genera cuando se registra la huella. Además en todas las imágenes se recuperan zonas que no hacen parte de la huella, un ejemplo claro es la imagen C, sin embargo después es aplicado la máscara de confiabilidad para evitar estas zonas.

Se probó el uso de $w=32$, pero sin variar el ancho de cuando se van a contar las columnas que era de $[2w, w]$, en lugar de ello se usó $[w, w]$, el utilizar un bloque tan grande no es muy adecuado para imágenes donde la huella queda al borde, ya que aparecerá un recuadro negro en todos los bordes de la imagen, sin embargo se puede utilizar pixeles falsos fuera de la imagen para que se pueda registrar en los bordes. En la siguiente sección 2.3.2.8, se ve

observará la ventaja de usar la frecuencia local contra una frecuencia fija. Las imágenes utilizadas en la Figura 39 son: la imagen A imagen propia del autor de tamaño 361 x 469, B = 24_7_{DB1_A}, C = 69_8_{DB4_B}, y D=18_2_{DB2_B}, los tiempos promedios de esta operación para estas imágenes son: 45.59, 38.72, 51.1 y 20.4 ms, respectivamente.

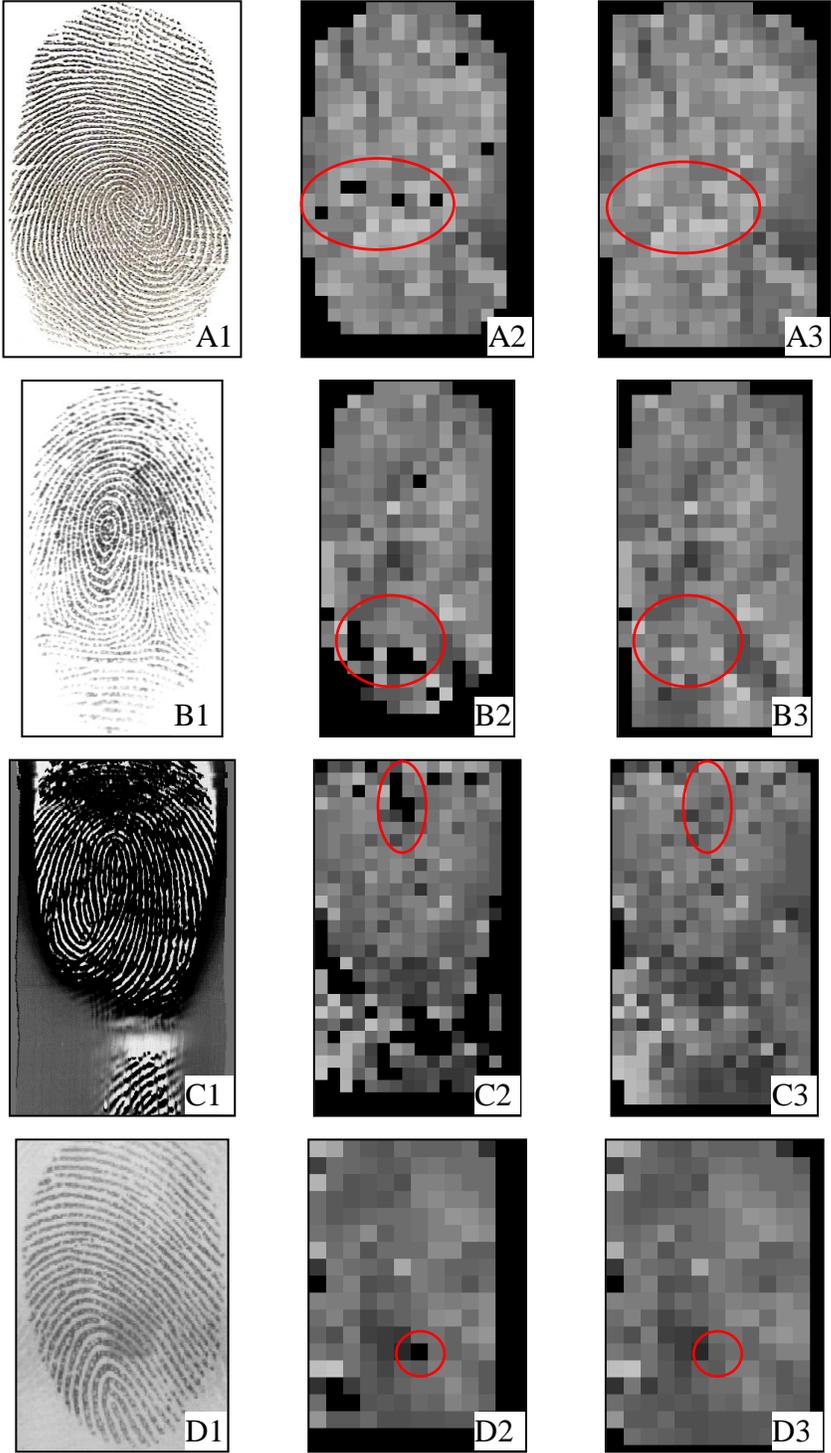


Figura 39. Estimación de la frecuencia.

Las recomendaciones para la estimación de la frecuencia son las siguientes:

- Cuando se debe rotar el bloque de análisis en sentido perpendicular a la orientación, se debe promediar los ángulos que estén en el bloque, comúnmente se tomaría la suma de todos los valores dividido por el número de valores, no obstante Peter [21] nos da la sugerencia que se tome el promedio con estas ecuaciones:

$$y = \frac{1}{w^2} \sum_{u=0}^{w-1} \sum_{v=0}^{w-1} \sin(2\theta(i+u, j+v)) \quad (25)$$

$$x = \frac{1}{w^2} \sum_{u=0}^{w-1} \sum_{v=0}^{w-1} \cos(2\theta(i+u, j+v)) \quad (26)$$

$$\theta = \frac{1}{2} \operatorname{atan2}\left(\frac{y}{x}\right) \quad (27)$$

Donde se debe w significa el ancho del bloque, y $\theta(i, j)$ es la matriz que contiene las orientaciones, entonces se debe primero convertir los valores de $\theta(i, j)$ en sus valores originales de V_x y V_y , y a estos se les saca el promedio, para reconvertirlas se utiliza $\operatorname{atan2}$, que en este caso se trata de el ángulo en radianes.

- En el trabajo se eligió $w = 16$, el cual es un valor pequeño, se debería tener en cuenta la creación de pixeles falsos para que cubra toda la imagen, y así al borde de esta se pueda determinar correctamente la frecuencia, cuando se haga ello se podría pensar en cambiar el ancho $w = 32$, que en algunas ocasiones puede resultar más ventajoso.

3.3.5. Filtrado

Esta es la etapa más importante para el mejoramiento de las huellas, no obstante se necesita tener bien establecido todos los valores de orientación y frecuencia, para obtener una buena reconstrucción de la huella.

Los parámetros de ajuste del tamaño del filtro depende de los valores de frecuencia, k_x y k_y , que en este caso son iguales a 0.5 (esto la convierte en una envolvente circular) y $\psi=8$. Debido a que la frecuencia puede tomar valores (como 0,122345) con 6 decimales, se ajusto a solo dos decimales (0,12), entonces los tamaños del filtro, igual al valor de w_x y w_y ($w_x = \psi k_y / \omega$) toman valores de: 57, 50, 44, 40, 36, 33, 31, 29, 27, 25, 24, 22 y 21. Estos son los mismos parámetros de Kovési[21], también se probaron con otras configuraciones, pero esta fue la mejor.

En este trabajo se aplica el filtro de Gabor como una convolución espacial entre la imagen y la matriz del filtro. Para optimizar los procesos, los filtros son pre-calculados y guardados

antes de usarse. La función de Gabor no se normalizó, ya que cuando se aplicaba de esta forma los máximos y mínimos de la imagen se saturan, con valores para las crestas por debajo de cero (0), y los surcos por encima de uno (1) (porque cuando se aplica un filtro como éste se debe manejar imágenes entre 0 y 1 tipo *float*, y no entre 0 y 255 *int*), entonces la binarización es más sencilla de hacer.

Se probaron distintos ajustes del filtro de Gabor, variando especialmente la orientación y la frecuencia, para todos ellos se dejó la normalización como etapa estándar para la adecuación del histograma, y en el filtro siempre se utilizó los valores de k_x y $k_y = 0.5$ y $\psi=8$.

La primera (*mejoramiento 1*) prueba se hizo con:

1. Orientación con $T=31$ y $DE=5$.
2. Frecuencia con $w=16$.

Los resultados se pueden ver en la Figura 40.

En el mejoramiento 1, las huellas presentan buenos resultados, se puede ver minuciosamente como la imagen mejorada no cambia las formas de las crestas, claro está que el ruido se quita, para las huellas A1, A2, B1, B2, D1 y D3 se presentan las mejores reconstrucciones, a pesar de que estas imágenes tienen altos niveles de ruido, tales como crestas sin completar, cortaduras, sobre-entintado y sub-entintado; las cuales son adquiridas con sensores ópticos y generadas por computador.

Debido a que no se usa una frecuencia estable para todas las imágenes, el filtro se puede adaptar muy bien cuando las crestas están muy alejadas o muy cerca, porque se varía el ancho de cada filtro de Gabor, un ejemplo donde las crestas se ven separadas es en A1, comparada con las de D2 que están poco espaciadas.

Unas de las imágenes que más contenía ruido son A2 y D1, la primera de un sensor óptico y la otra sintética, las cuales tienen una gran cantidad de ruido y cortaduras, además el fondo tiene unos niveles de gris cercanos a los de las crestas, lo cual lo hace más difícil, no obstante el filtro de Gabor logró mejorar la huella satisfactoriamente, aunque la parte señalada en rojo no se logró mejorar, el resto de la huella contiene información suficiente para compararla con otra.

Para las imágenes tomadas con el sensor térmico, como C1 no se tiene una buena reconstrucción en toda la huella, debido al alto nivel de ruido que presentan algunas zonas (*encerradas en un círculo*), esto se debe a la mala estimación de la orientación, sin embargo evitando esta zona se logró obtener la información del resto de la imagen, que es muy rica en minucias, en C2 que es del mismo sensor, se obtuvo una buena reconstrucción porque ésta contenía muy poco ruido, no obstante se agregó para ver la separación de las crestas, comparando C1 y C2 es evidente que en C2 las crestas están mucho más separadas que en C1, por ello el uso de una frecuencia fija ofrecería resultados bueno para una de ellas, mientras se empeora para la otra, y viceversa.

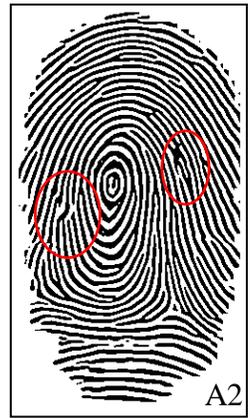


Figura 40. Filtro de Gabor con los parámetros N , $T=31$ y $DE=5$, $w=16$.

La segunda (*mejoramiento 2*) prueba se hizo con:

1. Orientación con $T=31$ y $DE=5$, $T=51$ y $DE=10$, $T=71$ y $DE=20$.
2. Todos con frecuencia $w=16$.

Los resultados se pueden ver en la Figura 41, donde $C1$, $B1$ y $D1$ son las imágenes originales, $C2$, $B2$ y $D2$ tienen parámetros de orientación de $T=31$ y $DE=5$, $C3$, $B3$ y $D3$ tienen parámetros de orientación de $T=51$ y $DE=10$, y por último $C4$, $B4$ y $D4$ tienen parámetros de orientación de $T=71$ y $DE=20$.

En el mejoramiento 2, se eligieron estas 3 huellas para ver las diferencias cuando se varía los parámetros en la orientación. Cuando se va aumentando T y DE la dirección de la orientación se vuelve más inmune al ruido, cuando se aplica el filtrado para algunas imágenes la reconstrucción es buena, mientras para otra es mala.

Por ejemplo en C el aumento de T y DE disminuye la mala reconstrucción en la zona en rojo, no obstante el ruido es tan grande que así las crestas sigan el camino que deberían no se reconstruye bien la imagen, más aun mirando bien esta zona no se determina a simple vista que clase de minucias hay allí, y si por ejemplo un dactiloscopista no ve las minucias por el ruido, no podrá determinar que es realmente lo que hay allí, así el programa indique cualquier cosa, ya que él tiene la última decisión sobre si dos huellas son iguales o no.

En el caso de B el aumento de T y DE tiene consecuencia graves en la reconstrucción, para este caso se trata de una huella de tipo *verticilo*, esta presenta dos núcleos centrales, ello implica que sus crestas cerca a estas zonas tendrán una alta curvatura, al aplicar mayor suavizado esta alta curvatura es afectada y se pierde, entonces al aplicar el filtro de Gabor en una dirección no adecuada el resultado es una mala reconstrucción, señalados por un círculo rojo. Por ello como este es un tipo de huellas comunes, el alto suavizado no es aconsejado.

Para el último caso en D el aumento de T y DE no presenta mayores cambios en la huella, aunque la zona señalada en rojo se nota una leve mejoría, debido a que la curvatura de sus crestas es poco, para el resto de la imagen solo algunas minucias cambian y unas crestas se completan.

Como se miro en los casos, el aumento del suavizado no proporciona una buena reconstrucción de las crestas para todo tipo de huella, por ello es mejor mantener un suavizado bajo, esta conclusión fue hallada también por Duque [19].

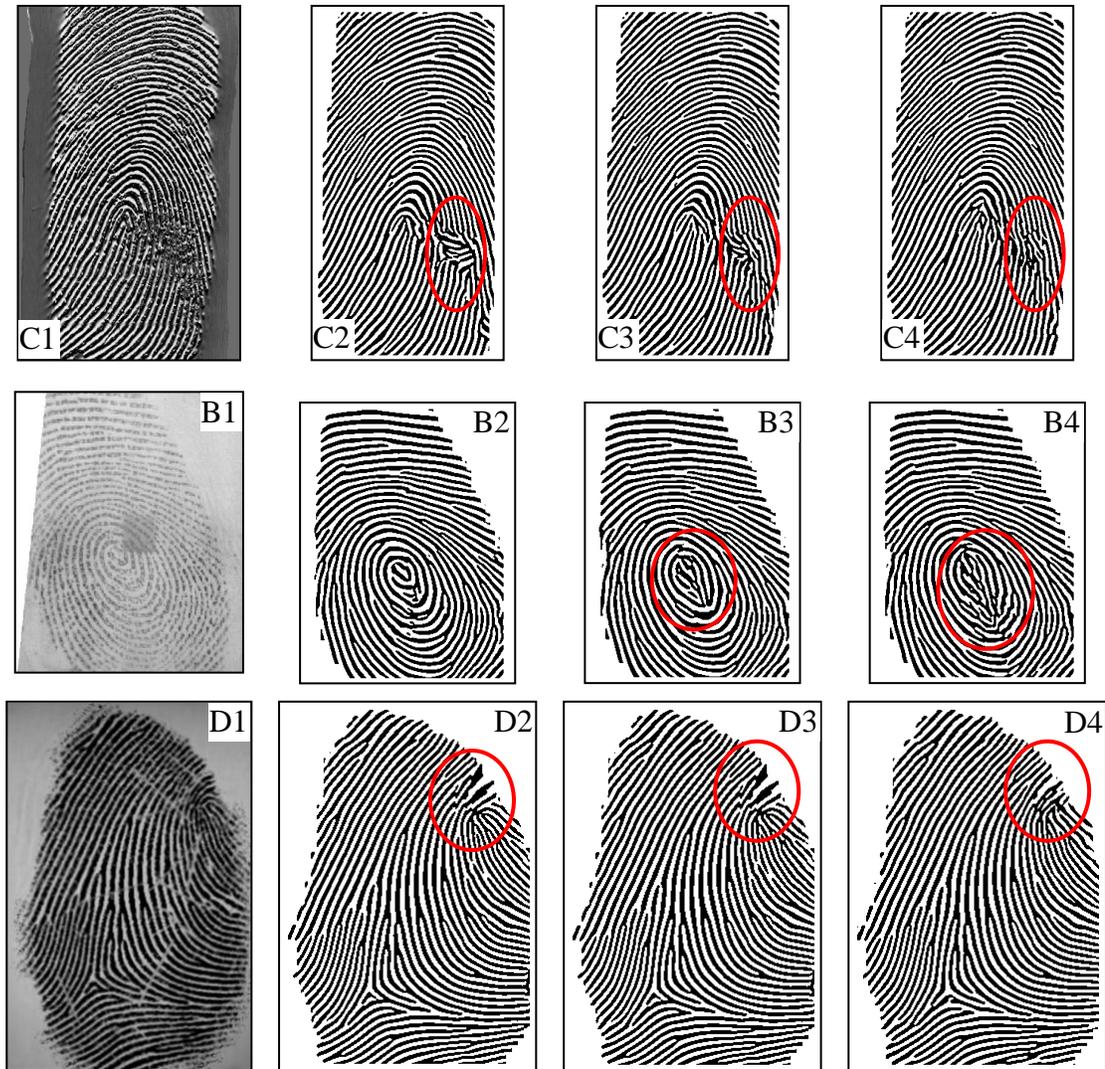


Figura 41. Reconstrucción variando la orientación.

La tercera (*mejoramiento 3*) prueba se hizo con:

1. Orientación con $T=31$ y $DE=5$, $T=51$ y $DE=10$, $T=71$ y $DE=20$.
2. Usando las frecuencia de $w=32$ y $w=16$.

Los resultados se pueden ver en la Figura 42, donde Comparación en la reconstrucción de las huellas usando el filtro de Gabor, B1 imagen original, B2 con $[T=31, DE=5, w=16]$, B3 con $[T=51, DE=10, w=16]$, B4 con $[T=71, DE=10, w=16]$, B5 con $[T=31, DE=5, w=32]$, B6 con $[T=51, DE=10, w=32]$, B7 con $[T=71, DE=20, w=32]$.

En el mejoramiento 3, se vario el tamaño w para la frecuencia, y T y DE en la orientación, cuando se utilizo $T=31$, $DE=5$ y $w=32$, los resultados no presentan grandes cambios que con respecto a usar $w=16$, solo algunas minucias cambian de terminación a bifurcación, y

viceversa. Por lo cual el tamaño de la frecuencia no influye en gran medida en la reconstrucción de la huella, aunque muchos autores utilizan un bloque de frecuencia de 32, porque se estima mucho más rápido que con un bloque de 16. No obstante cuando se utiliza una frecuencia grande, la estimación de esté en los bordes muchas veces no se obtiene, esto por la forma en cómo se aplica aquí, por lo tanto se elimina información de minucias.

El suavizado de la orientación sigue afectando las huellas, esto ocurre para $T=51$ y $DE=10$, y para $T=71$ y $DE=20$, en ambos casos la reconstrucción final es mala, esto lo demuestra las imágenes B3, B4, B6 y B7, donde las minucias del núcleo se pierden.

En general en todos los casos las terminaciones y bifurcaciones se confunden por el ruido de la imagen, y se cometen “errores” al determinar qué tipo de minucia es, no obstante este error se propaga en todas las huellas de un mismo dedo, por lo cual siempre se va a identificar como la misma minucia cuando se comparen, más aun en el trabajo de Chikkerur [15], la distinción entre minucias no se tiene en cuenta, entonces no será necesario distinguir las entre ellas, y ya no serian más errores.

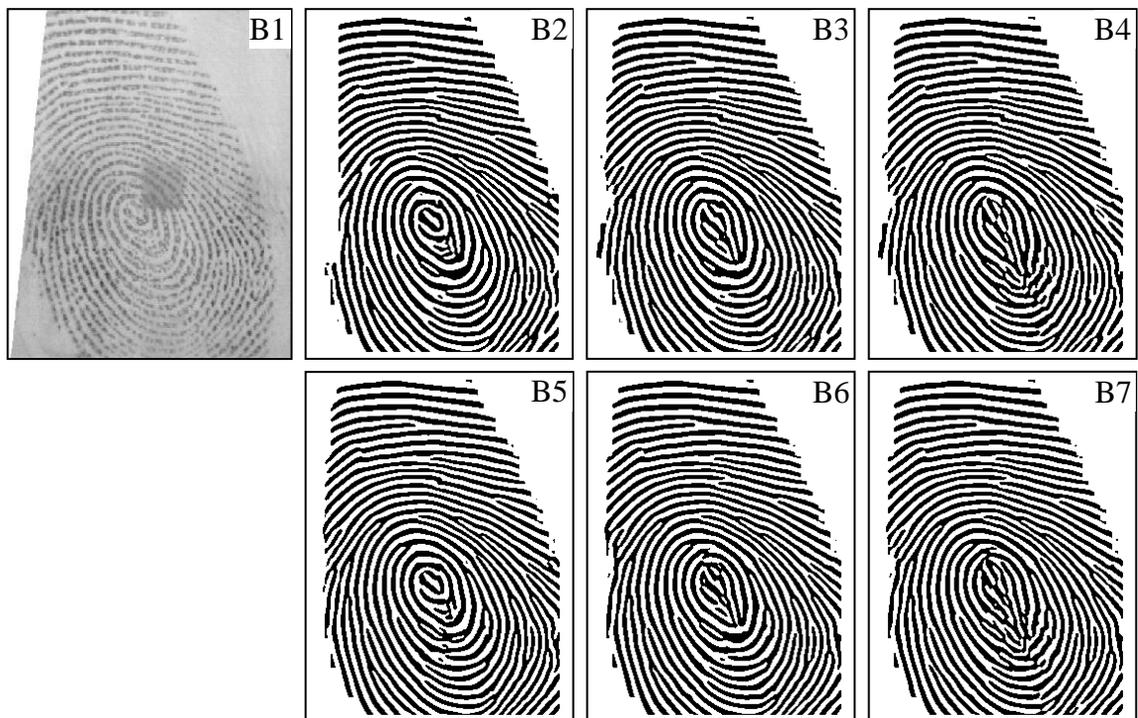


Figura 42. Reconstrucción variando orientación y frecuencia.

Las imágenes anteriores utilizadas son: la imagen $A1 = 90_2_{DB1_A}$, $A2 = 24_7_{DB1_A}$, $B1=10_2_{DB2_A}$, $B2=13_5_{DB2_A}$, $C1=4_1_{DB3_A}$, $C2=53_1_{DB3_A}$, $D1=16_8_{DB4_A}$ y $D2=19_4_{DB4_A}$, con tiempos de procesamiento de 176.53, 249.23, 299.5, 254.06, 316.37, 434.36, 210.25 y 285.25 ms, respectivamente.

También se realizó la comparación con diferentes algoritmos de mejoramiento, que usan el filtro de Gabor con una frecuencia fija [7, 17-19] (*algoritmo A_1*), el filtro anisotrópico (*algoritmo A_2*) y el filtro de mediana direccional reportados en [26] (*algoritmo A_3*), por ultimo nuestro algoritmo que usa el filtro de Gabor y frecuencia local variable (*algoritmo A_4*).

El Algoritmo A_1 usan una frecuencia de 0.125 Hz y el filtro de Gabor afinado en 8 orientaciones entre 0° – 180° , es decir pasos de ángulos de 22.5° , después se usa un selector para determinar las mejores reconstrucciones de la huella, éste algoritmo fue implementado en el trabajo y los valores especificados en [17]. El selector se realizó con el efecto de posterizar la matriz de orientaciones, es decir se toma la matriz que está entre 0° y 180° , y se hace que todos los ángulos entre 0 y 22.5° sean $22.5^{\circ}/2$, después los que están entre 22.5° y 45° se aproxima a $45^{\circ}/2$, y así para el resto de orientaciones. A ello se aplica el filtro de Gabor y se obtiene la reconstrucción. El filtro de Gabor afinado a una sola frecuencia arroja buenos resultados, sin embargo en las áreas del núcleo y delta su reconstrucción es mala, con el método propuesto se eliminan estos problemas (*los problemas están señalados con círculos rojos*).

Cuando se utiliza el filtro aniso-trópico deja muchas crestas incompletas, y la cantidad de minucias falsas es alta, este filtro en realidad no se aplica en una dirección específica, por lo cual no necesita de la orientación de la imagen, éste es un filtrado poco adecuado para este tipo de imágenes, ya que cuando se binariza no se presentan mayores cambios en la huella.

Con el filtro de mediana direccional se eliminan muchos de estos errores, ya que este se aplica en varias direcciones de acuerdo a la dirección de las crestas, este puede ser un buen método para imágenes con poco ruido, no obstante en la realidad la mayoría de imágenes presentan alto ruido, inclusive en la imagen de la Figura 43 las crestas rotas siguen prevaleciendo.

En comparación de los 3 métodos expuestos el filtro de Gabor con frecuencia variable, presenta un mejor rendimiento en cuanto a la reconstrucción de la huella, sin embargo el procedimiento para usar el filtro de Gabor con una frecuencia es más rápido, y para huellas con un ruido aceptable este funciona muy bien.

En la Figura 43 se pueden ver las comparaciones de estos algoritmos. Donde se tienen la comparación usando una frecuencia única, y con la medida de la frecuencia local, basado en los parámetros de Hong [13]; todas las imágenes están binarizadas.



Original



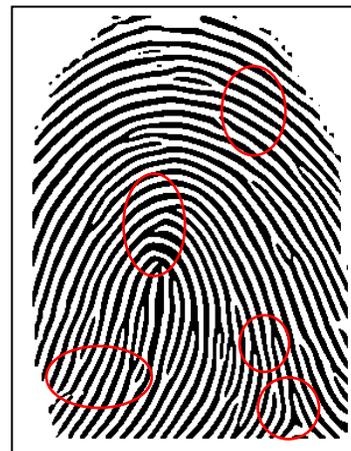
Aniso-tropical



Media Direccional



Gabor una Frecuencia



Gabor varias Frecuencias

Figura 43. Comparación del filtro de Gabor con otros.

Cuando se aplica el filtro de Gabor se deben tener en cuenta estas recomendaciones:

- Se observó anteriormente como los efectos de la orientación y la frecuencia afecta la reconstrucción final, otro parámetro que afecta a la huella es ψ , en el trabajo se aplico $\psi = 8$, cuando este valor es muy bajo o alto se compromete seriamente el kernel de convolución, para valores muy bajos la huella presenta un fondo negro, y algunas de las crestas tenderán a unirse, por el contrario con un valor alto la reconstrucción es buena, pero el tamaño del kernel aumenta considerablemente, en lo que se traduce en mayor tiempo de procesamiento, esto se puede ver en la Figura 44, donde $\psi = 4$ presenta unas zonas negras señaladas con un círculo, $\psi = 8$ y $\psi = 12$ no presentan cambios.

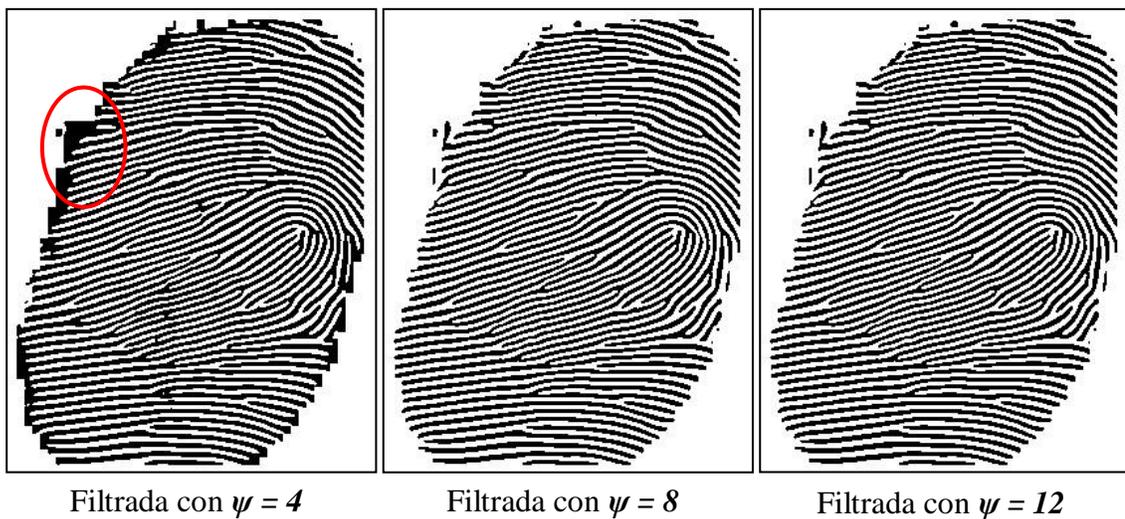


Figura 44. Reconstrucción de la huella variando ψ .

- Los filtros pre-calculados son una gran ayuda, permiten mejorar mucho el tiempo de procesamiento, en los primeras etapas del trabajo no se utilizaba esto, y el tiempo de mejoramiento de una huella tardaba alrededor de 40 segundos, cuando se implemento estos filtros se redujo a 0,5 segundo, lo cual es 80 veces más rápido, en un futuro se puede pensar en pre-calculando los filtros y guardarlos en un archivo de texto, ya que en el trabajo siempre se están calculando una y otra vez para cada huella.

3.3.6. Conclusiones Parciales del Mejoramiento

Respecto a la estimación de la orientación

Se encontró que aplicando más suavizado a la estimación de la orientación, es decir aumentando el tamaño y la desviación estándar, se obtiene una mala reconstrucción de las zonas cerca al delta y núcleo; las cuales son ricas en minucias, pero se puede estimar mejor las minucias en las áreas fuera de ella que contengan alto o poco ruido, aunque con una tendencia a unir las terminaciones a las crestas. Si se realiza lo contrario (pequeño tamaño y desviación estándar) se obtiene una buena reconstrucción de la huella, preservando la forma real de las minucias, el inconveniente se presenta cuando se tiene mucho ruido en la imagen donde la reconstrucción es de mala calidad.

Respecto a la máscara de confiabilidad

La determinación de la matriz de confiabilidad es muy buena segmentando zonas corruptas de la huella, aunque esto implique en algunas ocasiones la eliminación de buenas minucias, pero se concluyo que es mejor eliminarlas ya que se presentan confusión cuando se comparan las minucias.

Respecto a la frecuencia

La determinación de la frecuencia local es un proceso que toma poco tiempo, y aumenta la calidad de la reconstrucción, permite adaptar el filtrado a huellas con poco o mucho espacio en sus crestas; comparado con la aplicación a una sola frecuencia la cual se aplica más rápido pero su reconstrucción es de calidad media. Se concluye que para aplicaciones donde no se requiera una reconstrucción de calidad es mejor aplicar el filtrado propuesto por [9], pero para reconstrucciones por ejemplo de huellas latentes es mejor aplicar el filtrado con frecuencia local.

Respecto a la etapa del filtrado

Para el filtrado se concluye que los parámetros ajustados para el filtro propuestos por [11], ofrecen buenos resultados, y se determino que los aspectos más influyente para esto son la buena determinación de la frecuencia y la orientación de las crestas, además el uso de los filtros pre-calculados es una forma eficiente para hacer la convolución espacial, y obtener así la imagen mejorada. Además el uso del filtro de Gabor ayuda a eliminar la mayoría de ruido presente en la huella, sin cambiar la estructura de las crestas. Comparando el algoritmo de filtrado con otro tipo de filtros propuestos en la literatura, se puede decir que el filtro de Gabor tiene mayores rendimientos los cuales se observaron en la Figura 43.

3.4. EXTRACCIÓN DE MINUCIAS

3.4.1. Binarización

Convertir una imagen en escala de grises a una imagen binaria, es un proceso trivial, y para este caso se utilizó un umbral de 127, en la escala de 0 a 255.

Aunque en las figuras anteriores se presento las imágenes filtradas ya estaban binarizadas, cuando es aplicado el filtro de Gabor, sin normalizar, el resultado es una imagen donde las crestas y valles son apreciados como una imagen binaria, con algunas zonas en escala de grises, cuando se normaliza la función de Gabor, la imagen se ve rica en escala de grises y la transición entre crestas y valles se asemeja a una onda sinusoidal, para entender esto mejor observemos la Figura 45, donde se puede ver la imagen normalizada, no-normalizada y finalmente binarizada. Los tiempos de esta operación son extremadamente bajos de alrededor de 0,1 ms, porque se utilizan funciones optimizadas de OpenCV, esta es *cvThreshold()*.

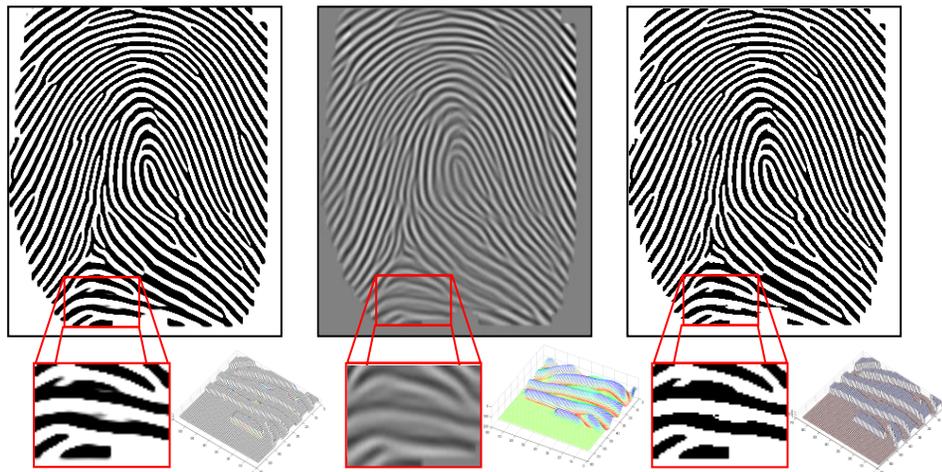


Figura 45. Binarización de la imagen.

3.4.2. Adelgazamiento

Originalmente se usó el algoritmo de ROS [31], el cual establecía que se debía detectar los bordes (*arriba, derecho, abajo e izquierdo*) en secuencia y eliminarlos, ello supone que se deba recorrer la imagen 4 veces para eliminar el primer borde de toda la imagen, para optimizar el proceso se puede modificar este proceso y se detectaron todos los bordes en un solo recorrido, sin embargo tal y como este algoritmo esta implementado ofrece buenos resultados y como el promedio del ancho de las crestas es pequeño, no toma mucho tiempo el adelgazar la imagen.

Se utilizó la conectividad 8 porque no aparecían muchas “ramas”, como si aparece cuando se utiliza conectividad 4, lo cual se ocasiona principalmente cuando una línea termina, en ella donde debería existir una sola terminación aparecen dos terminaciones y una bifurcación, la comparación de estos dos resultados se puede observar en la Figura 47 donde se adelgaza la huella filtrada, y en la Figura 48 cuando se detectan las minucias.

Solo para el caso de las huellas es útil la conectividad 8, porque para otro tipo de imágenes con esta conectividad las formas de la imagen desaparecen ó quedan con pocas características, que si las da con el uso de la conectividad 4, como se ve en la Figura 46, en ella se observa como el uso de la conectividad 8, desaparece al rombo desaparece, la gota y el círculo simplemente aparece una línea, y al sol le desaparecen muchas ramas comparado con el uso de la conectividad 4.

Las imágenes en la Figura 47 son: la imagen $A=7_6_{DB1_A}$, $B=17_4_{DB2_A}$, C y D son del autor con tamaños de 261×433 y 259×435 pixeles, los tiempos promedios de esta operación son de 35.16, 19.57, 24.60 y 24.62 ms, respectivamente.

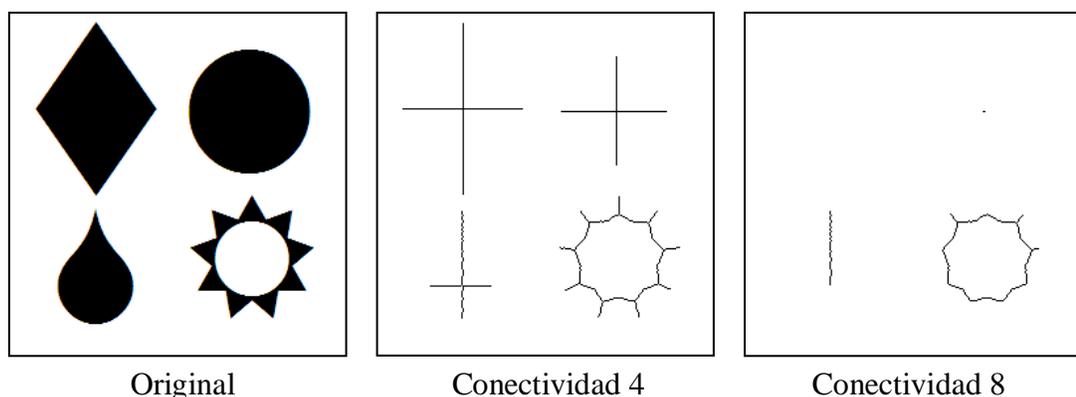


Figura 46. Adelgazamiento de figuras geométricas simples.

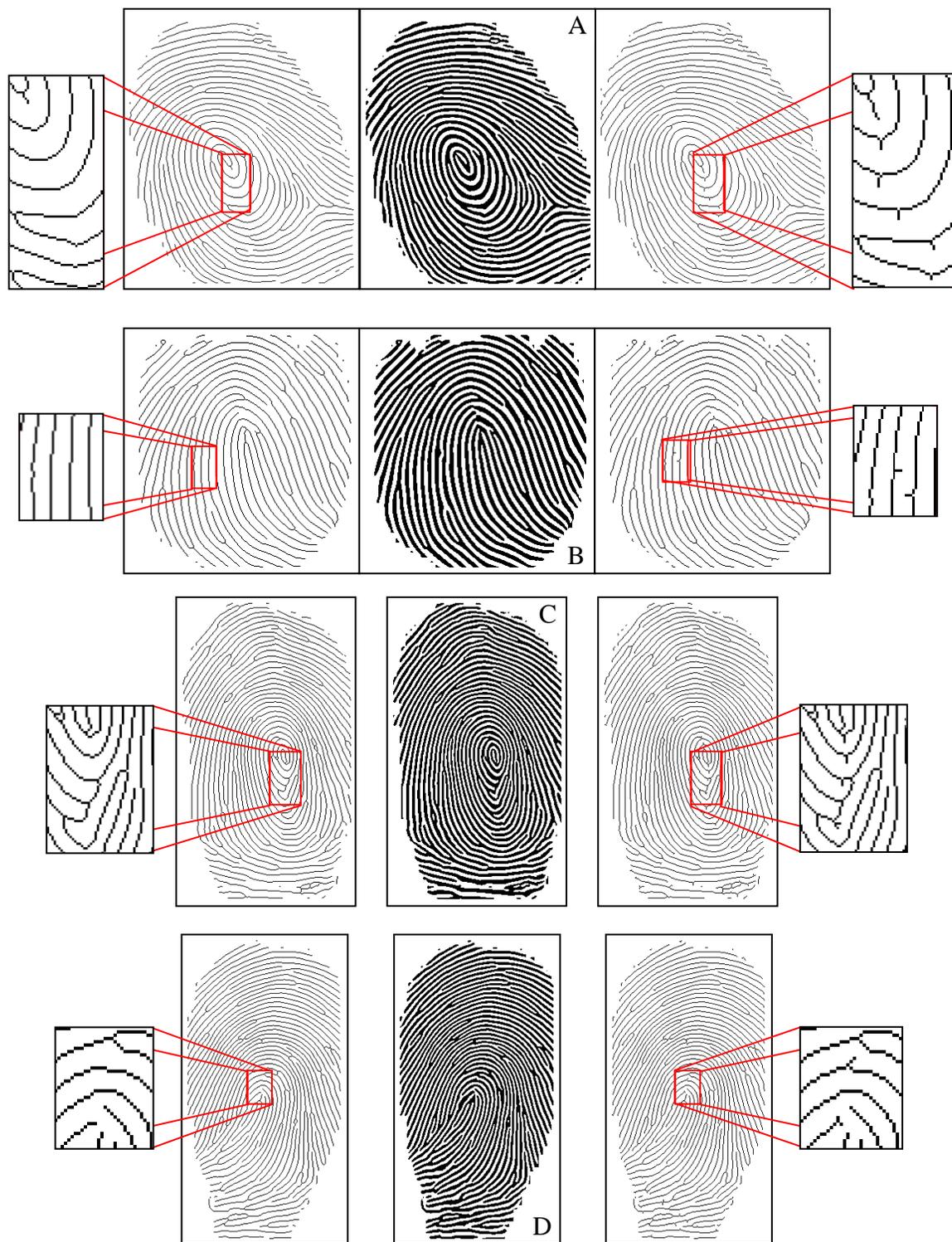


Figura 47. Imagen Adelgazada con conectividad 8 y 4.

3.4.3. Detección de Minucias

Para la detección de minucias se estudiaron todos los casos para encontrar terminaciones y bifurcaciones, como se explicó antes, todos los casos son evaluados en un solo recorrido a la imagen, y por lo tanto rápido. Sin embargo este algoritmo puede hacer sobre detección de minucias, que principalmente se ven en bifurcaciones unidas ó separadas por un pixel, y las terminaciones del borde de las crestas son detectadas también, aunque estas son minucias reales de la imagen adelgazada, no corresponden a la realidad y se corrigen en la siguiente sección.

Por ello solo se presentan los resultados de la detección de minucias, cuando se utiliza el adelgazamiento con conectividad 4 y 8, los puntos negros son bifurcaciones y terminaciones falsas, porque están muy cerca una de la otra y se superponen los colores, ello se ve en la Figura 48, donde A1, B1, C1 y D1 son adelgazadas con conectividad 8 y el resto con conectividad 4, aquí se guarda cada posición de la minucia con su respectivo valor de orientación, según sea el valor que exista en la matriz $\theta(i, j)$.

Las imágenes en la Figura 48 son: la imagen A=7_6_{DB1_A}, B=17_4_{DB2_A}, C y D son del autor con tamaños de 261x433 y 259x435 pixeles, los tiempos promedios de esta operación son de 1.42, 0.81, 1.31 y 1.22 ms, respectivamente.

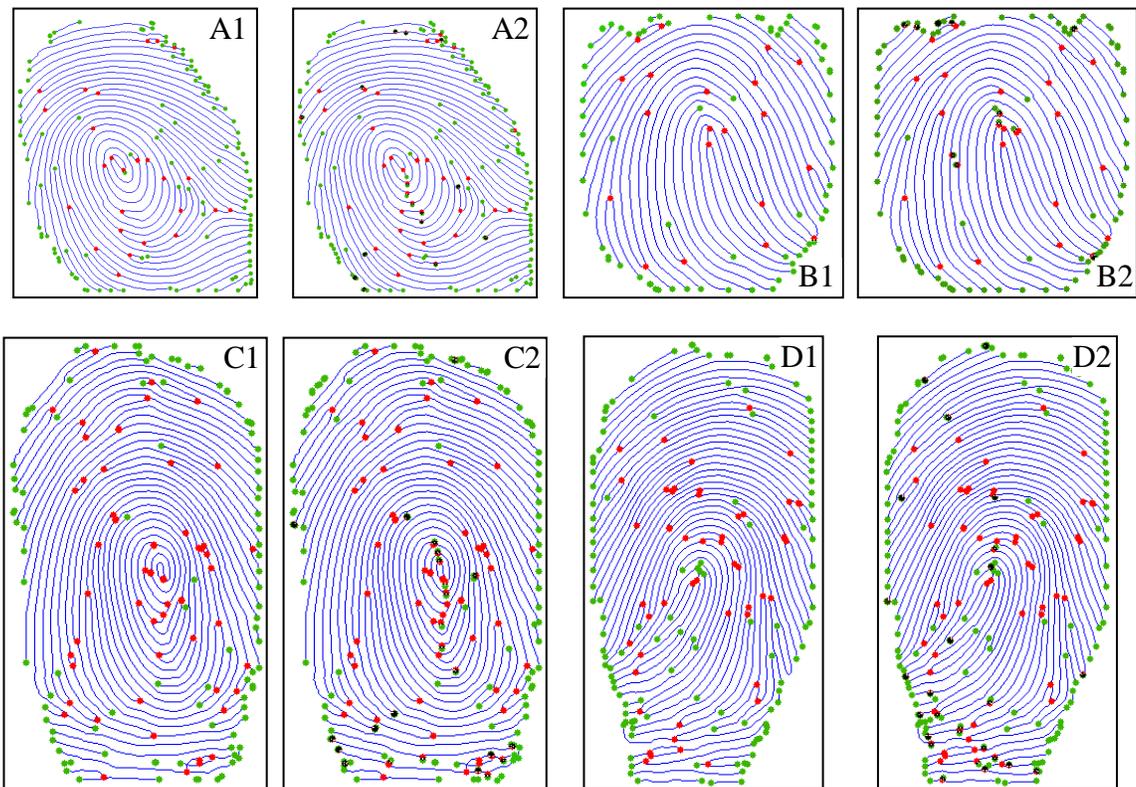


Figura 48. Extracción de minucias.

3.4.4. Eliminación de Falsas Minucias

Las minucias falsas que aparecen en la Figura 20 fueron implementadas, para las terminaciones que aparecen al final de cada cresta, se utiliza la máscara de segmentación, entonces cada terminación es encerrada en un cuadro de 10x10 píxeles, si en el cuadro está presente la máscara (*color negro*), la terminación se elimina.

Para las minucias internas se toma la misma idea, para cada terminación y bifurcación se encierra en un cuadro de 14x14 píxeles, si dos terminaciones aparecen en un mismo cuadro, se debería analizar las condiciones del ángulo expuestas en el apartado 2.3.3.6, sin embargo no se hizo, porque no fue necesario, en general para este ancho las minucias muy cercanas se eliminan, esto implica que en la zona del núcleo se eliminan más minucias que en otras partes, no obstante se ha visto que esta parte es la más difícil de reconstruir, por lo cual hallar minucias allí será también difícil.

De la misma forma se trabaja cuando hay una terminación y una bifurcación, en el caso que dentro del cuadro existan más de dos minucias, se analizan los puntos más cercanos. El último caso se presenta cuando dos bifurcaciones o más están en el mismo cuadro, si dos de bifurcaciones son mayores a dos píxeles se eliminan ambas minucias, de lo contrario se elimina una, y se deja la otra para maximizar las opciones de que se una con otra minucia.

Las imágenes en la Figura 49 son: la imagen A=7_6_{DB1_A}, B=17_4_{DB2_A}, C y D son del autor con tamaños de 261x433 y 259x435 píxeles, los tiempos promedio de esta operación son de 11, 6, 8 y 8 ms, respectivamente.

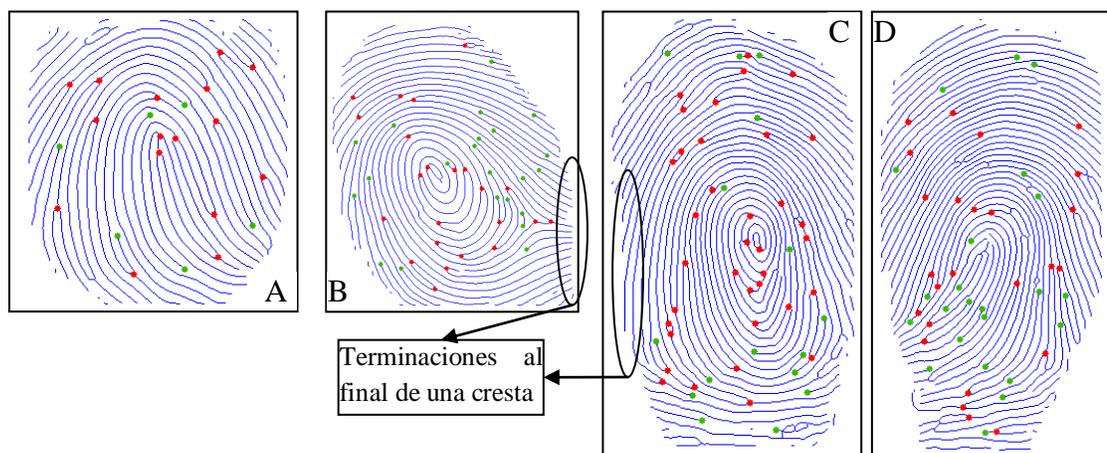


Figura 49. Eliminación de falsas minucias.

El cuadro de análisis de 14x14 es adecuado para no eliminar la mayoría de las falsas minucias, se probó con cuadros más grandes y la sobre-eliminación se ve incrementada, esto se puede ver en la Figura 50, donde A es la imagen con todas las minucias detectadas, para el resto B, C y D se eliminan las minucias, usando un tamaño de 14x14, 16x16 y

20x20 píxeles, respectivamente, se puede notar en los círculos encerrados en rojo, como la imagen D a perdido algunas bifurcaciones y terminaciones comparado con la imagen B.

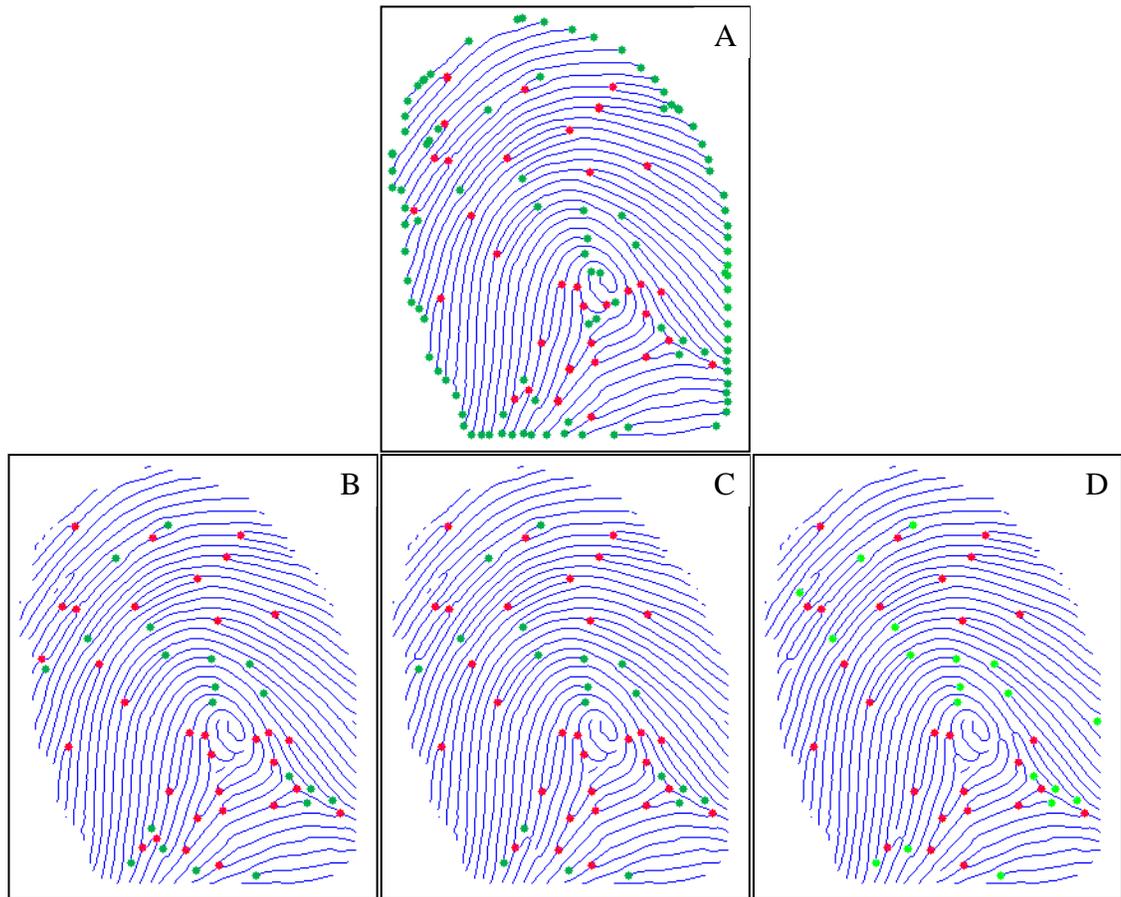


Figura 50. Eliminación de minucias variando el tamaño del cuadro.

3.4.5. Tiempo total de inscripción de la huella (*enroll*)

El tiempo total no es necesariamente la suma de los tiempos independientes, ya que los recursos de memoria asignados a la ejecución pueden variar, además de procesos internos que no se mencionan afectan en general el tiempo. Para ello se debe ejecutar todas las etapas mencionadas antes, es decir la normalización, la estimación de la frecuencia, el filtrado, la binarización, el adelgazamiento, la extracción y eliminación de falsas minucias.

Se evalúa los tiempos de la base de datos de la FVC2004 DB1_A, se toman 32 imágenes desde la 1_1.tif hasta la 4_8.tif y se promedia los tiempos de ejecución, dando un valor entre ~400 y ~500 ms/imagen, dependiendo del tamaño de la imagen. Revisando todos los tiempos de inscripción de la FVC (algunos mostrados en la Tabla 4) en la categoría *open*,

se observa que el tiempo de inscripción de nuestro algoritmo es muy similar a los de esta tabla, en los cuales se encuentran tiempos de hasta 7 segundos [10]. Inclusive en la tabla de medalleros de la FVC categoría open, se encuentra un algoritmo con un tiempo de inscripción de 2.07 segundos, es cual es 4 veces mayor que el tiempo nuestro, no obstante los algoritmos de la FVC se desarrollan a una menor frecuencia de procesador.

3.4.6. Conclusiones Parciales de la Extracción de Minucias

Respecto a la extracción y detección de minucias

En la etapa del adelgazamiento de la imagen se concluye, que el uso de adelgazamiento de la conectividad 8 produce pocas falsas minucias que el uso de conectividad 4, aunque este último es bueno para otras imágenes como por ejemplo círculos, triángulos, rombos, etc.

Las condiciones halladas para detectar las minucias son suficientes para detectar todas las minucias en una huella, aunque existen otras posibilidades en las bifurcaciones, no se forman cuando se utiliza la conectividad 8.

Un tamaño de bloque grande puede ocasionar un exceso de eliminación de minucias verdaderas, y uno pequeño lo contrario, por lo tanto el valor elegido para este propósito ofrece resultados que eliminan la mayoría de las falsas minucias y unas pocas de las verdaderas.

3.5. COMPARACIÓN DE LAS HUELLAS

Los resultados de la comparación solo involucran minucias, no se intento realizar la comparación por correlación porque el tiempo de procesamiento es muy alto, esto así lo sugiere la literatura [2, 30, 32, 33], por ello solo se estudian los resultados en minucias.

3.5.1. Comparación por Alineamiento Previo

La comparación de minucias por alineamiento previo, se realizó teniendo las huellas S y C alineadas previamente. Para ello las huellas se alinearon usando dos minucias iguales en cada huella, (las cuales debe elegir el usuario), tales como terminaciones, bifurcaciones, núcleos ó deltas.

Con ellas se puede establecer la traslación, rotación y escala entre las dos huellas, como se explico en la sección 2.3.4.3, con los parámetros establecidos se transforma las demás minucias y se comparan.

Debido a que la alineación es manual, es muy tedioso el trabajo de calcular la FAR y la FRR para una gran base de datos, sin embargo se probó el método en algunas huellas, y actualmente se utiliza como herramienta en el C.T.I., ofreciendo buenos resultados tanto para la base de datos del C.T.I. y la base de datos de la FVC2004 DB1_A.

Uno de los problemas que se presento es la determinación de los puntos en cada huella, ya que el usuario es quien los elige, se podía estimar mal el punto donde las minucias en cada huella son iguales, se determinó que esto ocurría por el tamaño de las crestas. En pruebas hechas se utilizó las imágenes de la FVC2004, las cuales tienen las siguientes características:

- Ancho entre crestas de 13 a 14 pixeles.
- Ancho de surco de 5 a 6 pixeles.

En estas imágenes la determinación de los puntos fue buena, porque las minucias se podían ver con buen detalle, estos parámetros entonces servirían de referencia para adecuar las imágenes del C.T.I.

Cuando se utilizó la base de datos del C.T.I. el tamaño podía variar, para una de sus bases de datos, los tamaños de las crestas en la imagen son de:

- Ancho entre crestas de 7 a 8 pixeles.
- Ancho de surco de 2 a 3 pixeles.

Con este tamaño los puntos no se podían determinar con exactitud, porque no se podían ver bien los detalles en la imagen, además se podía confundir el operario sobre cuál era la coordenada exacta de los puntos de cada huella.

Otra base de datos tipo de imágenes que manejan, son huellas tomadas con fotografía, las cuales por el gran tamaño de la imagen no se pueden procesar adecuadamente, ya que ellos utilizan entre 6 y 10 megapixeles para tomar la foto, el cual genera imágenes de 2000 a 3000 pixeles cuadrados, con lo cual las crestas tienen:

- Ancho entre crestas superiores de 100 pixeles.
- Ancho de surco de superiores a 40 pixeles.

Se presenta un problema similar al anterior donde no se puede precisar con exactitud las coordenadas de alineación, con la diferencia de que en este caso era por el tamaño excesivo de las imágenes, y por ende de las minucias, además una imagen tan grande no se podría procesar porque la frecuencia de las crestas no se podría determinar. De forma visual se puede ver esto en la Figura 51. Donde A es una imagen de la FVC2004 a 500 dpi, B es una imagen pequeña que pertenece a una base de datos del CTI y no se especifica la resolución, C es una imagen grande escaneada a 650 dpi, y D es la imagen C redimensionada a 500x500, en ella se puede notar cómo después del ajuste de la escala, la distancia en pixeles de las crestas es similar a las imágenes de la FCV2004, con ello se logró tener una condiciones optimas para el mejoramiento y posteriormente en su comparación.

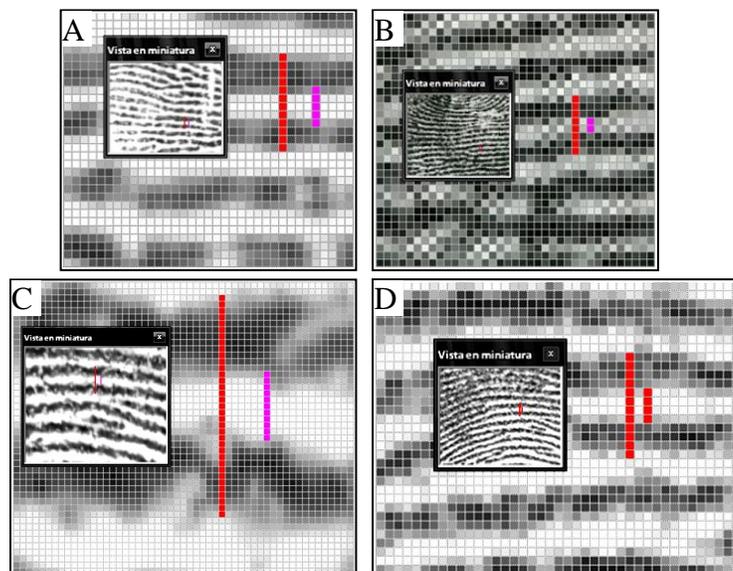


Figura 51. Huellas escaneadas a distintas resoluciones.

Solución para la comparación

Para solucionar los problemas mencionados, se optó por redimensionar la imagen, hasta obtener un tamaño de cresta similar a los de la base de datos de la FVC2004. Se encontró entonces que se debía redimensionar a un tamaño de 500x500 píxeles, pero sin cambiar las proporciones de la imagen, es decir si se tiene una imagen de 464x638 píxeles, entonces se toma el mayor valor entre el ancho y el largo, para este caso el de 638 y se cambia proporcionalmente las dimensiones a 364x500 píxeles, donde el tamaño se disminuye en un porcentaje del 78.3%. Con ello se obtuvo un ancho de cresta adecuado para la extracción de características, y el problema de encontrar los puntos para la alineación se hicieron más fáciles de hallar, por ende más exactos y precisos, un ejemplo puede ser visto en la Figura 51 en ella A es una imagen de la FVC, B es una imagen la cual está muy reducida en sus dimensiones y sus crestas están muy unidas, C es una imagen demasiado grande, típica de cámaras con alta resolución y sus crestas muy separadas, por último D es la imagen C redimensionada a 500x500 píxeles, en ella se puede ver como esto se asemeja al tamaño de cresta de A, las cuales se pueden contar por el número de píxeles rojos en la imagen.

El método propuesto por alineamiento previo, es escogido para el análisis de imágenes en el C.T.I. porque ofrece un buen tiempo de procesamiento y se ajusta a los tipos de huellas que manejan, además en huellas de mala calidad encontrar las minucias reales es muy difícil [2, 17, 32], ya que el ruido de la imagen mezcla las falsas y verdadera minucias, las cuales son ocultas por la máscara de confiabilidad, por ello es de esperar que los algoritmos de alineación automática puedan alinearlas incorrectamente, y establecer igualdades entre minucias que no son.

Sin embargo los dactiloscopistas necesitan tener la total certeza de que esa minucia es real, y que la comparación corresponda, por ello cuando el usuario alinea la huella, es mucho más confiable ya que él puede determinar qué puntos son iguales en cada huella, y no solo se limita a las terminaciones y bifurcaciones, también hace uso del núcleo y el delta.

Es por ello que se elige el método del alineamiento previo, para implementarlo en un software amigable al usuario. Un breve resumen de este se encuentra en el apéndice A, donde también encontramos algunas ejemplos de huellas comparadas (ver Figura 58).

El pseudocódigo para el alineamiento y la comparación se presentan a continuación, primero la comparación:

3.5.1.1. Tiempo total de Comparación por alineamiento previo

Al igual que en la sección 3.4.5 se toman las mismas imágenes de prueba y con las consideraciones internas de los procesos, para las 32 imágenes el tiempo promedio es: ~2ms/imagen, es realmente corto el tiempo ya que no se deben estimar ningunos parámetros de alineación de huellas, y solo se comparan las posiciones de las minucias.

3.5.2. Comparación por Grafos

La comparación por grafos utilizada fue propuesta en [15], se evaluó bajo las mismas circunstancias, y se toma solo la parte de comparación de las minucias, adaptando esto al algoritmo del trabajo.

Este algoritmo no ajusta la escala de las huellas, pero si se utiliza un lector de huella ó se escanean a la misma resolución, todas las imágenes van a tener la misma escala, y por lo tanto se podrán comparar adecuadamente.

Resultados de la base de datos DB1_A de la FVC2004

Para la evaluación del algoritmo se utilizo la base de datos de la FVC2004 DB1_A, que contiene 800 huellas, que consta de 8 huellas por dedo de 100 personas (un dedo por persona), tomadas por un sensor óptico (CrossMatch V300), el tamaño de la imagen es de 640×480 y la resolución es de 500 dpi.

Para estimar la tasa de falso rechazo (FAR), se compara la primera imagen de cada dedo contra la primera imagen de todos los otros dedos (100 dedos), pero evitando los partidos simétricos; es decir si una huella j es comparada contra una huella k , después la huella k no puede ser comparado con k . La cantidad de intentos se puede calcular con la formula de

Gauss³ de sumas sucesivas de números, para un total de 2800 comparaciones de impostores. Entonces usando la ecuación (1) la $FAR = N_{FA}/2800$.

Para calcular la tasa de falsa aceptación (FRR), la primera imagen de cada dedo es comparada entre las imágenes de un mismo dedo, para este caso son 8 imágenes por dedo de 100 dedos diferentes, también se debe evitar los partidos simétricos. Luego la cantidad de intentos son 4950 comparaciones de genuinos. Entonces usando la ecuación (2) la $FRR = N_{FR}/4950$.

Con el fin de poder comparar el algoritmo desarrollado en este trabajo, con otros algoritmos, se estimó la tasa de error medio (EER) a partir de la FAR y FRR, para ello se utiliza un puntaje de comparación basado en la ecuación (24), donde 1 significa que las huellas son iguales y 0 que son desiguales. Para la construcción de las curvas de FAR y FRR se tomaron 20 umbrales, desde 0 hasta 1 en pasos de 0.05, con lo cual se construye la Figura 52.

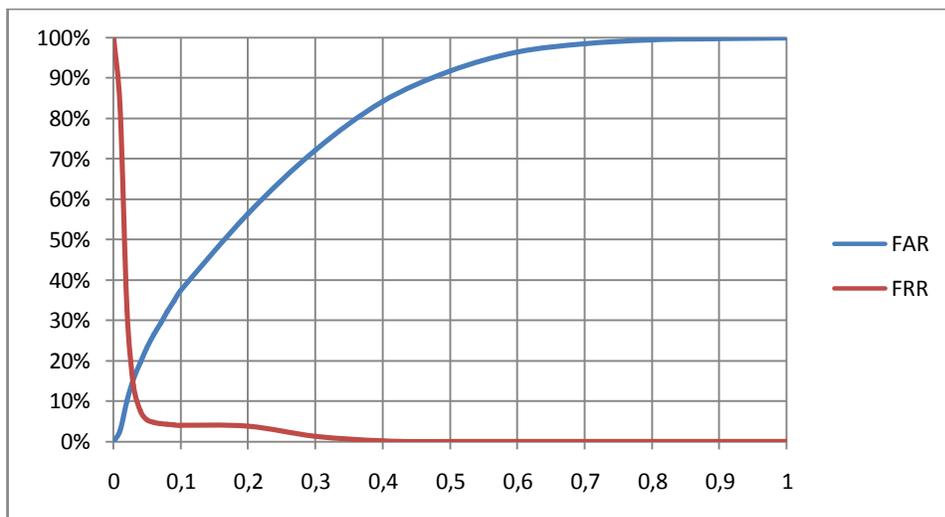


Figura 52. Resultados de la FAR, FRR para la DB1_A de la FVC2004.

Como se puede ver en la Figura 52, las curvas de la FAR y FRR producen un ERR del 15%, es un error medio muy grande, que se debe a que en general las huellas de la FVC2004 son difíciles de trabajar, porque se le pidió a los individuos que prestaron sus huellas que: *colocaron el dedo en posición vertical, aplicando altas y bajas presiones sobre el sensor, exagerar la distorsión de la piel y rotación, tener los dedos secos y húmedos. Además no se tuvo cuidado en la toma de huellas y no se limpiaba el sensor, con el fin de asegurar un mínimo de calidad en todas las imágenes. Todo esto con el propósito de poner a prueba máxima los algoritmos* [38].

³ La fórmula de Gauss para la sumatoria de números enteros sucesivos como $1+2+3+\dots+100 = 5050$ fue descubierta por él mismo a la edad de 10 años.

Para comparar los resultados del trabajo, con los de la FVC2004, observemos primero el ERR para la base de datos DB1_A en la Tabla 5, categoría light, organizados de mayor a menor ERR. Comparando estos errores con los estimados en el trabajo, se puede decir que el algoritmo tiene un mediano desempeño, y se podría clasificar en la posición del algoritmo P027, el cual ocupó la posición 17 entre 26 algoritmos que se presentaron a esta prueba, y está por encima de otros algoritmos de industrias y academias a nivel mundial.

Tabla 5. Resultados de la FVC2004 para la DB1_A.

Algoritmo	ERR (%)	Organización
P009	3.89	Suprema Inc (industria)
P065	10.26	ActivCard (industria)
P027	15.85	Zaklad Techniki Mikroprocesorowej (industria)
P105	18.03	Li Lijuan (desarrollador independiente)
P087	34.80	Anónimo (industria)
P006	37.97	Anónimo (industria)
P099	39.03	Anónimo (academia)
P050	50.00	Anónimo (academia)

Algunas de las imágenes donde las características son difíciles de determinar, muestran en la Figura 53, como se ve en ellas, su calidad es pésima, la primera huella tiene sub-entintado y las 3 últimas con sobre-entintado. Estas imágenes son difíciles porque la extracción de características no fue adecuada, debido a que no se filtro adecuadamente la huella. Por ello se debe mejorar la etapa del filtrado, porque la extracción de características y comparación son buenas.



Figura 53. Imágenes de la DB1_A de mala calidad.

Resultados de la base de datos del autor

También se realizó el procedimiento anterior para la base de datos del autor, y se dividió en dos partes, una la cual tiene huellas de mediana calidad (80 imágenes) y la otra con el total de las imágenes de mediana y baja calidad (160 imágenes), las cuales se nombraremos Al_1 y Al_2, respectivamente.

Para el cálculo de la FAR en Al_1, se toma el procedimiento anterior, donde se compara la primera imagen de cada dedo contra la primera imagen de todos los otros dedos (10 dedos), pero evitando los partidos simétricos. La cantidad de intentos es de 280 comparaciones de impostores. Entonces usando la ecuación (1) la $FAR = N_{FA}/280$.

De igual forma para Al_2, el número de dedos es 20, y la cantidad de comparaciones es de 560 de impostores, entonces la $FAR = N_{FA}/560$.

Para calcular la FRR en Al_1, la primera imagen de cada dedo es comparada entre las imágenes de un mismo dedo, para este caso son 8 imágenes por dedo de 10 dedos diferentes, también se debe evitar los partidos simétricos. Entonces la cantidad de intentos es de 45 comparaciones de genuinos. Entonces usando la ecuación (2) la $FRR = N_{FR}/45$.

De igual forma para Al_2, con el número de dedos de 20, y la cantidad de comparaciones es de 190 genuinos, entonces la $FRR = N_{FR}/190$.

El resultado de la FRR y FAR para Al_1 se puede ver en la Figura 54, y para Al_2 en Figura 55.

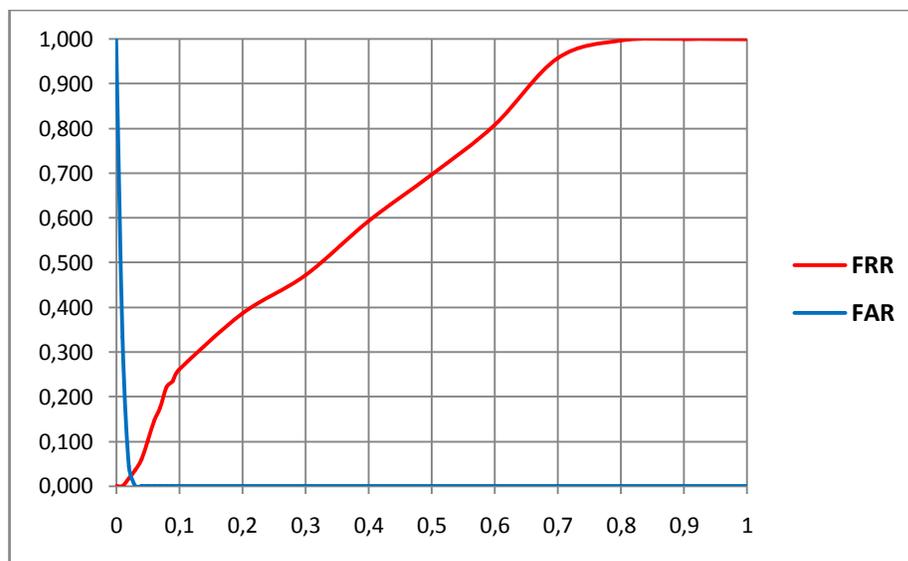


Figura 54. Resultados de la FAR, FRR con ERR = 2.3%.

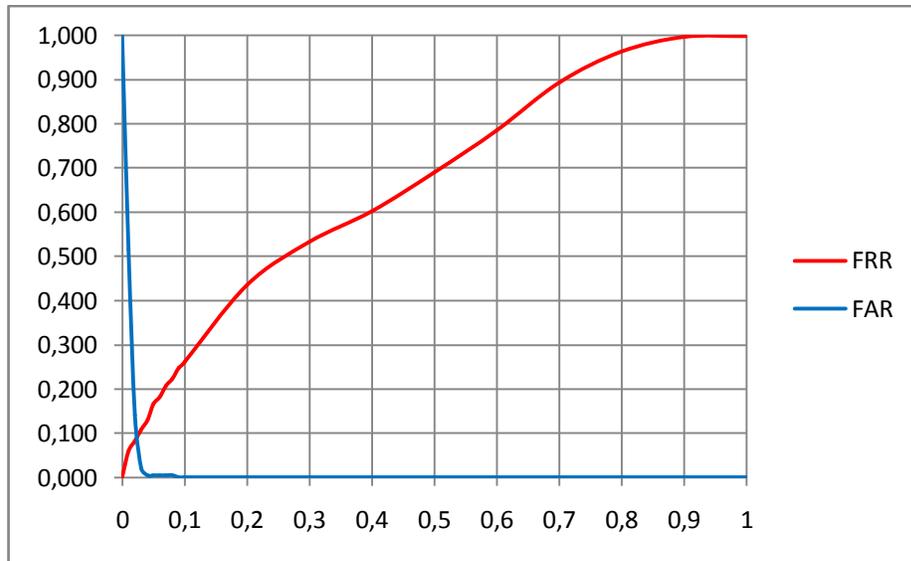


Figura 55. Resultados de la FAR, FRR con ERR = 9.7%.

Como se ve en las figuras anteriores, la buena calidad de las huellas es algo indispensable para obtener buenos rendimientos en los sistemas.

El error en AI_1 se debe a que las huellas del dedo meñique son difíciles de trabajar porque las crestas no se ven muy bien definidas, y se presenta mucho desgaste. En AI_2 el error se debe a que no se tuvo cuidado a la hora de tomar las huellas, además se tomó en personas que presentaban desgaste por contacto excesivo con jabón, agua, etc. Algunas de estas imágenes de mala calidad se ven en Figura 56, las cuales son imágenes que presentan mediana y mala calidad. De izquierda a derecha, dedo de una persona que lava mucho con jabón y cocinan que además se adquirió sin ninguna precaución, y los dos últimos son dedos meñiques que en general tienen mala formación de las crestas.



Figura 56. Imágenes con mediana y mala calidad.

3.5.2.1. Tiempo total de Comparación por Grafos

Al igual que en la sección 3.4.5 se toman las mismas imágenes y con las consideraciones internas de los procesos, para las 32 imágenes el tiempo promedio es ~ 276 ms/imagen.

El tiempo promedio de comparación de un par de huellas es mucho mayor que el anterior, porque debe establecer la traslación y rotación a partir de todas las minucias. Además este puede aumentar o disminuir según sean el número de minucias.

3.5.3. Conclusiones Parciales de la Comparación

Respecto a la comparación por alineamiento previo

La comparación es exitosa siempre y cuando las minucias elegidas para el alineamiento previo sean adecuadas, por lo cual se concluye que un tamaño de por lo menos 500x500 pixeles sin cambiar las proporciones de la imagen, es suficiente para asemejar a las características de las huellas de la FVC. También se concluye que es un método que brinda más certeza cuando se comparan las minucias, ya que el alineamiento previo es mucho mejor que un automático, para el caso de huellas de mala calidad.

Respecto a la comparación por Grafos

Se logró obtener un buen rendimiento del sistema cuando las huellas son adquiridas con buena calidad, como es el caso de la Figura 54 donde el error es bastante bajo, donde se determino que las huellas del dedo meñique de una persona, son las que causaban este error, porque estas tienen una tendencia natural a tener mala calidad, esta conclusión es soportada por los dactiloscopistas expertos en la materia.

Cuando se mezcló las huellas de buena calidad con las de mala calidad, el error determinado aumenta considerablemente, lo cual es de esperar, debido a las malas adquisiciones que se hicieron, por ello se concluye que el factor más importante para el reconocimiento de huellas es la calidad de la imagen.

Respecto a los tiempos

Comparando el tiempo total de inscripción de la huella con los algoritmos de la FVC2004, se concluye que el tiempo es adecuado, no obstante se puede mejorar mucho más optimizando los procesos de filtrado, el cual es la etapa menos optimizada. Sin embargo el tiempo de comparación se debe mejorar mucho más, para poder utilizarlo en un sistema AFIS, ya que el tiempo de ~276 ms/imagen es bastante grande, y en estos software se necesitan tiempos menores a 1 ms.

Capítulo 4

4. CONCLUSIONES

El algoritmo para la verificación de huellas, se divide básicamente en dos etapas, una en la cual se inscribe la huella, y la otra cuando se comparan.

En el primer caso, un logro importante es la flexibilidad del algoritmo para adaptarse a los distintos anchos de cresta, esto se debe al cálculo de la frecuencia local, que permite obtener buenas reconstrucciones a resoluciones entre 500 y 400 dpi, así las configuraciones de captura pueden variar entre estos parámetros.

Los parámetros de afinación del filtro de Gabor dados por Kovési[21] y Thai[11], aplicados al mejoramiento de la imagen ofrecen buenos resultados, comparado con otras técnicas de mejoramiento como los filtros de media direccional, el aniso-trópico y el de Gabor a una frecuencia. Además se logró buena estimación de la orientación y la frecuencia local necesaria para aplicar el filtro de Gabor.

Se determinó que variar el tamaño del bloque para estimar la frecuencia no afecta en gran medida la reconstrucción, se observó que las minucias sólo cambian de terminación a bifurcación, y viceversa; sin embargo se encontró que aumentar el suavizado en la orientación, afecta desfavorablemente la reconstrucción, principalmente en huellas con uno o más núcleos.

El segundo paso importante es el cotejo, realizado por dos métodos, donde se encontró que la comparación por grafos es más apropiada, ya que realiza una alineación automática de las minucias, aunque el tiempo de comparación es mayor que por alineamiento previo.

No se realizó una comparación por correlación, ya que esta técnica es muy lenta cuando se implementa en un algoritmo, por ahora la mejor forma de usar este método es por correlación óptica [33].

Se logró un error medio (EER) aceptable del sistema, comparado con los resultados de la FVC2004, donde se determinó que la principal desventaja del sistema ésta en la etapa de inscripción y no en la comparación.

En general todo el proceso de inscripción y comparación de la huella es rápido, el tiempo total por el método de grafos, es menor a 1 segundo para un par de huellas, y usando el de alineamiento previo, depende de la habilidad del dactiloscopista para ingresar los parámetros iniciales, en general este tiempo es de alrededor de 10 segundos, mucho menor comparado con el cotejo manual de 5 a 15 minutos.

4.1. Trabajos Futuros

Utilizar sistemas de multi-resolución para disminuir el tiempo de inscripción de las huellas.

Probar el filtrado de la imagen utilizando el espacio de las frecuencias, el cual se ha demostrado puede ser más eficiente que en el dominio espacial.

Disminuir el tiempo de comparación de las huellas; para así realizar el Sistema Automático de Identificación de Huellas (AFIS), con el se puede ajustar para las distintas aplicaciones civiles, identificación criminal y el control de acceso.

REFERENCIAS

1. Maltoni, D. and R. Cappelli, *Fingerprint Recognition*, in *Handbook of Biometrics*, A.K. Jain, P. Flynn, and A.A. Ross, Editors. 2008, Springer.
2. Maltoni, D., et al., *Handbook of Fingerprint Recognition* ed. Springer-Verlag. 2003, New York.
3. Cole, S.A., *History of Fingerprint Pattern Recognition*, in *Automatic Fingerprint Recognition Systems*, N. Ratha and R. Bolle, Editors. 2004, Springer: New York.
4. Cappelli, R., et al., *FVC2006: the Fourth International Fingerprint Verification Competition*. *Biometric Technology Today*, 2007. **15**(7-8): p. 7-9.
5. Jain, A.K. and A. Ross, *Introduction to Biometrics*, in *Handbook of Biometrics*, A.K. Jain, P. Flynn, and A.A. Ross, Editors. 2008, Springer.
6. Group, I.B. *Biometrics Market and Industry Report 2009-2014*. 2009 [cited 2009 2 octubre]; Available from: http://www.biometricgroup.com/reports/public/market_report.php.
7. Prabhakar, S., *Fingerprint classification and matching using a filterbank*. 2001, Michigan State University: East Lansing.
8. Berry, J. and D.A. Stoney, *History and Development of Fingerprinting*, in *Advances in Fingerprint Technology* H.C. Lee and R.E. Gaensslen, Editors. 2001, CRC series in forensic and police science.
9. Hong, L., *Automatic Personal Identification Using Fingerprints*, in *Computer Science*. 1998, Michigan State University.
10. Maio, D., et al. *FVC2004: the Third International Fingerprint Verification Competition*. 2004 [cited 2009 1/10]; Available from: <http://bias.csr.unibo.it/fvc2004/download.asp>.
11. Thai, R., *Fingerprint Image Enhancement and Minutiae Extraction*. 2003, The University of Western Australia.
12. Wu, C., *Advanced Feature Extraction Algorithms For Automatic Fingerprint Recognition Systems*. 2007, University of New York at Buffalo.
13. Bo, J., T.H. Ping, and X.M. Lan, *Fingerprint Singular Point Detection Algorithm by Poincaré Index* WSEAS TRANSACTIONS on SYSTEMS, 2008. **7**(12).
14. Gonzalez, R.C. and R.E. Woods, *Digital Image Processing*. 3rd ed. 2007: Prentice-Hall, .
15. Chikkerur, S.S., *Online Fingerprint Verification System*, in *Department of Electrical Engineering*. 2005, State University of New York at Buffalo.
16. Wang, Y., J. Hu, and F. Han, *Enhanced gradient-based algorithm for the estimation of fingerprint orientation fields*. *Applied Mathematics and Computation*, 2007. **185**(2): p. 823-833
17. Ross, A., J. Reisman, and A. Jain, *Fingerprint Matching Using Feature Space Correlation*. *Workshop on Biometric Authentication*, 2002(2359): p. 48 - 57.
18. Ross, A.A., *Information Fusion in Fingerprint Authentication*, in *Department of Computer Science & Engineering*. 2003, Michigan State University.

19. Duque, H.D.C. and J.A.Q. Sepúlveda, *Adquisición y Procesamiento de Huellas Dactilares Para Verificación de Identidad*, in *Departamento de Electricidad, Electrónica y Computación*. 2004, Universidad Nacional de Colombia: Manizales.
20. Hong, L., Y. Wan, and A. Jain, *Fingerprint image enhancement: algorithm and performance evaluation*. *Pattern Analysis and Machine Intelligence*, IEEE Transactions on, 1998. **20**(8): p. 777-789.
21. Kovesi, P. *Fingerprint Enhancement*. 2005 [cited 2009/ 28/ 09]; Available from: <http://www.csse.uwa.edu.au/~pk/Research/MatlabFns/index.html>.
22. Lin, H., *Fingerprint Enhancement*, in *Proceedings of the 3rd IEEE Workshop on Applications of Computer Vision (WACV '96)*. 1996, IEEE Computer Society. p. 202.
23. Afsar, F.A., M. Arif, and M. Hussain, *Fingerprint Identification and Verification System using Minutiae Matching*, in *National Conference on Emerging Technologies*. 2004.
24. Cheng, Y.J. and P.D. Sun, *Fingerprint Feature Extraction based on Invariant Moments and Gabor Filters* *Complex Systems and Applications-Modeling, Control and Simulations*, 2007. **14**(2): p. 1441-1444.
25. Zhixin, S. and G. Venu, *A chaincode based scheme for fingerprint feature extraction*. *Pattern Recogn. Lett.*, 2006. **27**(5): p. 462-468.
26. Wu, C., Z. Shi, and V. Govindaraju. *Fingerprint image enhancement method using directional median filter*. 2004: SPIE.
27. Blotta, E. and E. Moler, *Fingerprint image enhancement by differential hysteresis processing*. *Forensic Science International*, 2004. **141**(2): p. 109-113.
28. Liu, L. and T.-S. Dai, *Ridge Orientation Estimation and Verification Algorithm for Fingerprint Enhancement*. *Journal of Universal Computer Science*, 2006. **12**(10): p. 1426--1438.
29. Chikkerur, S., A.N. Cartwright, and V. Govindaraju, *Fingerprint enhancement using STFT analysis*. *Pattern Recognition*, 2007. **40**(1): p. 198-211.
30. Ito, K., et al., *A Fingerprint Recognition Algorithm Using Phase-Based Image Matching for Low-Quality Fingerprints*, in *International Conference on Image Processing*. 2005, IEEE Computer Society. p. II- 33-6.
31. Hall, R.W., *Topological Algorithms for Digital Image Processing*, in *Parallel Connectivity-Preserving Thinning Algorithms*, T.Y.K.a.A. Rosenfeld, Editor. 1996, Elsevier Science: Netherlands.
32. Jianxin, Z., O. Zongying, and W. Honglei, *Fingerprint Matching Using Phase-Only Correlation and Fourier-Mellin Transforms*, in *Proceedings of the Sixth International Conference on Intelligent Systems Design and Applications - Volume 02*. 2006, IEEE Computer Society.
33. Tíjaro, O.J., *Desarrollo De Un Prototipo Optoelectrónico Para Evaluación En La Verificación De Huellas Dactilares*, in *Facultad de Ingenierías Fisico-Mecánicas*. 2008, Universidad Industrial de Santander: Bucaramanga.
34. Cheng, Y.J. and P.D. Sun, *Fingerprint Verification Based on Invariant Moment Features and Nonlinear BPNN* *International Journal of Control, Automation, and Systems*, 2008. **6**(6): p. 800-808.
35. Marana, A.N. and A.K. Jain, *Ridge-Based Fingerprint Matching Using Hough Transform*. 2005.

36. Tsai-Yang, J., *Minutiae-based partial fingerprint recognition*. 2005, State University of New York at Buffalo. p. 169.
37. Nalini, K.R., et al., *A Real-Time Matching System for Large Fingerprint Databases*. IEEE Trans. Pattern Anal. Mach. Intell., 1996. **18**(8): p. 799-813.
38. Alonso-Fernandez, F., et al., *Fingerprint Recognition*, in *Guide to Biometric Reference Systems and Performance Evaluation*, D.P. Delacrétaz, G. Chollet, and B. Dorizzi, Editors. 2008, Springer.

Anexo A

Software De Comparación

El software creado para el cotejo de huellas en el C.T.I., se denomina *Manipulador de Impresiones Dactilares (M.I.D.)*. En este se implementan los algoritmos descritos antes, las funciones principales del MID se adecuaron a los requerimientos de los lofoscopistas del C.T.I, los cuales son:

- Eliminar el ruido de la huella.
- Presentar los resultados de las minucias comparadas.
- Manipular las minucias (adicionar y borrar minucias).
- Guardar las imágenes de la comparación.

Teniendo en cuenta esto se realizó la interface grafica que se ve en la Figura 57, está dividido en 3 pestañas (*Análisis, Comparación, Visualización*) y una barra de menú (*para acceso rápido a las funciones principales*), cuando se despliega el programa la mayoría de las funciones están desactivadas, estas se activan y desactivan conforme se interactué con él.

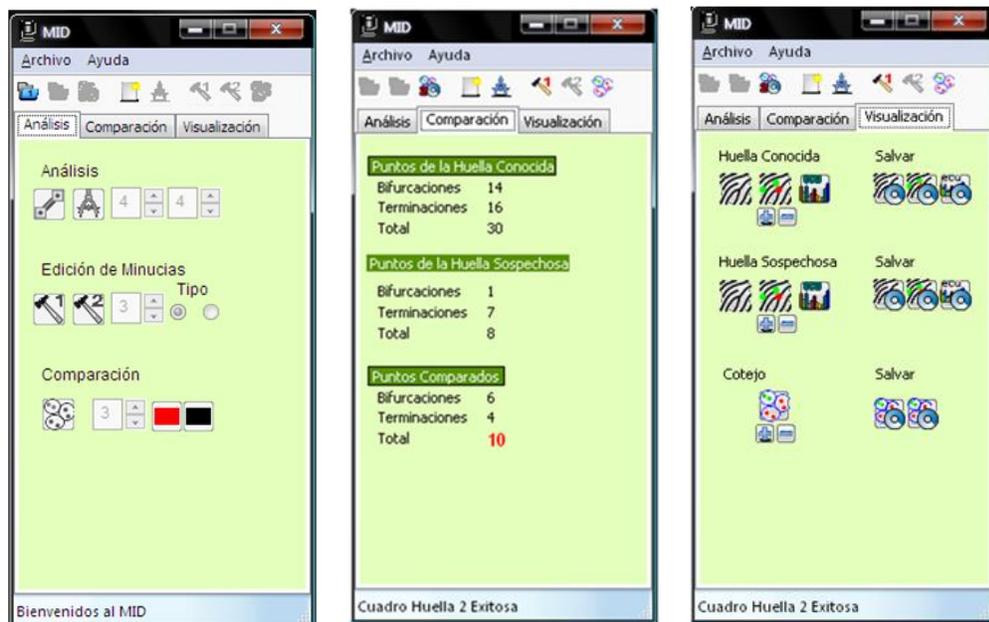


Figura 57. Interface grafica del MID.

Pestaña análisis: contiene botones para realizar el cotejo de un par de huellas, ó la simple extracción de minucias de una huella, se puede ajustar (*translación, rotación y escala*) para que coincidan las dos huellas, el nivel de suavizado de las orientaciones se puede variar, permite adicionar o eliminar minucias, y finalmente la comparación la cual etiqueta cada minucia hallada e indica con una línea donde se encuentra, ver la Figura 58, el programa está limitado a mostrar 20 minucias, ya que con 10 es suficiente para asegurar que las dos huellas son iguales.

Pestaña comparación: muestra el resultado de la extracción de minucias de las huellas comparadas, indicando el número de terminaciones y bifurcaciones, para cada huella (*conocida y sospechosa*) y cuáles de ellas son iguales.

Pestaña Visualización: estos botones permiten desplegar o guardar la imagen, se puede ver la huella limpia de ruido (*la imagen filtrada por Gabor*), la huella con sus minucias, y la huella ecualizada, se puede salvar cada una de ellas en formatos bmp, png y jpg.

Adicional a esto se entrega con un tutorial de ayuda, donde se explica el funcionamiento del programa cuando se utiliza para una comparación de huellas, o la mejora de una huella.

Para medir el tiempo de cotejo manual y el tiempo realizándolo mediante el MID, se tomaron 3 pares de huellas, el objetivo final era determinar las minucias iguales y hacer el soporte grafico.

Cuando se realiza esto de forma manual, el tiempo promedio de comparación fue de 40 minutos, ellos utilizan la lupa galtoniana para amplificar la huella y encontrar los puntos característicos, lo cual puede tomar unos 10 a 20 minutos dependiendo de la calidad de las huellas, después se digitalizan las huellas con ayuda de una cámara de alta resolución, con estas imágenes utilizan word o photoshop para poner una línea y un numero de donde se encuentran las minucias, para entregar el soporte grafico, lo cual puede tomar alrededor de 30 minutos.

Cuando se utiliza el MID el tiempo promedio es de 7 minutos, porque se debe digitalizar las huellas con un escáner, se deben buscar las dos minucias para el ajuste, y después se hace el procesamiento, análisis y comparación de las huellas lo cual toma alrededor de 2 segundos, entregando el soporte grafico como en la Figura 58.

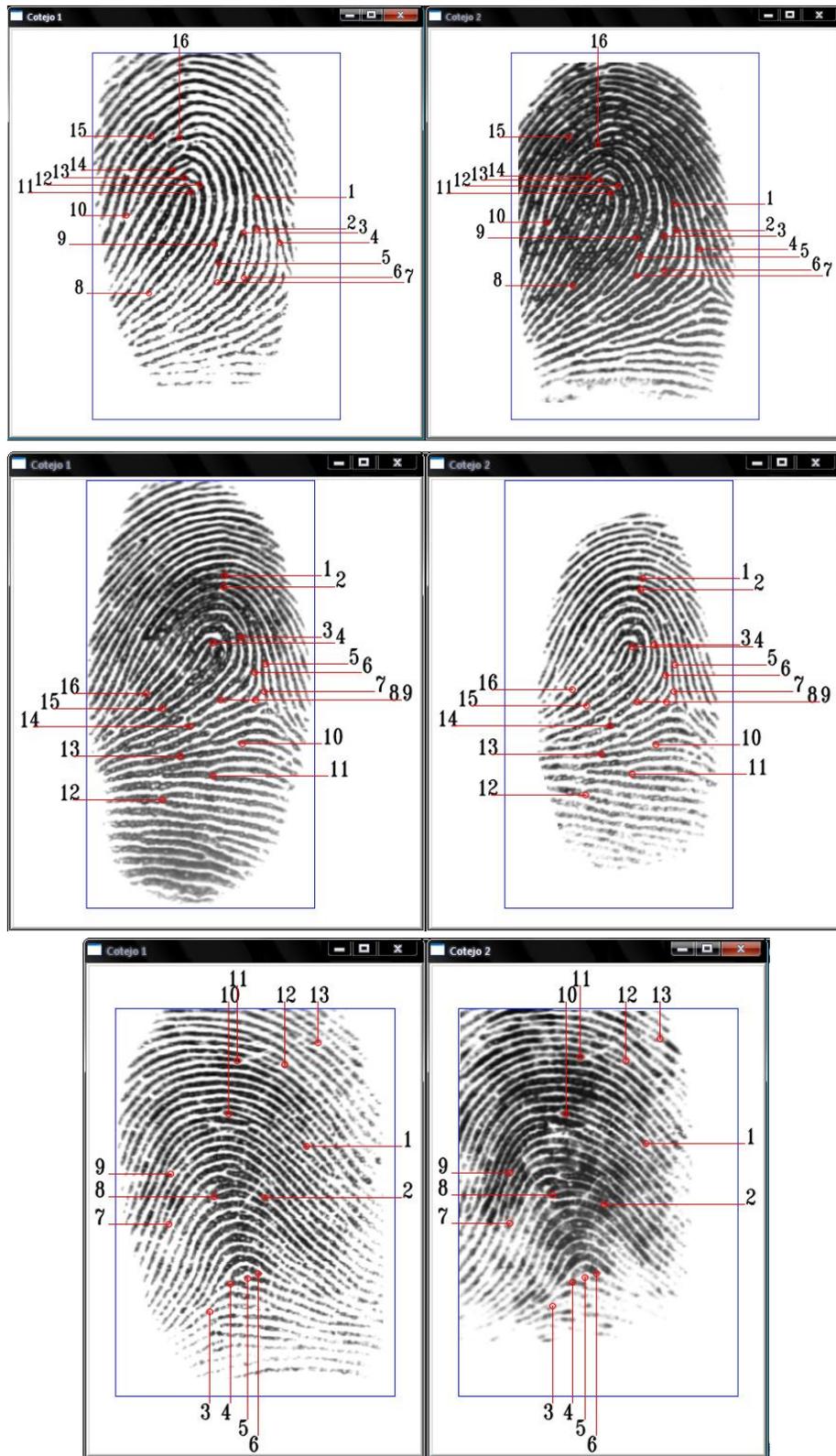


Figura 58. Despliegue del MID para una comparación de dos huellas.

El código fuente del programa se presento en forma de pseudocódigo a lo largo del trabajo, pero para mayor claridad se deja el código completo del programa, el cual contiene la interface grafica del programa, que debido a su exención se copió en el CD con el que se entrega la tesis, la compilación del software fue realizada en wxDevC++ versión 7.0 RC5 build 7.0.0.76, el cual puede ser descargado de internet gratuitamente en:

http://sourceforge.net/projects/wxdsgn/files/wxDev-C%2B%2B/Version%207/wxdevcpp_7.3.0_full_setup.exe/download.

O buscar su última versión en <http://wxdsgn.sourceforge.net/?q=node/4>. y con la versión 1.0 de OpenCV, la cual puede ser descargada de:

http://sourceforge.net/projects/opencvlibrary/files/opencv-win/1.0/OpenCV_1.0.exe/download.

Anexo B

Medida del tiempo de ejecución

Existen diversas formas para medir el tiempo de ejecución de un programa, se debe ejecutar varias veces el mismo programa y tomar el promedio de estos valores, el número de veces elegido fue de 8 veces. La forma más sencilla es utilizando la función clock de la librería time.h, el algoritmo toma la siguiente forma:

```
#include <stdio.h>
#include <time.h>
int main(int argc, char *argv[])
{
    time_t t_inicial, t_final;
    double tiempo;
    double ajuste = 1000.0/CLK_TCK;
    t_inicial = clock();
    /* AQUÍ VA LA FUNCIÓN O EL PROCESO QUE SE QUIERE MEDIR EL TIEMPO*/
    t_final = clock();
    tiempo = (t_final - t_inicial)*ajuste;
    printf("tiempo: %f ms\n", tiempo);
    return 0;
}
```

Esta función no es muy buena, en especial cuando los tiempos de ejecución son menores a 10 ms, en algunos casos cuando se ejecute un algoritmo el tiempo será cero, lo cual no es posible ya que debe tomarle cierto tiempo al programa ejecutarse. Otra posibilidad es utilizando las funciones de OpenCV, las cuales son mucho más precisas con el siguiente algoritmo:

```
#include <cv.h>
#include <stdio.h>
int main(int argc, char *argv[]){
    double tiempo, resultado;
    tiempo= (double)cvGetTickCount();
    /* AQUI VA LA FUNCION O EL PROCESO QUE SE QUIERE MEDIR EL TIEMPO*/
    tiempo = (double)cvGetTickCount() - tiempo;
    Resultado = tiempo/((double)cvGetTickFrequency()*1000.)
    printf( "tiempo = %g ms\n", resultado);
    return 0;
}
```

Algunas veces la función de OpenCV no ofrece buenos resultados, la mayoría de veces ocurre cuando se toma el tiempo y dentro de ella hay declaración de variables, por ello hay que asegurarse de no tener variables dentro de ella, si no se cuenta con estas librerías se puede utilizar la librería Windows.h las cuales están casi en todas las plataformas de programación de c++. El algoritmo se encontró en:

<http://cmasomenos.blogspot.com/2008/03/medir-el-tiempo-de-una-rutina.html>

```
#include <stdio.h>
#include <windows.h>
/* retorna "a - b" en segundos */
double performancecounter_diff(LARGE_INTEGER *a, LARGE_INTEGER *b)
{
    LARGE_INTEGER freq;
    QueryPerformanceFrequency(&freq);
    return (double)(a->QuadPart - b->QuadPart) / (double)freq.QuadPart;
}

int main(int argc, char *argv[])
{
    LARGE_INTEGER t_ini, t_fin;
    double secs;
    QueryPerformanceCounter(&t_ini);
    /* ... PROCESO QUE SE DESEE MEDIR EL TIEMPO... */
    QueryPerformanceCounter(&t_fin);
    secs = performancecounter_diff(&t_fin, &t_ini);
    printf("%.16g milliseconds\n", secs * 1000.0);
    return 0;
}
```

Cuando se mide el tiempo de una rutina, es importante meter esta rutina dentro de un bucle cíclico (for) y medir el tiempo varias veces, después se toma el promedio de todas las medidas para obtener el tiempo de ejecución. Esto se hace muchas veces porque al inicio de una rutina el tiempo va a ser mayor que cuando el programa se sigue ejecutando constantemente.

Anexo C

Seudocódigo

Para entender cómo se realiza la comparación de un par de minucias, se presenta en forma deseudocódigo el algoritmo, se divide en las siguientes partes:

1. Menú principal
2. Alineación
3. Segmentación
4. Normalización
5. Estimación de la orientación
6. Calculo de la máscara de confiabilidad
7. Estimación de la frecuencia local
8. Filtrado
9. Detección de minucias
10. Corrección de falsas minucias
11. Comparación

Algoritmo **MENÚ PRINCIPAL**

VARIABLES

img1[][] : imagen que contiene la huella Sospechosa

img2[][] : imagen que contiene la huella Conocida

MÉTODO

mostrar : muestra una imagen

PROCESO

1. Se cargan las imágenes a procesar.
2. Se realiza la alineación de las huellas, basado en 2 puntos por cada huella.
3. Se realiza el proceso de mejoramiento de la imagen para cada huella.
4. Se detectan y se filtran las minucias de ambas huellas.
5. Se comparan las huellas y se muestra el resultado en la huella.

INICIO

leer (img_H1[][]), leer(img_H2[][]), leer (P1), leer(P2), leer(P3), leer(P4)

img_H2[][] <-- Alineación (img_H1[], P1, P2, P3, P4)

mask_H1[][] <-- Segmentación (img_H1[][])

img_H1[][] <-- Normalización (img_H1[][])