

**UNIVERSIDAD NACIONAL DEL ALTIPLANO
FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA,
ELECTRÓNICA Y SISTEMAS
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**



**IMPLEMENTACIÓN DE UN NUEVO MÉTODO DE CAPTURA DE
HUELLAS DACTILARES PARA UN SISTEMA BIOMÉTRICO**

LIMA – 2019

TESIS

PRESENTADA POR:

LUIS MIGUEL MAMANI HUMPIRI

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO DE SISTEMAS

PUNO – PERÚ

2019

UNIVERSIDAD NACIONAL DEL ALTIPLANO
FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA,
ELECTRÓNICA Y SISTEMAS
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS


**IMPLEMENTACIÓN DE UN NUEVO MÉTODO DE CAPTURA DE
HUELLAS DACTILARES PARA UN SISTEMA BIOMÉTRICO
LIMA – 2019**

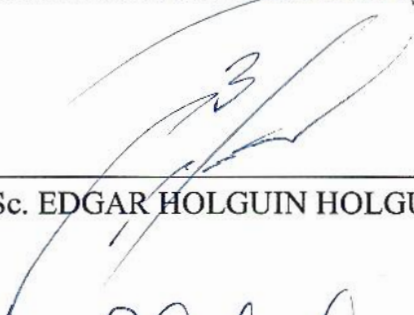
TESIS PRESENTADA POR:
LUIS MIGUEL MAMANI HUMPIRI

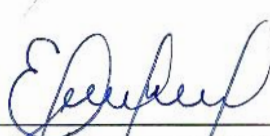



PARA OPTAR EL TÍTULO PROFESIONAL DE:
INGENIERO DE SISTEMAS

APROBADA POR EL JURADO REVISOR CONFORMADO POR:

PRESIDENTE : 
Mg. CARLOS BORIS SOSA MAYDANA

PRIMER MIEMBRO : 
M.Sc. EDGAR HOLGUIN HOLGUIN

SEGUNDO MIEMBRO : 
M.Sc. EDELFRÉ FLORES VELÁSQUEZ

DIRECTOR / ASESOR : 
M.Sc. WILLIAM EUSEBIO ARCAYA COAQUIRA

TEMA: Computación e Ingeniería del Software
ÁREA: Software de base y de aplicación

FECHA DE SUSTENTACIÓN 12 DE JULIO DEL 2019

DEDICATORIA

Para mi compañera de vida, con quien todos los días aprendo el significado de la palabra *amar*. Para mi padre, que me enseñó el valor del trabajo. Para mi madre, que me enseñó a escribir: aún tengo su letra. Para mis amigos, de quienes siempre hay algo que aprender.

... y para todos aquellos que logran vencer sus miedos.

*“I must not fear.
Fear is the mind-killer.
I will face my fear.
I will permit it to pass over me and through me
and when it has gone past I will turn the inner eye to see its path.
Where the fear has gone there will be nothing.
Only I will remain.”*

Dune

ParA la flor de mi vida,
para la luz de mis ojos,
para la soNrisa en mi tristeza,
para la belleza de la mañana,
para lA voz de mi silencio...

AGRADECIMIENTO

Un agradecimiento especial para mis jurados Mg. Carlos Boris Sosa Maydana, M.Sc. Edelfré Flores Velásquez, M.Sc. Edgar Holguin Holguin y para mi director M.Sc. William Eusebio Arcaya Coaquira. Y a todos los docentes de la EPIS, porque ser docente es una tarea noble.

Un agradecimiento especial para la UNA y la EPIS. Allí viví una de las etapas más gratificantes de mi vida.

ÍNDICE GENERAL

ÍNDICE DE FIGURAS	8
ÍNDICE DE TABLAS	12
ÍNDICE DE ANEXOS.....	13
ÍNDICE DE ACRÓNIMOS	14
RESUMEN	15
ABSTRACT.....	16
CAPÍTULO I.....	17
1.1. INTRODUCCIÓN	17
1.1.1. Planteamiento del problema	19
1.1.2. Formulación del problema.....	20
1.1.3. Justificación	20
1.1.4. Limitaciones y restricciones	21
1.2. OBJETIVOS DE LA INVESTIGACIÓN.....	22
1.2.1. Objetivo general	22
1.2.2. Objetivos específicos.....	22
1.3. HIPÓTESIS DE LA INVESTIGACIÓN	23
1.3.1. Hipótesis general	23
CAPÍTULO II. REVISIÓN DE LA LITERATURA.....	24
2.1. ANTECEDENTES DE LA INVESTIGACIÓN	24
2.2. MARCO TEÓRICO.....	30
2.2.1. Biometría	30
2.2.2. Biometría dactilar	30
2.2.3. Dispositivos de captura biométrica	33
2.2.4. Sistema biométrico	34
2.2.5. Rendimiento de un sistema biométrico	36
2.2.6. Procesamiento de imágenes.....	39
2.2.7. Proceso de software	41
2.2.8. Software de Imagen Biométrica del NIST (NBIS).....	42
2.2.9. Librería OpenCV	42
CAPÍTULO III. MATERIALES Y MÉTODOS	44
3.1. POBLACIÓN Y MUESTRA DE INVESTIGACIÓN	44
3.1.1. Ubicación geográfica del estudio	44

3.1.2.	Período de duración del estudio	44
3.1.3.	Población del estudio.....	44
3.1.4.	Muestra del estudio.....	44
3.2.	MÉTODO DE INVESTIGACIÓN	45
3.2.1.	Técnicas e instrumentos de recolección de datos	45
3.2.2.	Procedimiento de recolección de datos	46
3.2.3.	Método de tratamiento de datos	47
3.3.	OPERACIONALIZACIÓN DE VARIABLES	48
3.4.	MÉTODOS EMPLEADOS	49
3.4.1.	Flujo de procesamiento de imágenes.....	49
3.4.2.	Implementación de aplicación móvil.....	49
3.4.3.	Determinación de diferencias en las características de la huella entre ambos métodos de captura.....	49
3.4.3.1.	Extracción de minucias	49
3.4.3.2.	Evaluación de calidad	50
3.4.3.3.	Cálculo del puntaje de coincidencia	51
3.4.4.	Procedimiento de análisis estadístico	52
3.5.	MATERIALES EMPLEADOS	53
3.5.1.	Dispositivo biométrico ZKTECO Live20R.....	53
3.5.2.	Dispositivo móvil Xiaomi Redmi 5 Plus.....	54
3.5.3.	Equipo de desarrollo.....	55
3.5.4.	Entorno integrado de desarrollo	55
3.5.5.	OpenCV	56
3.5.6.	Software NBIS del NIST	56
3.5.7.	Software de análisis estadístico	57
CAPÍTULO IV. RESULTADOS Y DISCUSIÓN		58
4.1.	FLUJO DE PROCESAMIENTO DE IMÁGENES	58
4.1.1.	Detección del objeto de interés.....	59
4.1.2.	Análisis estructural y descriptores de formas	61
4.1.3.	Recorte, mejoramiento y binarización.....	62
4.2.	IMPLEMENTACIÓN DE LA APLICACIÓN MÓVIL	63
4.2.1.	Especificación de requerimientos de software	63
4.2.2.	Prototipos de la aplicación móvil	65

4.3. DIFERENCIAS EN LAS CARACTERÍSTICAS DE LA IMAGEN DE LA HUELLA OBTENIDA POR AMBOS MÉTODOS DE CAPTURA.....	66
4.3.1. Diferencias en el número de minucias.....	66
4.3.2. Diferencias en el índice de calidad.....	70
4.4. RENDIMIENTO DE UN SISTEMA BIOMÉTRICO.....	74
CONCLUSIONES	78
RECOMENDACIONES	80
REFERENCIAS BIBLIOGRÁFICAS	81

ÍNDICE DE FIGURAS

Figura N° 1.1: Capturas de la misma huella dactilar usando diferentes sensores.	19
Figura N° 1.2: Imágenes con distorsiones capturadas por sensores tradicionales.....	21
Figura N° 2.1: Curvas ROC para capturas realizadas, Deb et al. (2018).....	25
Figura N° 2.2: Imagen capturada con aplicación móvil antes y después del tratamiento, Gupta et al. (2017).....	26
Figura N° 2.3: Detección de ROI en entorno controlado, Oude Elfenrink (2014).	29
Figura N° 2.4: Formulario para identificación biométrica dactilar en 1879.....	31
Figura N° 2.5: Características de nivel 2 y nivel 3 de una huella dactilar.....	31
Figura N° 2.6: Sensor biométrico óptico para captura de huellas.	33
Figura N° 2.7: Diagrama conceptual de un sistema biométrico.	36
Figura N° 2.8: Ratio de Falsa Coincidencia (FMR) y Ratio de Falsa No-Coincidencia (FNMR) para un umbral t.	37
Figura N° 2.9: Curva ROC con 4 valores predichos por un clasificador binario.	38
Figura N° 2.10: Puntos de operación típicos para diferentes aplicaciones en una curva ROC.....	39
Figura N° 2.11: Ejemplo de un proceso de adquisición de imágenes digitales.....	40
Figura N° 2.12: Flujo de procesamiento de imágenes en Gupta et al. (2017).....	40
Figura N° 2.13: Flujo de procesamiento de imágenes en Oude Elfenrik (2014).....	41
Figura N° 2.14: Paradigma de desarrollo de software basado en prototipos.....	41
Figura N° 3.1: Diseño del experimento.	45
Figura N° 3.2: Procedimiento de recolección de datos.....	47
Figura N° 3.3: Comando para extracción de minucias mediante la aplicación MINDTCT.....	50
Figura N° 3.4: Comando para cálculo del índice de calidad mediante la aplicación NFIQ.	51
Figura N° 3.5: Comando para el cálculo del puntaje de coincidencia mediante la aplicación BOZORTH.	52
Figura N° 3.6: Fórmula para estandarización de datos.....	52
Figura N° 3.7: Fórmula para normalización softmax.	52
Figura N° 3.8: Procedimiento de análisis estadístico.	53
Figura N° 3.9: Dispositivo biométrico ZKTECO Live20R.....	54
Figura N° 3.10: Dispositivo móvil Xiaomi Redmi 5 Plus.	54

Figura N° 4.1: Flujo de procesamiento de imágenes propuesto.	58
Figura N° 4.2: Fórmula para binarizar la imagen usando el color de la piel.	60
Figura N° 4.3: Imagen de entrada e imagen binarizada luego del análisis de color.	60
Figura N° 4.4: Imagen con región del dedo resaltada y rotada.	61
Figura N° 4.5: Imagen con región del dedo recortada y mejorada.	62
Figura N° 4.6: Captura por cámara y captura por sensor (ratio 1:2).	63
Figura N° 4.7: Fórmula para calcular un puntaje de calidad de una huella capturada por cámara.	63
Figura N° 4.8: Diagrama de clases de diseño de la aplicación móvil.	64
Figura N° 4.9: Diagrama de casos de uso de la aplicación móvil.	65
Figura N° 4.10: Captura de pantalla de la aplicación móvil.	66
Figura N° 4.11: Diagrama de cajas del número de minucias agrupado por método de captura. Histogramas para el número de minucias por cada método de captura.	67
Figura N° 4.12: Resultados de la prueba de normalidad LKS para el número de minucias del grupo cámara.	68
Figura N° 4.13: Resultados de la prueba de normalidad LKS para el número de minucias del grupo sensor.	69
Figura N° 4.14: Resultados de la prueba F de homogeneidad de varianzas para el número de minucias de los grupos: cámara y sensor.	69
Figura N° 4.15: Resultados de la prueba bilateral no paramétrica de Wilcoxon-Mann-Whitney para el número de minucias de los grupos: cámara y sensor.	70
Figura N° 4.16: Resultados de la prueba unilateral no paramétrica de Wilcoxon-Mann-Whitney para el número de minucias de los grupos: cámara y sensor.	70
Figura N° 4.17: Diagrama de cajas del índice de calidad agrupado por método de captura. Histogramas para el índice de calidad por cada método de captura.	71
Figura N° 4.18: Resultados de la prueba de normalidad LKS para el índice de calidad del grupo cámara.	72
Figura N° 4.19: Resultados de la prueba de normalidad LKS para el índice de calidad del grupo sensor.	73
Figura N° 4.20: Resultados de la prueba F de homogeneidad de varianzas para el índice de calidad de los grupos: cámara y sensor.	73

Figura N° 4.21: Resultados de la prueba bilateral no paramétrica de Wilcoxon-Mann-Whitney para el índice de calidad de los grupos: cámara y sensor.	74
Figura N° 4.22: Resultados de la prueba unilateral no paramétrica de Wilcoxon-Mann-Whitney para el índice de calidad de los grupos: cámara y sensor.	74
Figura N° 4.23: Diagramas de cajas del puntaje de comparación para los grupos: cámara-sensor, sensor-cámara, sensor-sensor cuando las dos huellas pertenecen a la misma persona o a personas distintas.....	76
Figura N° 4.24: Curvas ROC cuando se comparan imágenes capturadas por distintos métodos: cámara-sensor (cam-sensor), sensor-sensor (sensor-sensor), cámara-cámara (cam-cam) e imagen sin procesamiento-sensor (raw-sensor).	77
Figura N° A.1: Modelo de casos de uso.	89
Figura N° A.2: Diagrama de clases de análisis.....	94
Figura N° A.3: Diagrama de clases de diseño.	95
Figura N° A.4: Diagrama de secuencia.	96
Figura N° B.1: Flujo de procesamiento de imágenes propuesto.....	106
Figura N° B.2: Imágenes de la etapa de detección del objeto de interés.....	107
Figura N° B.3: Imágenes de la etapa de análisis estructural.....	108
Figura N° B.4: Imágenes de la etapa de mejoramiento.	108
Figura N° C.1: Muestra del conjunto de datos de 40,138 imágenes generadas por el investigador.	110
Figura N° C.2: Arquitectura del modelo completamente convolucional.....	110
Figura N° C.3: Gráfico de pérdida del conjunto de entrenamiento y validación.....	111
Figura N° C.4: Máscaras de segmentación inferidas en imágenes de prueba.	112
Figura N° D.1: Muestra del conjunto de datos de 5,388 imágenes de máscaras de segmentación construidas por el investigador.....	113
Figura N° D.2: Diagrama de cajas de la media y la mediana de los canales HSV de las regiones de las manos.....	115
Figura N° D.3: Diagrama de conglomerados de la mediana de los canales HSV de las regiones de las manos.....	116
Figura N° D.4: Dendograma que confirma la existencia de 4 grupos.	116
Figura N° D.5: Diagrama de cajas de los canales HSV de los 4 grupos encontrados. .	117
Figura N° E.1: Plantilla de minucias obtenidas al ejecutar la aplicación mindtct.	118

Figura N° E.2: Minucias de una imagen capturada por sensor y de una imagen capturada por cámara.....	119
Figura N° E.3: Script bash para extraer las minucias de las imágenes de un directorio.....	119
Figura N° E.4: Script bash para generar el conjunto de datos minutiae.csv.....	119
Figura N° E.5: Estructura del conjunto de datos minutiae.csv.....	120
Figura N° E.6: Script en R para generar el conjunto de datos de números de minucias.....	120
Figura N° F.1: Índice de calidad obtenido al ejecutar la aplicación nfiq.....	121
Figura N° F.2: Script bash para calcular los índices de calidad de las imágenes de un directorio.....	122
Figura N° F.3: Estructura del conjunto de datos nfiq.csv.....	122
Figura N° F.4: Script en R para generar el conjunto de datos de índices de calidad....	122
Figura N° G.1: Puntaje de comparación obtenido al ejecutar la aplicación bozorth3..	124
Figura N° G.2: Script bash para calcular los puntajes de comparación de las plantillas XYT de un directorio.....	124
Figura N° G.3: Estructura del conjunto de datos results.csv.....	124
Figura N° G.4: Script en R para generar el conjunto de datos de puntajes de comparación.....	125

ÍNDICE DE TABLAS

Tabla N° 2.1: Ratio TAR cuando se usa un FAR de 0.1%, Deb et al. (2018).....	24
Tabla N° 2.2: Diferentes patrones de huellas dactilares.	32
Tabla N° 2.3: Ratios de rendimiento de un sistema biométrico.	37
Tabla N° 3.1: Población de estudio.	44
Tabla N° 3.2: Operacionalización de variables.	48
Tabla N° 3.3: Especificaciones técnicas del dispositivo ZKTECO Live20R.....	54
Tabla N° 3.4: Especificaciones técnicas del dispositivo Xiaomi Redmi 5 Plus.	55
Tabla N° 4.1: Estructura del conjunto de datos de números de minucias.	67
Tabla N° 4.2: Estadísticos descriptivos conjunto de datos de número de minucias.....	68
Tabla N° 4.3: Estructura del conjunto de datos de índices de calidad.....	71
Tabla N° 4.4: Estadísticos descriptivos del conjunto de datos de índices de calidad.....	72
Tabla N° 4.5: Estructura del conjunto de datos de puntajes de comparación.....	75
Tabla N° 4.6: Estadísticos descriptivos del conjunto de datos de puntajes de comparación cuando las dos huellas pertenecen a la misma persona o a personas distintas.	75
Tabla N° D.1: Estructura del conjunto de datos de estadísticas de los canales HSV de cada imagen.....	114
Tabla N° D.2: Percentiles 5 y 95 para cada grupo encontrado.	117
Tabla N° E.1: Estructura del conjunto de datos de números de minucias.	120
Tabla N° F.1: Estructura del conjunto de datos de índices de calidad.....	122
Tabla N° G.1: Número de puntajes de comparación agrupados según la fuente de las imágenes: cámara-cámara, cámara-sensor, y sensor-sensor.	123
Tabla N° G.2: Estructura del conjunto de datos de puntajes de comparación.....	125
Tabla N° H.1: Costos de sensores biométricos a Julio del 2019.	126

ÍNDICE DE ANEXOS

ANEXOS.....	88
ANEXO A. Especificación de requerimientos de software para la aplicación móvil	
88	
ANEXO B. Detalle del flujo de procesamiento de imágenes propuesto.....	106
ANEXO C. Modelo de aprendizaje profundo para segmentación de palmas	
basado en una CNN	109
ANEXO D. Análisis de colores HSV para la etapa de detección de objeto de	
interés del flujo propuesto de procesamiento de imágenes	113
ANEXO E. Elaboración del conjunto de datos de número de minucias.....	118
ANEXO F. Elaboración del conjunto de datos de índices de calidad	121
ANEXO G. Elaboración del conjunto de datos de puntajes de comparación	123
ANEXO H. Costos de sensores biométricos a julio del 2019	126

ÍNDICE DE ACRÓNIMOS

AFIS	:	Sistema Automatizado de Identificación de Huellas
DSP	:	Procesamiento de Señales Digitales
EER	:	Ratio de Error Igual
FAR	:	Ratio de Falsa Aceptación
FMR	:	Tasa de Falsa Coincidencia
FNMR	:	Ratio de Falsa No Coincidencia
FPR	:	Ratio de Falso Positivo
FRR	:	Ratio de Falso Rechazo
FTIR	:	Frustrated Total Image Reflection
FVC	:	Competencia de Verificación de Dedos
HSV	:	Matiz, Saturación y Brillo
LKS	:	Lilliefors-Kolmogorov-Smirnov
NBIS	:	Software de Imagen Biométrica del NIST
NIST	:	Instituto Nacional de Estándares y Tecnología de Estados Unidos
ROC	:	Característica Operativa del Receptor
ROI	:	Región de Interés
RTFI	:	Reflection-Based Touchless Finger Imaging
TAR	:	Ratio de Verdadera Aceptación
TPR	:	Ratio de Verdadero Positivo
TTFI	:	Transmission-based Touchless Finger Imaging

RESUMEN

Los sistemas biométricos de reconocimiento de huellas dactilares tradicionales usan imágenes obtenidas por sensores biométricos, estos sensores son costosos y presentan diversos problemas asociados a la interacción con el área de captura (tamaño limitado, transmisión de bacterias, instalación y calibración, distorsiones). Esta investigación tiene como objetivo determinar si existen diferencias en las características de la imagen de la huella (número de minucias e índice de calidad) obtenida por una aplicación móvil que usa un flujo de procesamiento de imágenes propuesto para obtener una imagen de huella dactilar a partir de las imágenes tomadas por la cámara del dispositivo, a diferencia de la imagen de la huella obtenida por un sensor biométrico. Esta investigación de tipo cuasi-experimental fue ejecutada en la ciudad de Lima, en donde se capturaron 184 imágenes con el método propuesto y 321 con un sensor biométrico, con esta información se generaron los conjuntos de datos de número de minucias, índices de calidad y puntajes de comparación. Luego del experimento se ha verificado que existen diferencias en las características de la imagen obtenida por ambos métodos de captura, donde el número de minucias de una imagen obtenida por el método propuesto es mayor al de una imagen obtenida por sensor biométrico. Finalmente, el rendimiento de un sistema biométrico cuando se comparan imágenes obtenidas por el método propuesto contra imágenes obtenidas por un sensor biométrico (cámara-sensor), medido por el ratio de verdadero positivo (TPR), es del 51.75%; y cuando ambas imágenes se obtienen por el mismo método es del 85.39% (cámara-cámara) y del 94.19% (sensor-sensor); por contraste, si imágenes sin ningún tipo de procesamiento se comparan contra imágenes obtenidas por un sensor biométrico (raw-sensor) el TPR es de 1.81%. En todos los casos se usa un ratio de falsos positivos del 2%.

Palabras Clave: biometría dactilar, cámara, aplicación móvil, visión computacional, procesamiento de imágenes

ABSTRACT

Traditional biometric fingerprint recognition systems use images acquired by biometric sensors, these sensors are expensive and present several problems associated with the interaction with its surface area (a limited surface area, transmission of bacteria, installation and calibration, distortions). This research aims to determine if there are differences in the image of a fingerprint acquired by a mobile application that uses an image processing flow to acquire a fingerprint image from the images taken by the device camera, in contrast of the image of the fingerprint acquired from a biometric sensor. This quasi-experimental research was carried out in the city of Lima, where 184 images were captured with the proposed method and 321 with a biometric sensor; with this information the datasets of minutiae count, quality indices and matching scores were generated. Through an experiment it has been verified that there are differences in the characteristics of the image acquired by both methods, where the number of minutiae of an image acquired by the proposed method are greater than that of an image acquired by a biometric sensor. The performance of a biometric system when comparing images acquired by the proposed method against images acquired by a biometric sensor (camera-sensor) measured by the true positive ratio (TPR) is 51.75%; and when both images are acquired by the same method it is 85.39% (camera-camera) and 94.19% (sensor-sensor); by contrast, if images without any type of processing are compared against images acquired by a biometric sensor (raw-sensor), the TPR is 1.81%. In all cases a false positive ratio of 2% is used.

Keywords: fingerprint biometrics, camera, mobile application, computer vision, image processing

CAPÍTULO I.

1.1. INTRODUCCIÓN

La biometría dactilar es una de las biometrías más ampliamente usadas debido a su nivel de unicidad y durabilidad, y su uso es dominante en una gran variedad de aplicaciones (Labati, Piuri & Scotti, 2016), por ejemplo, la autenticación basada en biometría se está convirtiendo en una de las formas de interacción humano-computador más populares (Buciu & Gacsadi 2016). En la actualidad existe una forma predominante de realizar la captura de las impresiones dactilares e involucra el contacto directo de la piel con un dispositivo de captura o sensor biométrico.

El reconocimiento de huellas dactilares sin contacto es un conjunto reciente de técnicas diseñadas para reducir las limitaciones de captura de otras tecnologías ya que permiten aumentar la usabilidad y reducen los costos del hardware con respecto a los sistemas tradicionales de reconocimiento de huellas dactilares. (Dongjae, Kyoungtaek, Heeseung & Jaihie, 2008) Sin embargo, este conjunto de técnicas presenta un gran desafío debido a las condiciones específicas de iluminación, los reflejos, las sombras, el fondo, la posición del dedo, el grado de giro del dedo, y una resolución no consistente de la cámara (Labati et al., 2016; Khalil, Kurniawan & Saleem, 2013).

En esta investigación se ha definido como objetivo determinar si existen diferencias en las características de la imagen de la huella (número de minucias y puntuación de calidad) obtenida por un nuevo método de captura de huellas dactilares a diferencia de la imagen de la huella obtenida por un sensor biométrico. Este nuevo método de captura ha sido implementado en una aplicación móvil que usa la cámara del dispositivo para realizar la captura, en el que todas las imágenes capturadas pasan por un flujo de procesamiento de imágenes que las convierte en una imagen compatible con un sistema biométrico. Se realizó un experimento en el que se compara el número de

minucias y el índice de calidad de las imágenes de las huellas procesadas por la aplicación con las imágenes capturadas por el sensor biométrico. Finalmente se determina el rendimiento del sistema biométrico cuando se comparan imágenes obtenidas por ambos métodos de captura.

La presente investigación se encuentra dividida en cuatro capítulos: En el **CAPÍTULO I**, se menciona la introducción y se detalla el planteamiento y formulación del problema, la justificación, las limitaciones y restricciones, los objetivos e hipótesis de la investigación. En el **CAPÍTULO II, REVISIÓN DE LA LITERATURA**, se desarrolla la revisión de literatura, donde se detallan los antecedentes de la investigación y el marco teórico sobre la que se fundamenta. En el **CAPÍTULO III, MATERIALES Y MÉTODOS**, se presenta el lugar de estudio, la población y la muestra. Asimismo, se describe la metodología de investigación, las técnicas e instrumentos de recolección de datos, la operacionalización de variables, los métodos y materiales empleados en la investigación. En el **CAPÍTULO IV, RESULTADOS Y DISCUSIÓN**, se exponen los resultados de la investigación, donde se detalla el flujo de procesamiento de imágenes propuesto a fin de procesar una imagen de una huella dactilar tomada por la cámara de un teléfono móvil y convertirla en una imagen compatible con un sistema biométrico, la implementación de una aplicación móvil que usa el flujo propuesto de procesamiento de imágenes para evaluar cada fotograma y obtener imágenes de huellas dactilares, el análisis estadístico para determinar si existen diferencias entre las características de las imágenes de huellas dactilares procesadas por la aplicación y las imágenes tomadas por un sensor biométrico, y se determina el rendimiento de un sistema biométrico cuando se comparan imágenes capturadas con ambos métodos. Finalmente, se presentan las **CONCLUSIONES** y **RECOMENDACIONES** de este trabajo de investigación.

1.1.1. Planteamiento del problema

Los sistemas biométricos de reconocimiento basados en huellas dactilares requieren que los usuarios toquen la superficie del sensor con una presión uniforme y apropiada. La diferencia entre las tecnologías involucradas en los sensores biométricos puede ocasionar variaciones en la calidad de las imágenes capturadas e inclusive una disminución del rendimiento de los algoritmos de comparación cuando operan con imágenes capturadas con distintos dispositivos tal y como se puede apreciar en la Figura N° 1.1.

Figura N° 1.1: Capturas de la misma huella dactilar usando diferentes sensores.



FUENTE: CERIASPURDUE (10 de Mayo 2012), minuto 11:12.

Estos sistemas tradicionales son costosos y presentan problemas asociados a la interacción con un área de captura limitada, a la transmisión de bacterias, al transporte, la instalación y calibración de los dispositivos, las distorsiones introducidas en la imagen debido a la presión del dedo sobre el dispositivo, las distorsiones debido a las huellas latentes o fantasmas, las distorsiones debido a la suciedad en el área de captura,

las condiciones del entorno, del dispositivo o del dedo que afectan la imagen capturada. (Gashi, Mason, Lugini, Marasco & Cukic, 2014)

Finalmente, la captura a distancia de huellas dactilares presenta muchos desafíos generados por las condiciones específicas de iluminación y del entorno, los reflejos, las sombras, el fondo de la imagen, la posición, distancia y giro del dedo y una resolución no consistente (Labati et al., 2016; Khalil et al., 2013).

1.1.2. Formulación del problema

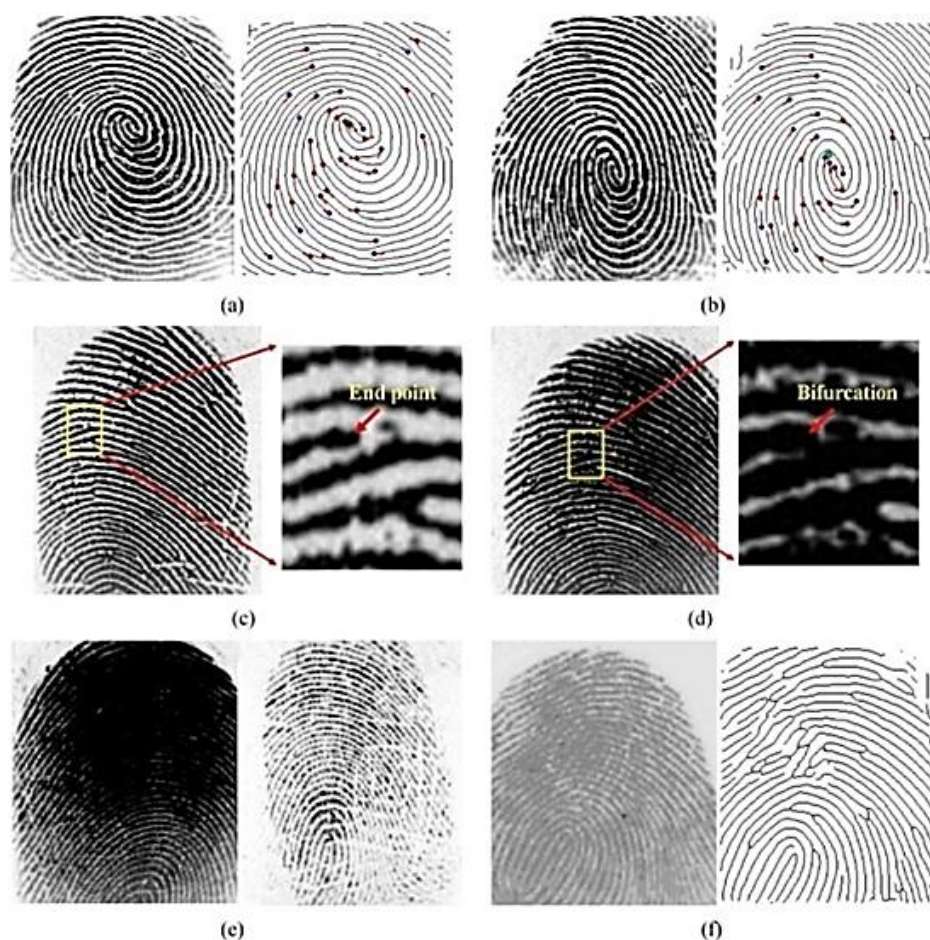
¿Existen diferencias en las características de la imagen de la huella obtenida por un nuevo método de captura de huellas dactilares a diferencia de la imagen de la huella obtenida por un sensor biométrico?

1.1.3. Justificación

El reconocimiento biométrico es el reconocimiento automatizado de individuos en base a sus características biológicas y de comportamiento, tales como: huella dactilar, rostro, iris y voz. Los sistemas de reconocimiento basados en huellas dactilares requieren que los usuarios toquen la superficie del sensor con una presión uniforme y apropiada, ya que si no lo hacen podrían introducir distorsiones en la imagen capturada tal y como lo muestra la Figura N° 1.2 en el que (a) y (b) muestran dos imágenes del mismo dedo que son diferentes, (c) y (d) muestran el cambio de una minucia, (e) muestra una sección de baja calidad y (f) muestra una huella latente.

El reconocimiento de huellas dactilares sin contacto es un conjunto reciente de técnicas diseñadas para reducir las limitaciones de adquisición de otras tecnologías. Los enfoques de captura sin contacto permiten aumentar la usabilidad y reducir los costos del hardware con respecto a los sistemas tradicionales de reconocimiento de huellas dactilares (Dongjae et al., 2008).

Figura N° 1.2: Imágenes con distorsiones capturadas por sensores tradicionales.



FUENTE: Heeseung, Kyoungtaek y Jaihie (2010).

Según Acuity Market Intelligence, el mercado de la biometría móvil experimentará un crecimiento explosivo, pasando de \$1.6 billones en 2014 a \$34.6 billones en 2020. Esta predicción es prevista para todas las modalidades de biometría embebidas en dispositivos móviles (teléfonos inteligentes, tabletas y accesorios inteligentes) (Buciu et al., 2016).

1.1.4. Limitaciones y restricciones

La presente investigación, implementó el flujo de procesamiento de imágenes en una aplicación móvil para capturar y procesar las imágenes de huellas dactilares, es decir esta aplicación sólo está involucrada en el proceso de adquisición de imágenes de

un sistema biométrico. No es el objetivo de esta aplicación realizar otras tareas como extracción de características, enrolamiento, ni comparación de plantillas biométricas.

Así mismo, en el proceso de recolección de datos, se encontraron problemas para encontrar sujetos que formen parte de la investigación, ya que la mayoría de personas se muestran reticentes a compartir sus características biométricas por temas de privacidad; una restricción que también consideraron para no participar del estudio fue que el tiempo total de captura por sujeto es de aproximadamente 45 minutos (10 imágenes con sensor y 10 imágenes con cámara, multiplicada por un número de muestras que varía entre 1 a 6).

1.2. OBJETIVOS DE LA INVESTIGACIÓN

1.2.1. Objetivo general

Determinar si existen diferencias en las características de la imagen de la huella obtenida por un nuevo método de captura de huellas dactilares a diferencia de la imagen de la huella obtenida por un sensor biométrico.

1.2.2. Objetivos específicos

- Implementar un flujo de procesamiento de imágenes que pueda ser aplicado en tiempo real a las imágenes capturadas mediante la cámara de un dispositivo móvil para obtener una imagen de huella dactilar compatible con un sistema biométrico de reconocimiento dactilar basado en minucias.
- Implementar una aplicación móvil que use el flujo de procesamiento de imágenes propuesto para obtener la imagen de la huella dactilar a partir del dedo enfocado por la cámara del dispositivo.
- Determinar el rendimiento de un sistema biométrico de reconocimiento dactilar cuando se comparan imágenes de las huellas obtenidas por un nuevo método de captura contra las imágenes de las huellas obtenidas por un sensor biométrico.

1.3. HIPÓTESIS DE LA INVESTIGACIÓN

1.3.1. Hipótesis general

No existen diferencias en las características de la imagen de la huella obtenida por un nuevo método de captura de huellas dactilares a diferencia de la imagen de la huella obtenida por un sensor biométrico.

CAPÍTULO II.

REVISIÓN DE LA LITERATURA

2.1. ANTECEDENTES DE LA INVESTIGACIÓN

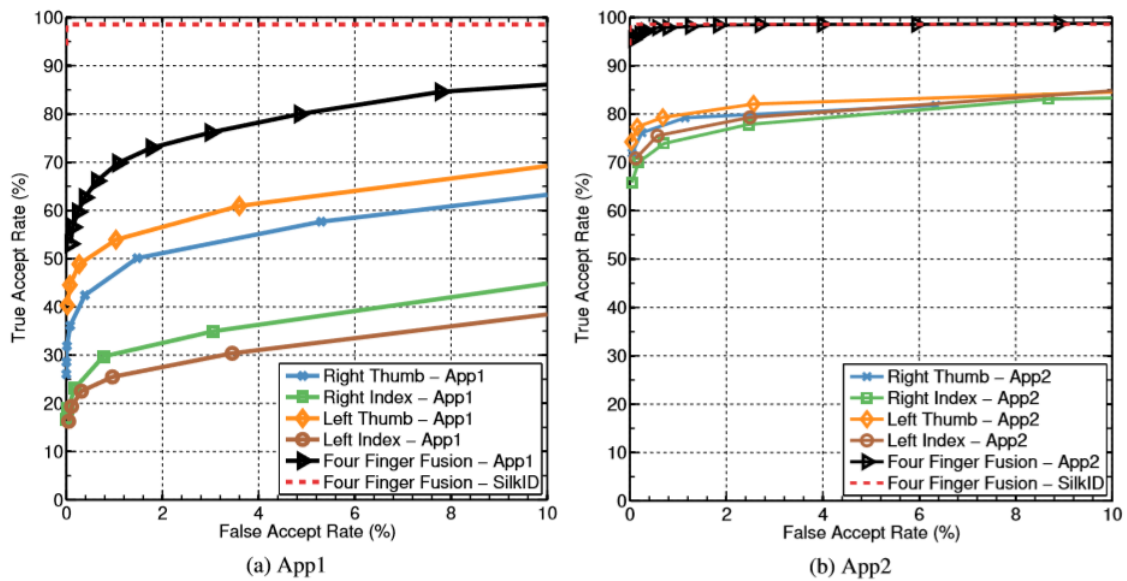
Deb et al. (2018) en el artículo “Matching Fingerphotos to Slap Fingerprint Images” [Comparando fotografías de huellas dactilares contra imágenes decadactilares] estudian el problema de la comparación biométrica entre imágenes de huellas dactilares tomadas con una cámara de un teléfono inteligente contra imágenes de huellas capturadas con sensores biométricos. En esta investigación se utilizaron dos aplicaciones para teléfonos móviles Android y un lector biométrico óptico para la recolección de huellas digitales de 309 sujetos que trabajan principalmente como trabajadores de la construcción, agricultores y trabajadores domésticos. Los resultados muestran que se puede lograr una Tasa de Verdadera Aceptación (TAR) del 95.79% con una Tasa de Falsa Aceptación (FAR) del 0.1% al fusionar los scores de comparación de 4 imágenes de huellas dactilares tomadas con cámara con sus correspondientes tomadas por un sensor biométrico (dos pulgares y dos dedos de índice), usando un comparador de huellas dactilares comercial.

Tabla N° 2.1: Ratio TAR cuando se usa un FAR de 0.1%, Deb et al. (2018).

Tipo de captura	TAR	TAR	TAR
	Android App1	Android App2	Sensor SilkID
Sólo pulgar derecho	37.07	74.14	97.83
Sólo pulgar izquierdo	46.35	76.63	94.20
Sólo índice derecho	21.17	67.96	97.46
Sólo índice izquierdo	18.97	70.79	92.39
Dos pulgares	58.05	89.86	98.55
Dos índices	27.97	83.76	98.19
Pulgar e índice derecho	43.68	86.43	98.55
Pulgar e índice izquierdo	47.51	88.14	94.20
Fusión de 4 dedos	54.98	95.79	98.55

FUENTE: Deb et al. (2018).

Figura N° 2.1: Curvas ROC para capturas realizadas, Deb et al. (2018).



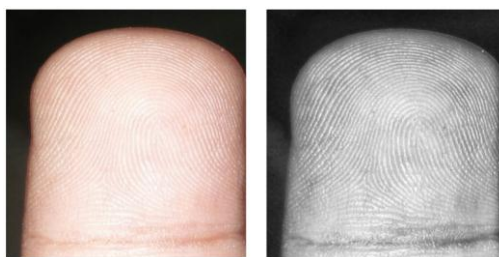
FUENTE: Deb et al. (2018).

En este estudio los autores presentan un protocolo de recopilación de datos y una evaluación del rendimiento de un sistema biométrico comercial entre conjuntos de datos interoperables (imágenes capturadas por teléfonos inteligentes contra imágenes capturadas por lectores biométricos). No se especifica el flujo de procesamiento de imágenes usado en la investigación. Tampoco se especifica la resolución ni las características de las imágenes capturadas.

Gupta, Anand & Rai (2017) en el artículo “Efficient Fingerprint Extraction and Matching Using Smartphone Camera” [Captura y comparación eficiente de huellas dactilares usando la cámara de un teléfono inteligente] proponen un flujo de procesamiento para capturar imágenes de dedos usando la cámara de un teléfono inteligente de manera que puedan ser comparadas con imágenes de sensores ópticos. El método propuesto procesa la imagen capturada usando técnicas de ecualización del histograma y eliminación de ruido, luego el patrón de la huella dactilar es mejorado usando filtros de Gabor y el núcleo de la huella es ubicado para facilitar la alineación del dedo. Finalmente, la imagen es comparada contra una base de datos de imágenes

ópticas para lo cual se usa el software de código abierto NIST-NBIS. Los autores concluyen que la autenticación de huellas dactilares es factible usando cámaras de teléfonos inteligentes, las cuales son comparadas con imágenes obtenidas con un sensor dedicado, el estudio obtiene un EER de 6% a 15%. Se obtiene el mejor score de comparación cuando las imágenes son mejoradas con un filtro de Gabor ($k=21$) y con un ROI de 800×700 píxeles (cámara de 16MP). En este estudio, los autores presentan una propuesta de un flujo de captura de imágenes de huellas dactilares usando una cámara de un teléfono inteligente, no se especifica el conjunto de datos usado en la investigación, ni tampoco la necesidad de usar los conjuntos de datos de la FVC ya que este conjunto de datos sólo contiene imágenes de huellas capturadas por sensores biométricos. La captura es realizada a través de un ROI predefinido de 800×700 píxeles y un ángulo de entrada para el dedo de 180° , por lo que se asume que no existe un proceso de segmentación. No se especifica si el flujo de procesamiento de imágenes es realizado en el dispositivo o en un dispositivo dedicado, tampoco se especifica la resolución de la imagen capturada ni como abordan el problema de la diferencia de resolución con las imágenes del sensor biométrico óptico.

Figura N° 2.2: Imagen capturada con aplicación móvil antes y después del tratamiento, Gupta et al. (2017).



FUENTE: Gupta et al. (2017).

Khalil et al. (2013) en el artículo para la revista “International Journal of Wavelets, Multiresolution and Information Processing Vol. 11 No. 5” titulado “Authentication of Fingerprint Biometrics Acquired using a Cellphone Camera: A

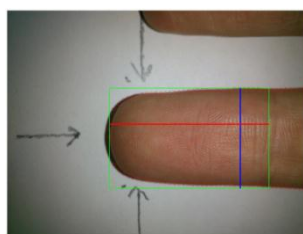
review [Autenticación Biométrica de Huellas Dactilares Adquiridas con una Cámara de Teléfono Móvil: Una reseña]” presenta la tendencia en técnicas de autenticación biométrica de huellas dactilares utilizando teléfonos móviles y explora algunas de las posibilidades futuras en este campo. El artículo presenta un análisis del estado del arte de todas las etapas del proceso que un sistema biométrico basado en imágenes capturadas por cámaras de teléfonos inteligentes debe considerar. Según los autores, estos sistemas poseen una gran complejidad debido a (1) el grado de libertad de la captura (2) la información biométrica del dedo depende mucho de las características de la cámara, la iluminación y la posición del dedo (3) el plano de fondo de la imagen no es controlado. Se describen los métodos aplicados en el estado del arte para cada etapa del proceso. En los métodos para la detección de dedo vivo, se describe el análisis basado en wavelets en cámaras de alta resolución (Moon et al.). Para el análisis de calidad se describe la coherencia de la orientación local (Parziale y Chen). Para el preprocesamiento y mejoramiento de la imagen se describe la normalización de la iluminación (Hiew et al.) y la reducción del desenfoque usando filtros (Piuri and Scotti). Para la segmentación se describe la segmentación basada en la varianza de la imagen en escala de grises (Mehtre), la distribución del color de una región inicial (Lee et al.) y el análisis de colores (Piuri y Scotti, Hiew et al.). Se describen métodos para la detección de características biométricas de nivel 1, como el núcleo y lo(s) delta(s). Para la normalización y alineamiento o, “registro” del dedo, se describen el método basado en el centroide y los ejes de una región (Piuri y Scotti), el alineamiento usando una estimación de la orientación de los gradientes (Kaas y Witkin, Ratha et al.). Para la extracción de características se describen métodos basados en los filtros de Gabor (Hiew y Teoh), el número de cruces (Jain et al., Ratha et al., Thai) y PCA y LDA (Yu et al.). Para el enrolamiento se debe usar entre 3 a 12 muestras. Para la autenticación se

describen métodos que usan diferencias de distancias, distancias euclidianas y similitud de cosenos (Hiew et al.). Este estudio concluye que se requiere una mayor cantidad de investigación en los temas de adquisición de la imagen, procesamiento de imágenes de huellas y seguridad en las referencias biométricas. Adicionalmente, se plantean como trabajos futuros los relacionados a la calidad de las imágenes capturadas y el uso apropiado de recursos en términos de memoria y velocidad.

Oude Elfenrink (2014) en el reporte para el “Grupo de Investigación Services, Cybersecurity and Safety (SCS) de la Universidad de Twente” titulado “Fingerprint Acquisition with a Smartphone Camera [Adquisición de Huellas Dactilares con una Cámara de un Teléfono Móvil]” estudia el procesamiento de imágenes de huellas dactilares capturadas por cámaras de teléfonos inteligentes para ser usado en software comercial para huellas dactilares capturadas por sensores biométricos. Se lleva a cabo un estudio de la literatura sobre el tema de investigación. Posteriormente se establece un conjunto de datos de imágenes capturadas por teléfonos inteligentes y un conjunto de imágenes capturadas con sensores biométricos. Se desarrolla un algoritmo de segmentación, de mejoramiento del patrón de crestas y de conversión a escala de grises. El autor propone un método de captura y mejoramiento de imágenes de huellas dactilares capturadas con cámaras de teléfonos inteligentes. Se concluye que software (extractor y comparador) que procesa imágenes de huellas capturadas por dispositivos biométricos también puede procesar imágenes de huellas capturadas por teléfonos inteligentes previo pre-procesamiento de la imagen. La comparación de SCF (Smartphone Captured Fingerprint)-SCF alcanza un EER 0.15% que es casi el mismo EER de la comparación DSCF (Dedicated Sensor Captured Fingerprint)-DSCF. Sin embargo, cuando se compara SCF-DSCF el EER alcanza un 11.37%. En esta investigación, el autor sólo contempla la etapa de pre-procesamiento de la imagen y la

captura de huellas es realizada en un entorno controlado sin rotación ni inclinación del dedo y con fondo de color uniforme y “no puede ser usada en aplicaciones reales que tienen fondos erráticos”. La investigación tampoco contempla otras dinámicas que se generan cuando la captura es realizada desde un teléfono inteligente como la iluminación, el reflejo de objetos cercanos, las restricciones de poder de cómputo de estos dispositivos, etc. No se menciona las características de la cámara, como la resolución, o el uso de modos de autofocus o macro.

Figura N° 2.3: Detección de ROI en entorno controlado, Oude Elfenrink (2014).



FUENTE: Oude Elfenrink (2014).

Chambi Mamani (2016) en la tesis para optar el grado académico de Magister Scientiae en Informática por la “Universidad Nacional del Altiplano” titulada “Reconocimiento y detección biométrica basado en imágenes de huellas digitales” estudia la aplicación de la biometría dactilar como medio de autenticación y plantea implementar un sistema de reconocimiento basado en imágenes de huellas digitales. Se lleva a cabo un experimento con una muestra aleatoria de 226 huellas de un total de 2884 huellas dactilares tomadas a los postulantes del proceso de admisión Examen CEPREUNA-Julio 2013, la muestra de imágenes pasan por un flujo de procesamiento de imágenes propuesto por el autor. Se plantea la siguiente hipótesis:

Ho: El sistema de reconocimiento desarrollado no tiene una eficiencia aceptable para el procesamiento y comparación de huellas dactilares.

Ha: El sistema de reconocimiento desarrollado tiene una eficiencia aceptable para el procesamiento y comparación de huellas dactilares.

La investigación obtiene un FAR (ratio de falsa aceptación) de “70% a 75%” con un umbral de 33 puntos de correspondencia. En esta investigación, el autor realiza la implementación de un prototipo sobre Matlab, no se especifica cuál es el rendimiento del sistema en términos de memoria y tiempo consumidos, tampoco si existe un conjunto de datos de enrolamiento con el cual se realiza la comparación para el cálculo del FAR. El autor concluye que el procesamiento de huellas digitales es “efectivo” con una probabilidad de significancia del 5%.

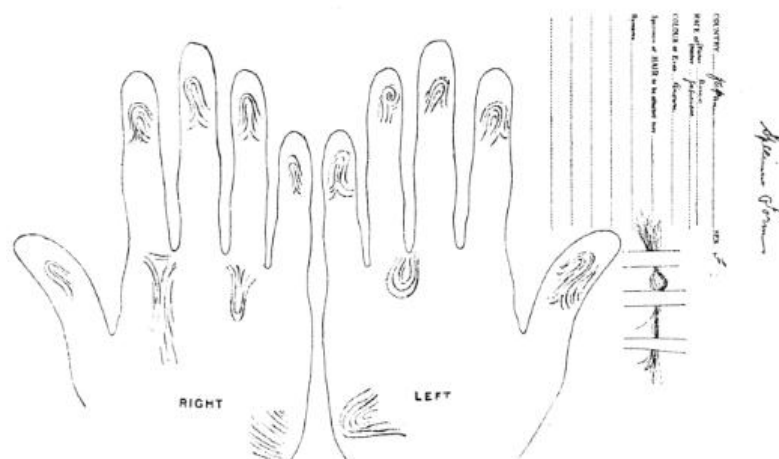
2.2. MARCO TEÓRICO

2.2.1. Biometría

La biometría se relaciona al uso de atributos específicos que reflejan características personales únicas, como una huella digital, una impresión de los vasos sanguíneos oculares o una impresión de voz, con el objetivo de validar la identidad de una persona (norma ISO/IEC 2382:2015). También se define como un término genérico para describir una característica biológica (anatómica y fisiológica) o de comportamiento que puede usarse para el reconocimiento automático, o para describir el proceso automático de reconocimiento basados en estas características biológicas o de comportamiento (ANSI NISTC, 2006).

2.2.2. Biometría dactilar

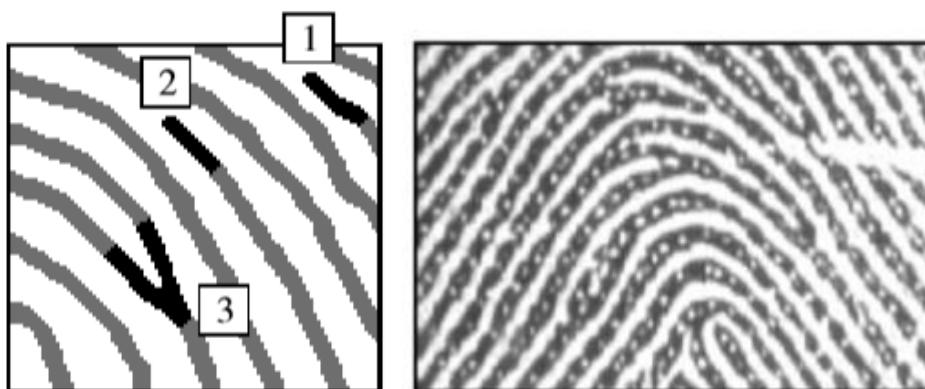
Estas características biológicas han sido usadas hace varios siglos, por ejemplo en cuevas de Nova Scotia hace 2,000 años, en documentos legales en Asiria y Babilonia hace 8,000 años, en documentos gubernamentales en Persia S. XIV (Hawthorne, 2009). Pero estas prácticas antiguas nunca fueron usadas para identificar personas, hasta que en 1877 Sir William Herschel en India usó las huellas dactilares para identificación con propósitos criminales (Faulds, 2015).

Figura N° 2.4: Formulario para identificación biométrica dactilar en 1879.

FUENTE: Faulds (2015).

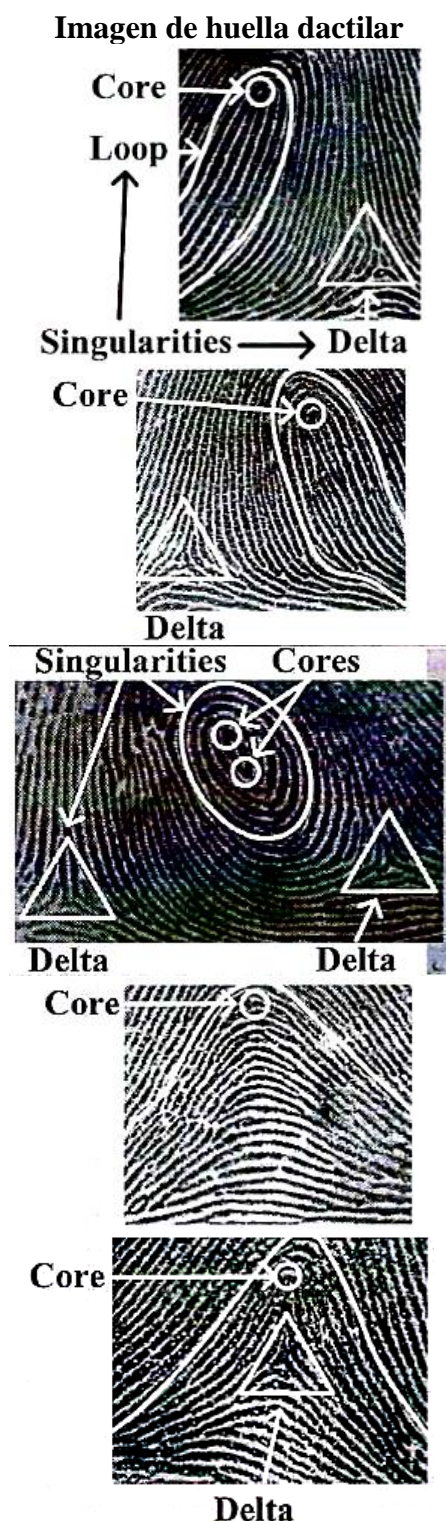
Se calcula que las probabilidades de que un individuo tenga las mismas huellas dactilares que otro es de 1 en 64 billones (Faulds, 2015). En *Finger Prints* (Galton, 2011) se desarrolla el sistema de clasificación de huellas dactilares que se usa hoy en día: arcos, bucles, espirales y compuestas, estos patrones se muestran en la Tabla N° 2.2.

Las huellas dactilares están compuestas por características distintivas en la piel del dedo y pueden ser clasificadas como características de nivel 1 o patrones (núcleo y delta), características de nivel 2 o minucias (densidad, número, orientación y topología de crestas y valles), y características de nivel 3 o poros y forma de los valles.

Figura N° 2.5: Características de nivel 2 y nivel 3 de una huella dactilar.

FUENTE: Maltoni, Maio, Jain & Prabhakar (2009).

Tabla N° 2.2: Diferentes patrones de huellas dactilares.



Patrón de huella dactilar

Bucle izquierdo

Todos los bucles tienen un delta. Los bucles se describen según la dirección de la pendiente descendente del bucle.

Comprenden el 31.7% de las características de la huella digital.

Bucle derecho

Comprenden el 2,9% de las características de la huella digital.

Espiral

En una espiral, las crestas giran alrededor de un núcleo (o núcleos). Todas las espirales tienen dos deltas. Comprenden el 27,9% de las características de huellas dactilares.

Arco

En un arco, las crestas fluyen en una dirección. Comprenden el 3.7% de las características de huellas dactilares.

Arco tendido

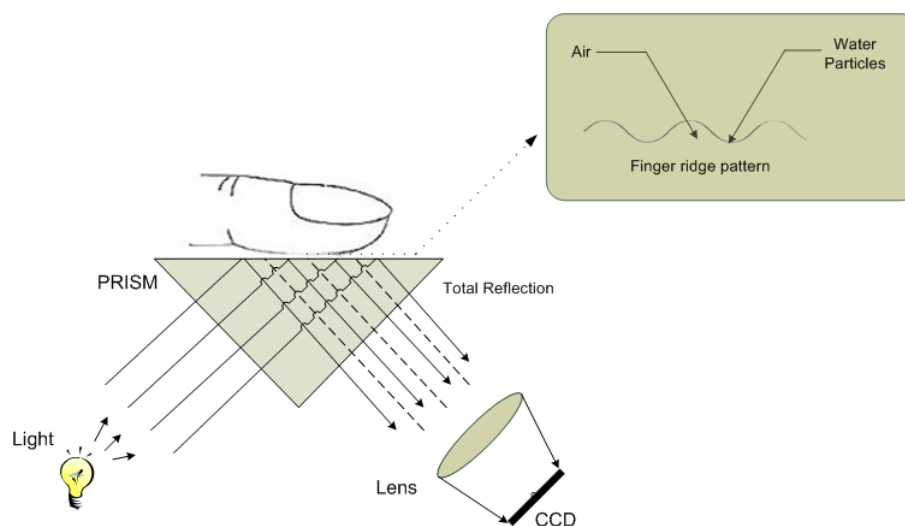
En un arco tendido, las crestas contienen un bucle y una delta. Comprenden el 33.8% de las características de las huellas dactilares.

FUENTE: IEEE CBP (2012).

2.2.3. Dispositivos de captura biométrica

A fin de procesar la biometría es necesario tener una muestra o una captura biométrica. Existen dos formas principales de realizar una captura offline: mediante superficies planas con tinta o mediante un dispositivo con contacto directo o indirecto. Para captura en vivo se pueden usar dispositivos ópticos, electro-ópticos, de estado sólido (capacitivos), por ultrasonido, piezoeléctricos, termales, multispectrales y sin contacto (Maltoni et al., 2009).

Figura N° 2.6: Sensor biométrico óptico para captura de huellas.



FUENTE: Derawi, Yang & Busch (2012).

Los dispositivos de captura biométrica sin contacto pueden usar cualquiera de los siguientes métodos (IEEE CBP, 2012):

- Por reflexión (RTFI). Múltiples fuentes de luz monocromática iluminan el dedo de tal manera que la luz rebota en una sola cámara. La imagen incluye datos ópticos de los valles, así como de las crestas; a diferencia de los sensores basados en contacto que producen una imagen sólo de las crestas con valles registrados solo por la ausencia de luz reflejada. Este método de captura de imágenes tiene inconvenientes. El usuario debe mantener su dedo estable y a una cierta distancia y la resolución puede variar según la imagen.

- Por transmisión (TTFI). Una luz visible atraviesa el dedo del lado de la uña y se desplaza hacia el sensor que se encuentra del lado de la huella. Tanto los métodos TTFI como RTFI son vulnerables a la falsificación usando imágenes 2D, como fotografías.
- Tridimensionales. Para producir una imagen tridimensional se debe capturar vistas equivalentes del dedo mediante RTFI o usar diversos sensores en un semicírculo a su alrededor. Las múltiples vistas se pueden combinar en una sola imagen y a su vez transformarse en un equivalente tridimensional proyectándolas en una superficie cilíndrica que simule un dedo (modelo paramétrico), para luego ser "desenrolado" en una impresión plana bidimensional. O, utilizando algoritmos que colocan las minucias donde estarían en una impresión enrollada físicamente (modelo no paramétrico). Los métodos no paramétricos pueden preservar mejor las características superficiales de las formas irregulares.

2.2.4. Sistema biométrico

Un sistema biométrico se define como:

Un sistema de reconocimiento de patrones que reconoce a una persona al determinar la autenticidad de una característica fisiológica y/o de comportamiento que posee. [...] Un sistema de verificación autentica la identidad de una persona al comparar sus características biométricas capturadas con las plantillas biométricas previamente almacenadas en el sistema. Realiza una comparación uno a uno para determinar si la identidad afirmada por el individuo es verdadera. Un sistema de verificación rechaza o acepta dicha afirmación (¿Soy quien digo ser?); [...] Un sistema de identificación reconoce a un individuo buscando una

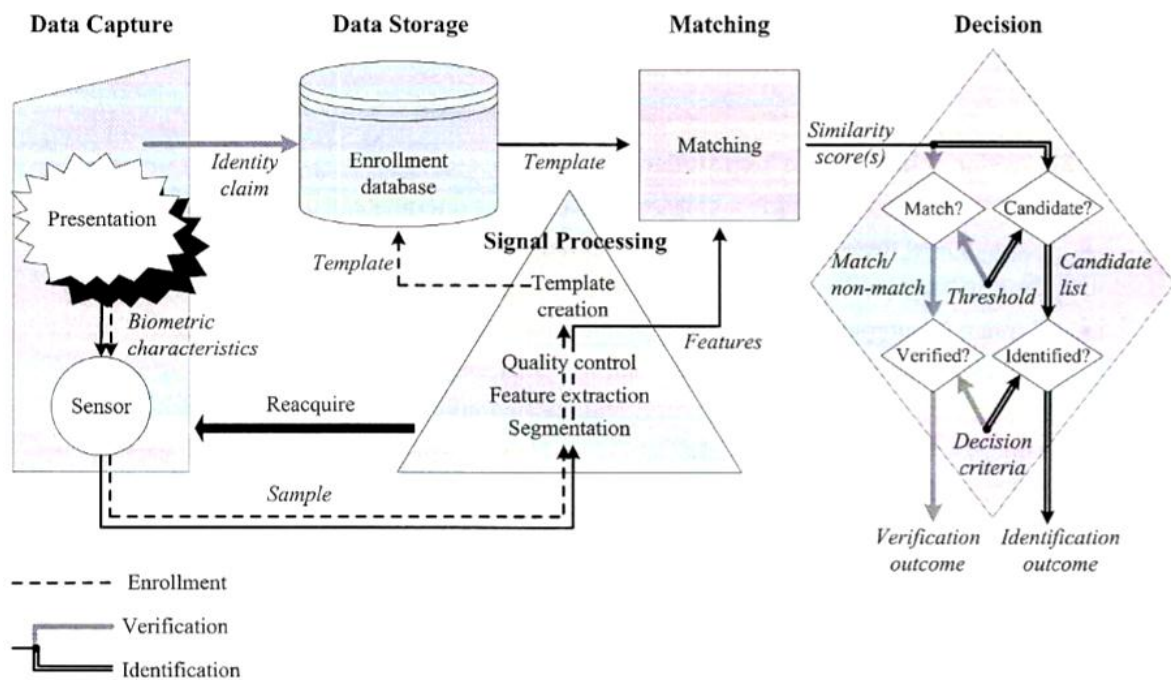
coincidencia en toda la base de datos. Realiza comparaciones de uno a muchos para establecer la identidad del individuo. En un sistema de identificación, el sistema establece la identidad del sujeto sin que el sujeto tenga que afirmar una identidad, o falla si el sujeto no está inscrito en la base de datos (¿Quién soy yo?). (Maltoni et al., 2009, p. 3)

Un Sistema Automatizado de Identificación de Huellas (AFIS) se define como:

El proceso automatizado (A) que ha eliminado la necesidad de un clasificador de impresión para ubicar las tarjetas de huellas de un archivo y compararlas. La base de datos de búsqueda está compuesta por imágenes de huellas digitales (F) recolectadas de individuos mediante el uso de tarjetas de huellas digitales o mediante captura electrónica utilizando un dispositivo similar a un escáner. El aspecto de identificación (I) se produce cuando las huellas digitales de la persona son capturadas y las imágenes resultantes se buscan en la base de datos de imágenes de huellas digitales registradas en la base de datos local, estatal o nacional. Se considera un sistema (S) porque utiliza computadoras y software y puede interactuar con subsistemas y otros sistemas de identificación, incluidos otros sistemas AFIS. (Komarinski, 2005, p. 4)

El diagrama presentando en la Figura N° 2.7 representa un modelo de un sistema biométrico y de las principales operaciones que pueden llevarse a cabo (enrolamiento, verificación e identificación). El enrolamiento es el proceso de registro de nuevas plantillas biométricas en el sistema, la verificación es el proceso de comparación de una plantilla biométrica contra la plantilla registrada en la base de datos, la identificación es el proceso de reconocimiento de una plantilla biométrica.

Figura N° 2.7: Diagrama conceptual de un sistema biométrico.



FUENTE: IEEE CBP (2012).

2.2.5. Rendimiento de un sistema biométrico

El Ratio de Falsa Aceptación (FAR) y el Ratio de Falso Rechazo (FRR) suelen ser medidas de laboratorio de la precisión de un sistema biométrico. (Haupt & Mozer, 2015) Sin embargo, estos ratios no son los únicos usados para este fin, en la Tabla N° 2.3 se muestran otros ratios comúnmente usados.

La evaluación de la precisión de un sistema biométrico se basa en las distribuciones de los puntajes de comparación de huellas dactilares idénticas (que pertenecen a la misma persona) y no idénticas (que no pertenecen a la misma persona). Tal y como se muestra en la Figura N° 2.8, si la plantilla biométrica se representa como T y la entrada adquirida como I . Luego, las hipótesis nula y alternativa son:

$H_0: I \neq T$, la entrada I no proviene de la misma persona que la plantilla T ;

$H_1: I = T$, la entrada I proviene de la misma persona que la plantilla T .

D_0 : la persona no es quien dice ser;

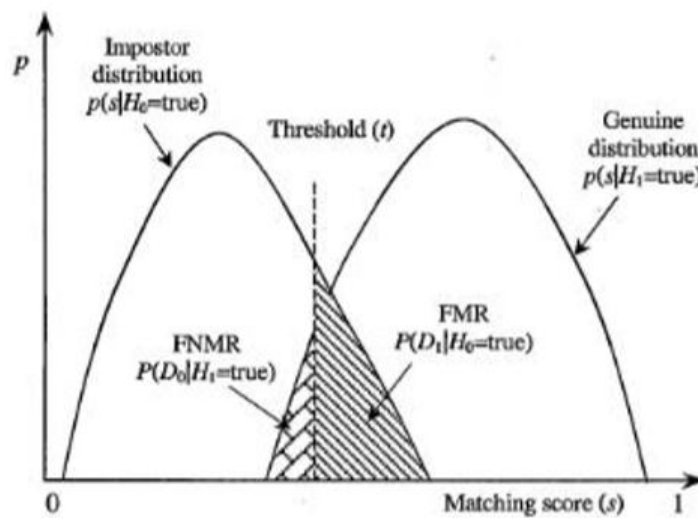
D_1 : la persona es quien dice ser.

Tabla N° 2.3: Ratios de rendimiento de un sistema biométrico.

COMPARACIÓN	Verificación	FALSA COINCIDENCIA FMR: False Match Rate (Ratio de Falsa Coincidencia)	FALSA NO COINCIDENCIA FNMR: False Non-Match Rate (Ratio de Falsa No Coincidencia)
	Identificación	FPIR: False Positive Identification Rate (Ratio de Falsos Positivos de Identificación)	FNIR: False Negative Identification Rate (Ratio de Falsos Negativos de Identificación)
DECISIÓN	Verificación	FAR: False Accept Rate (Ratio de Falsa Aceptación)	FRR: False Rejection Rate (Ratio de Falso Rechazo)
	Identificación	FAR: False Alarm Rate (Ratio de Falsa Alarma)	

FUENTE: IEEE CBP (2012)

Figura N° 2.8: Ratio de Falsa Coincidencia (FMR) y Ratio de Falsa No-Coincidencia (FNMR) para un umbral t .

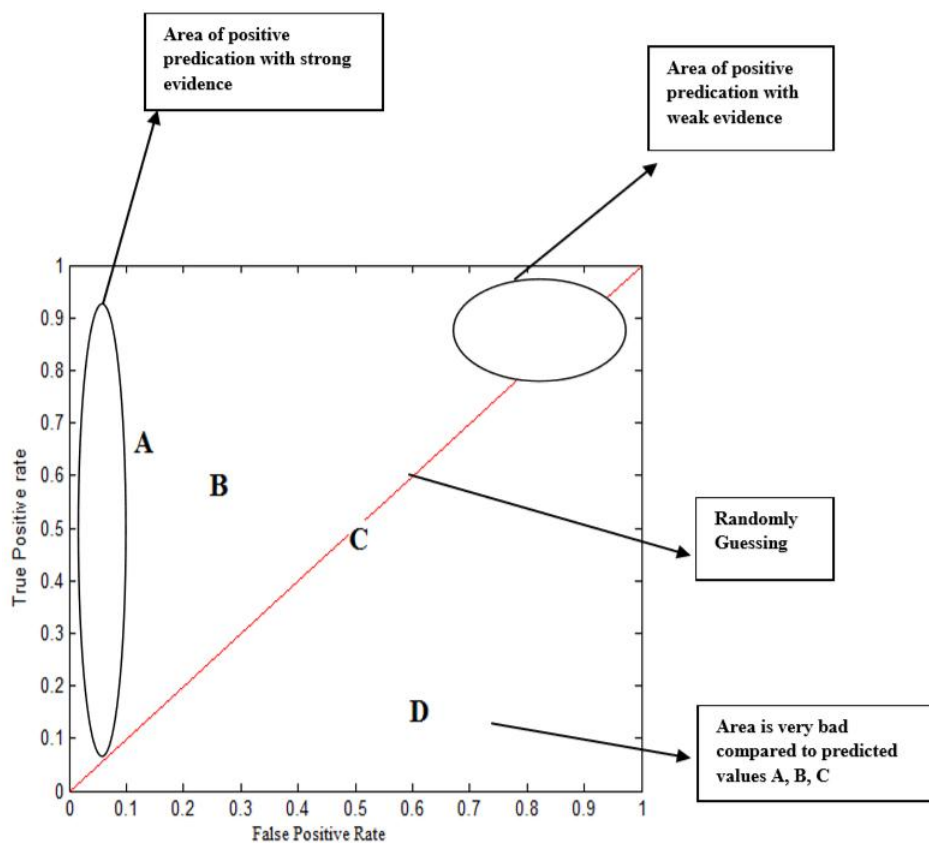


FUENTE: Maddala (2010).

Si el score de comparación (T, I) es mayor que el umbral t , la puntuación se clasifica en $D1$, de lo contrario, se categoriza en $D0$. Dichas pruebas de hipótesis contienen dos tipos de errores: **Error de tipo 1**: falsa coincidencia (se decide $D1$ cuando $H0$ es verdadero); **Error de tipo 2**: falso no-coincidencia (se decide $D0$ cuando $H1$ es verdadero). La tasa de falsa coincidencia (FMR) es la probabilidad de error de tipo 1 (probabilidad de que se decida que la persona es quien dice ser cuando la entrada I no proviene de la misma persona que la plantilla I) y la tasa de falsa no coincidencia (FNMR) es la probabilidad de error de tipo 2.

La técnica de la curva ROC (Característica Operativa del Receptor) que se usa para visualizar el rendimiento de un clasificador también puede ser usada en un sistema biométrico. (Maddala, 2010).

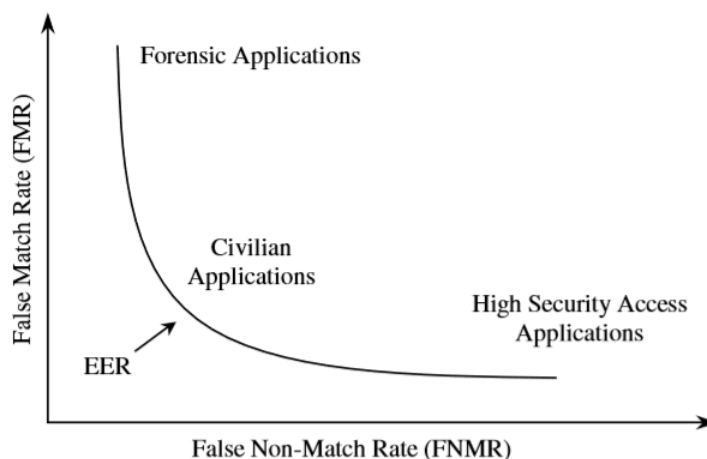
Figura N° 2.9: Curva ROC con 4 valores predichos por un clasificador binario.



FUENTE: Maddala (2010).

Un diagrama parecido se usa para calcular el mejor umbral de clasificación para un sistema biométrico según la aplicación. Por ejemplo para una aplicación civil se preferiría un umbral con el que el FNMR = FMR, a este umbral suele llamarse EER.

Figura N° 2.10: Puntos de operación típicos para diferentes aplicaciones en una curva ROC.



FUENTE: Maltoni et al. (2009).

2.2.6. Procesamiento de imágenes

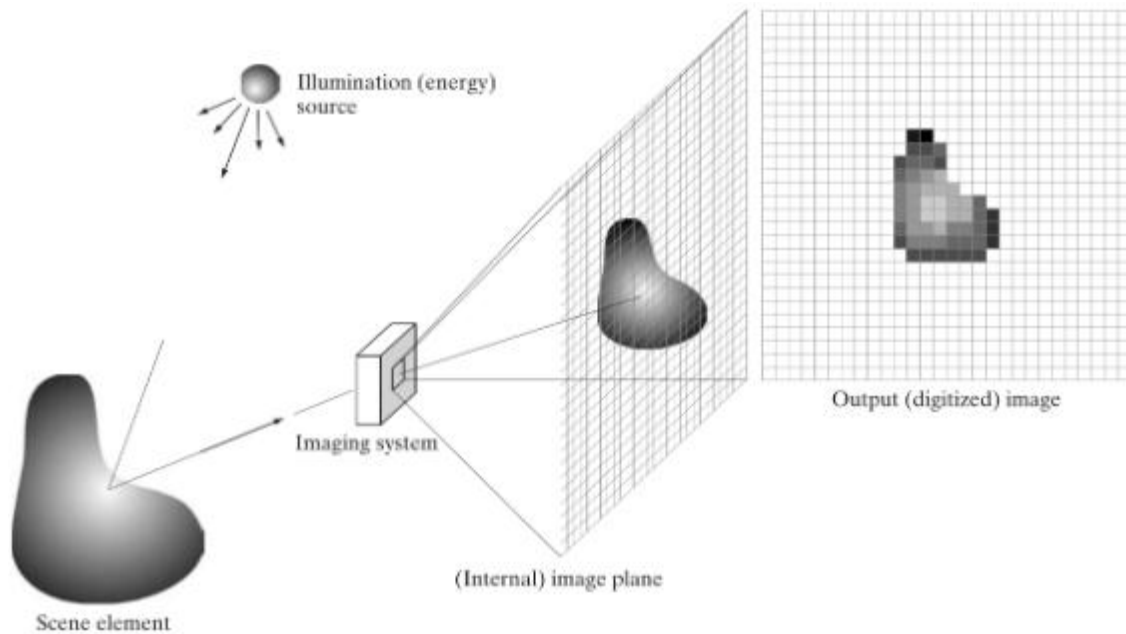
El Procesamiento de Señales Digitales (DSP) es el conjunto de técnicas, algoritmos y matemática que se usa para manipular las señales originadas como datos sensoriales del mundo real, tales como, imágenes, ondas de sonido, etc. (Smith, 2011). El interés en el procesamiento digital de imágenes proviene de dos áreas principales: el mejoramiento de la información de las imágenes con fines de interpretación y el procesamiento de los datos de la imagen para almacenamiento, transmisión y representación con fines de reconocimiento automático (Gonzales & Woods, 2008).

Una imagen puede definirse como:

[...] una función bidimensional, $f(x, y)$, donde x e y son coordenadas espaciales (planos) y la amplitud de f en cualquier par de coordenadas (x, y) se denomina intensidad o nivel de gris de la imagen en ese punto. Cuando x, y , los valores de intensidad de f son cantidades finitas y

discretas, llamamos a la imagen digital [...] una imagen digital se compone de un número finito de elementos, cada uno de los cuales tiene una ubicación y valor particular. (Gonzales & Woods, 2008, pp. 1-2)

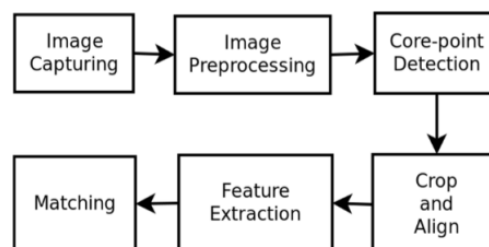
Figura N° 2.11: Ejemplo de un proceso de adquisición de imágenes digitales.



FUENTE: Gonzales y Woods (2008).

Gupta et al. (2017) plantea el siguiente flujo de procesamiento para la captura de imágenes desde teléfonos inteligentes:

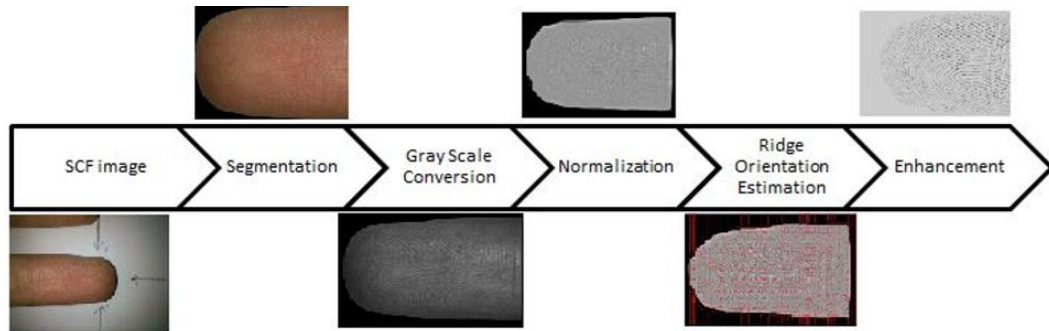
Figura N° 2.12: Flujo de procesamiento de imágenes en Gupta et al. (2017).



FUENTE: Gupta et al. (2017).

Oude Elfenrink (2014) plantea el siguiente flujo de procesamiento:

Figura N° 2.13: Flujo de procesamiento de imágenes en Oude Elfenrink (2014).

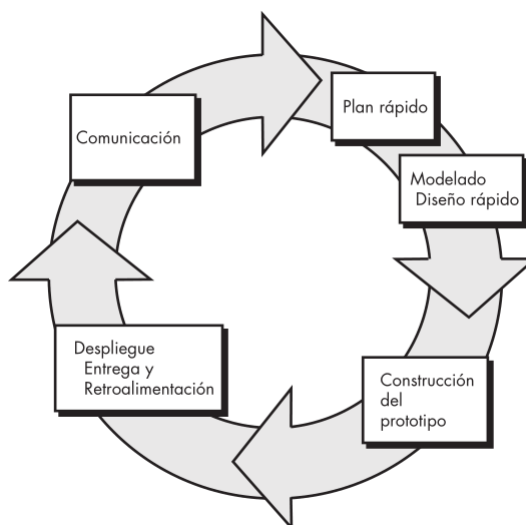


FUENTE: Oude Elfenrink (2014).

2.2.7. Proceso de software

Un proceso de software es un conjunto coherente de actividades para la producción de software. Existen diferentes procesos de software pero todos deben incluir cuatro actividades que son fundamentales: la especificación del software, el diseño e implementación del software, la validación del software y la evolución del software (Sommerville, 2011). Dentro de los modelos de proceso de software evolutivo encontramos el paradigma de prototipos que se caracteriza por ser iterativo y por construir versiones cada vez más completas:

Figura N° 2.14: Paradigma de desarrollo de software basado en prototipos.



FUENTE: Pressman y Maxim (2014).

2.2.8. Software de Imagen Biométrica del NIST (NBIS)

El Software de Imagen Biométrica (NBIS) es desarrollado por el Instituto Nacional de Estándares y Tecnología de Estados Unidos (NIST), y puede ser distribuido libremente sin requisitos de licencia y se considera de dominio público (NIST NBIS, 2019). El software NBIS está compuesto por programas y sus librerías; los paquetes principales son (NIST NBIS Userguide, 2019):

- MINDTCT es un extractor de minucias de huellas dactilares.
- NFIQ es un algoritmo para determinar la calidad de una huella dactilar.
- BOZORTH3 es un comparador de huellas dactilares basado en minucias.
- Otros utilitarios como PCASYS, AN2K7, IMGTOOLS, NFSEG.

2.2.9. Librería OpenCV

OpenCV es una librería optimizada para aplicaciones de visión computacional y de aprendizaje automático en tiempo real. OpenCV es de código abierto y es publicada bajo la Licencia BSD 3-Clause. Es gratis para uso comercial. OpenCV está escrito de forma nativa en C++ y tiene interfaces para C++, Python, Java y MATLAB y es compatible con Windows, Linux, Android y Mac OS. (OpenCV, 2019)

OpenCV contiene más de 500 algoritmos optimizados para el análisis de imágenes y de videos, y desde su introducción en 1999, ha sido ampliamente adoptada como la principal herramienta de desarrollo por la comunidad de investigadores y desarrolladores en visión computacional. Desde la versión 2.2 la librería se encuentra dividida en módulos, siendo los más comunes (Laganière, 2012):

- `opencv_core` contiene estructuras de datos básicas y funciones aritméticas.
- `opencv_imgproc` contiene funciones de procesamiento de imágenes principales.
- `opencv_highgui` contiene funciones de lectura y escritura de imágenes y videos, también contiene funciones de interfaz de usuario.

- `opencv_features2d` contiene detectores y descriptores de puntos característicos.
- `opencv_calib3d` contiene calibración de cámara, estimación de geometría de dos vistas y visión estéreo.
- `opencv_video` contiene estimadores de movimiento, seguimiento de características y extracción de primer plano.
- `opencv_objdetect` contiene detectores de objetos, como detectores de rostro y de personas.

CAPÍTULO III. MATERIALES Y MÉTODOS

3.1. POBLACIÓN Y MUESTRA DE INVESTIGACIÓN

3.1.1. Ubicación geográfica del estudio

La presente investigación se realizó en la ciudad de Lima, Perú, en una institución privada que brinda servicios de desarrollo de software relacionado a la biometría.

3.1.2. Período de duración del estudio

La presente investigación se realizó en el primer semestre del año 2019.

3.1.3. Población del estudio

La población de estudio está constituida por los individuos enrolados en la base de datos biométrica de una institución privada.

Tabla N° 3.1: Población de estudio.

	Población
Individuos	525

Elaboración propia.

3.1.4. Muestra del estudio

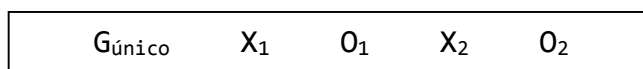
La selección de la muestra fue del tipo no probabilístico (Hernández, 2010). Se tomó del total de individuos a aquellos que se encuentran geográficamente accesibles, de los cuales aceptaron participar en el estudio 9. Al finalizar el proceso de captura de imágenes se obtuvieron 184 imágenes capturadas con el método de captura propuesto (cámara del dispositivo móvil) y 321 imágenes capturadas con el sensor biométrico. Se capturaron imágenes de los 10 dedos de cada participante.

Debido a que la información biométrica representa información confidencial y privada y que para obtenerla se requiere la presencia física del sujeto, varios de los individuos rechazaron su participación en el estudio.

3.2. MÉTODO DE INVESTIGACIÓN

La presente investigación es de tipo aplicada con diseño cuasi-experimental (diseño con tratamientos múltiples y un solo grupo). A toda la muestra se le administró el tratamiento experimental #1 (obtención de huella dactilar con un dispositivo biométrico óptico) y se registró la observación #1, luego, se le administró el tratamiento experimental #2 (obtención de huella dactilar con la cámara de un dispositivo móvil) y se registró la observación #2:

Figura N° 3.1: Diseño del experimento.



Elaboración propia.

Donde:

- $G_{\text{único}}$: muestra
- X_1 : administración del tratamiento con el sensor biométrico
- O_1 : observación #1
- X_2 : administración del tratamiento con la cámara del dispositivo móvil
- O_2 : observación #2

3.2.1. Técnicas e instrumentos de recolección de datos

Para obtener las imágenes de huellas dactilares con un dispositivo biométrico óptico se utilizó:

- Dispositivo móvil Xiaomi Redmi 5 Plus con sistema operativo Android 8.0.1.
- Aplicación móvil para captura de huellas mediante sensores biométricos ZKTeco desarrollada por el investigador.
- Sensor biométrico ZKTeco Live20R.

Para obtener las imágenes de huellas dactilares con la cámara de un dispositivo biométrico móvil se utilizó:

- Dispositivo móvil Xiaomi Redmi 5 Plus con sistema operativo Android 8.0.1.
- Aplicación móvil para captura de huellas mediante el API de la cámara desarrollada por el investigador, basado en el proyecto de código abierto OpenCamera.

Para extraer las minucias y el índice de calidad de las imágenes capturadas por ambos métodos se utilizó:

- MINDTCT del paquete NBIS 5.0.0 para la extracción de minucias.
- NFIQ del paquete NBIS 5.0.0 para el cálculo del índice de calidad.

El conjunto de datos generado contiene las siguientes mediciones:

- Tipo de captura.
- Número de minucias por imagen.
- Índice de calidad por imagen.

Para generar los puntajes de coincidencia de la combinación en grupos de 2 de todas las imágenes capturadas por ambos métodos forman se utilizó:

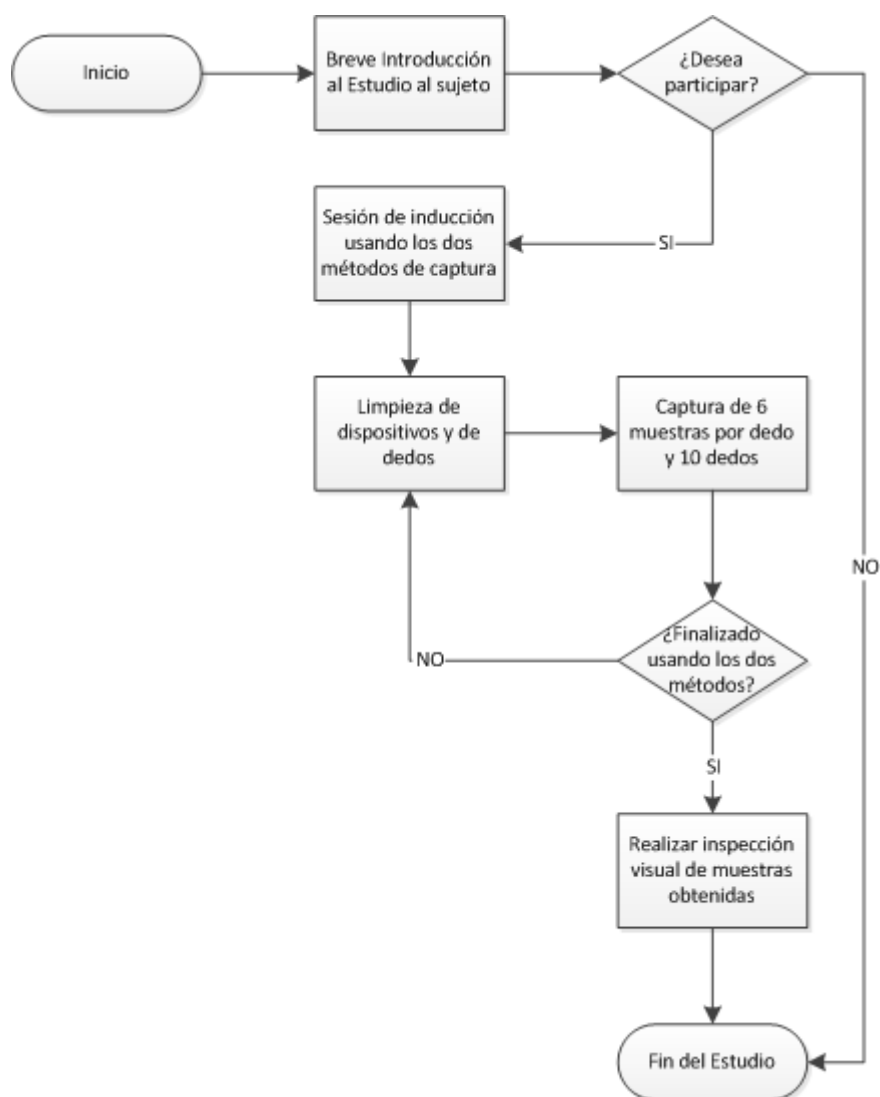
- BOZORTH del paquete NBIS 5.0.0 para el cálculo del puntaje de coincidencia.

El conjunto de datos generado contiene las siguientes mediciones:

- Tipo de comparación.
- Puntaje de coincidencia.

3.2.2. Procedimiento de recolección de datos

Las instrucciones, técnicas e instrumentos de trabajo fueron los mismos para todos los participantes. La recolección de datos se realizó en diferentes ambientes y en diferentes momentos. La persona que aplicó el tratamiento fue la misma (el investigador).

Figura N° 3.2: Procedimiento de recolección de datos.

Elaboración propia.

3.2.3. Método de tratamiento de datos

El tratamiento de datos se realizó de la siguiente forma:

1. Captura de imágenes de huellas dactilares usando un sensor biométrico para todos los participantes.
2. Captura de imágenes de huellas dactilares usando la cámara de un dispositivo móvil para todos los participantes.
3. Inspección visual de imágenes capturadas.
4. Generación de conjunto de datos: número de minucias e índice calidad.

5. Generación de conjunto de datos: puntaje de coincidencia.
6. Análisis exploratorio de los conjuntos de datos.
7. Ejecución del análisis estadístico (validación de las hipótesis estadísticas mediante la prueba de Wilcoxon-Mann-Whitney).

3.3. OPERACIONALIZACIÓN DE VARIABLES

Tabla N° 3.2: Operacionalización de variables.

Variable	Dimensión	Escala	Indicador
INDEPENDIENTE			
Método de captura de la imagen de la huella dactilar	Método de captura	sensor, cámara	<p>sensor: Imagen de huella dactilar obtenida por sensor biométrico.</p> <p>cámara: Imagen de huella dactilar obtenida por la cámara de dispositivo móvil a la que se le ha aplicado el flujo de procesamiento de imágenes propuesto.</p>
DEPENDIENTE			
Características de la imagen de la huella dactilar	Número de minucias Puntuación o índice de calidad	≥ 0 1 – 5	<p>Número de minucias obtenido por componente MINDTCT (NBIS).</p> <p>Puntuación de calidad obtenida por componente NFIQ (NBIS).</p>

Elaboración propia.

Método de captura de la imagen de la huella: Proceso por el que se obtiene la imagen de una huella dactilar, puede ser: mediante un sensor/dispositivo biométrico óptico (captura por contacto) o, mediante la cámara de un dispositivo móvil a la que se le ha aplicado el flujo de procesamiento de imágenes propuesto (captura sin contacto).

Características de la imagen de la huella: Se compone de dos dimensiones:

- El número de minucias, es el número de características de nivel 2 o puntos de interés de una huella dactilar, tales como las bifurcaciones y las terminaciones de crestas. Obtenido por el componente MINDTCT del paquete NBIS.
- El índice o puntuación de calidad, es el resultado del algoritmo de evaluación de la calidad de imágenes de huellas dactilares NFIQ. Obtenido por el componente NFIQ del paquete NBIS.

3.4. MÉTODOS EMPLEADOS

3.4.1. Flujo de procesamiento de imágenes

El flujo de procesamiento de imágenes propuesto se efectuó a partir de la experimentación y la aplicación de técnicas de visión computacional encontradas en la literatura. A fin de facilitar una rápida iteración y para realizar la experimentación de forma interactiva, inicialmente se usó el lenguaje Python como Jupyter Notebook sobre la plataforma Anaconda (Anaconda Inc., 2019).

3.4.2. Implementación de aplicación móvil

El proceso de implementación de la aplicación móvil se realizó usando el modelo de desarrollo evolutivo (Pressman, 2014), el cual se caracteriza por ser iterativo y por construir versiones cada vez más completas. La aplicación desarrollada forma parte de la etapa de adquisición de muestras de un sistema biométrico de reconocimiento dactilar. La Especificación de Requisitos de Software (ERS) para la aplicación se basó en el estándar IEEE 830.

3.4.3. Determinación de diferencias en las características de la huella entre ambos métodos de captura

3.4.3.1. Extracción de minucias

La extracción de minucias es realizada por el componente MINDCT (Minutiae Detection) del paquete NBIS (NIST Biometric Image Software):

MINDTCT toma una imagen de huella dactilar y localiza todas las minucias de la imagen, asignando a cada minucia su ubicación, orientación, tipo y calidad. El comando mindtct lee una imagen de huella digital de un archivo ANSI/NIST, WSQ, JPEG o en formato IHead. [...] Luego, detecta las minucias de la imagen y las escribe en un archivo. [...] Por defecto mindtct genera las minucias basadas en el estándar ANSI/NIST, [...] pero también tiene la opción de escribir las minucias de acuerdo al formato M1 (ANSI INCITS 378-2004). (NIST NBIS Userguide, 2019, p. 13)

Para extraer las minucias de una imagen de huella dactilar, la aplicación mindtct se usa con los siguientes argumentos:

Figura N° 3.3: Comando para extracción de minucias mediante la aplicación MINDTCT.

```
./mindtct -m1 fingerprint.jpg
```

Elaboración propia.

3.4.3.2. Evaluación de calidad

La evaluación de la calidad de las imágenes capturas es realizada por el componente NFIQ (Fingerprint Image Quality) del paquete NBIS (NIST Biometric Image Software).

NFIQ es una implementación del algoritmo de "Calidad de imagen de huella digital" del NIST [...] Calcula un vector de características utilizando un "mapa" de calidad de imagen y las estadísticas de calidad producidas por el algoritmo de detección de minucias MINDTCT. Luego, el vector de características se usa como entrada para un clasificador basado en una red neuronal de perceptrón multicapa (MLP), y el nivel de activación de salida de la red neuronal se usa para determinar el valor de

calidad de imagen de la huella digital. Hay cinco niveles de calidad, siendo 1 la calidad más alta y 5 la calidad más baja [...] no necesariamente asignará tipos de imágenes sin huellas dactilares al nivel cinco de peor calidad. (NIST NBIS Userguide, 2019, pp. 13-14)

Para determinar el índice de calidad de una imagen de huella dactilar, la aplicación nfiq se usa con los siguientes argumentos:

Figura N° 3.4: Comando para cálculo del índice de calidad mediante la aplicación NFIQ.

```
./nfiq fingerprint.jpg
```

FUENTE: NIST NBIS Userguide (2019).

3.4.3.3. Cálculo del puntaje de coincidencia

El cálculo del puntaje de coincidencia entre dos conjunto de minucias es realizada por el componente BOZORTH3 (Fingerprint Matcher) del paquete NBIS (NIST Biometric Image Software).

El algoritmo de comparación BOZORTH3 calcula una puntuación de coincidencia entre las minucias de cualquiera de dos huellas dactilares para ayudar a determinar si son del mismo dedo. [...] El comparador BOZORTH3 usa sólo la ubicación (x, y) y la orientación (theta) de las minucias para calcular el puntaje de coincidencia [...] es invariante a la rotación y traslación. (NIST NBIS-EC Userguide, 2019, p. 6).

La puntuación de coincidencia resultante de forma aproximada (pero no precisa) representa la cantidad de detalles que pueden combinarse entre las dos huellas dactilares. Como regla general, una puntuación de coincidencia de más de 40 generalmente indica una coincidencia verdadera. (NIST NBIS-EC Userguide, 2019, p. 21)

Para obtener el puntaje de coincidencia entre dos imágenes de huellas dactilares, la aplicación bozorth se usa con los siguientes argumentos:

Figura N° 3.5: Comando para el cálculo del puntaje de coincidencia mediante la aplicación BOZORTH.

```
./bozorth3 -m1 -b template1.xyt template2.xyt
```

FUENTE: NIST NBIS Userguide (2019).

Es necesario resaltar que este comparador da como salida un puntaje de coincidencia que se ha observado puede variar entre 0 y más de 1000. A fin de realizar la estandarización del puntaje se aplica la siguiente fórmula:

Figura N° 3.6: Fórmula para estandarización de datos.

$$u = \frac{x - \bar{x}}{s}$$

FUENTE: Veliz-Capuñay (2017).

A continuación el puntaje es normalizado en el intervalo [0-1] mediante la siguiente fórmula:

Figura N° 3.7: Fórmula para normalización softmax.

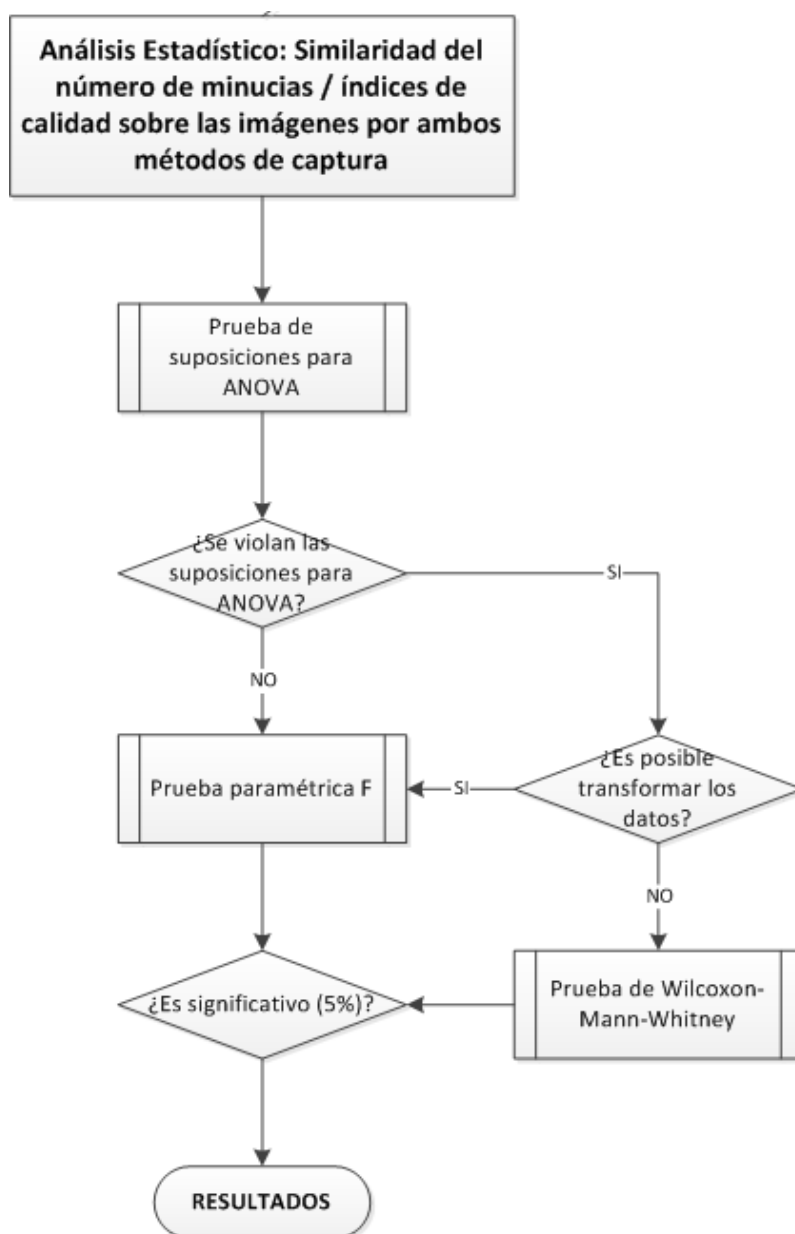
$$x^* = \frac{1}{1 + e^{-u}}$$

FUENTE: Veliz-Capuñay (2017).

3.4.4. Procedimiento de análisis estadístico

El procedimiento de análisis estadístico para determinar si existen diferencias en las características de la imagen de la huella (número de minucias y puntuación de calidad) obtenida por ambos métodos de captura es el siguiente:

Figura N° 3.8: Procedimiento de análisis estadístico.



Elaboración propia.

3.5. MATERIALES EMPLEADOS

3.5.1. Dispositivo biométrico ZKTECO Live20R

Este lector de huellas dactilares óptico permite la captura de imágenes a una resolución de 500dpi y es compatible con dispositivos Android que soporten OTG. Las especificaciones técnicas de este dispositivo se muestran en la Tabla N° 3.3.

Figura N° 3.9: Dispositivo biométrico ZKTECO Live20R.

FUENTE: Isecus (2019).

Tabla N° 3.3: Especificaciones técnicas del dispositivo ZKTECO Live20R.**Especificaciones técnicas dispositivo ZKTECO Live20R**

- Calidad de imagen: 2 millones de píxeles CMOS
- Detección de dedo vivo: Sí
- Formato de imagen: RAW, BMP, JPG
- Comunicación: USB 2.0 / USB1.1
- Tamaño de imagen: 300 x 400 píxeles (FAP20)
- Costo: \$50

FUENTE: Isecus (2019).

3.5.2. Dispositivo móvil Xiaomi Redmi 5 Plus

La cámara de este dispositivo móvil se usa para realizar la captura y el procesamiento de imágenes de huellas dactilares mediante una aplicación diseñada para este fin. Este dispositivo tiene las características que se muestran en la Tabla N° 3.4.

Figura N° 3.10: Dispositivo móvil Xiaomi Redmi 5 Plus.

FUENTE: Xiaomi (2019).

Tabla N° 3.4: Especificaciones técnicas del dispositivo Xiaomi Redmi 5 Plus.**Especificaciones técnicas Xiaomi Redmi 5 Plus**

- Diseño: 158,5 x 75,4 x 8,05 mm, 180 g, Lector de huellas trasero
- Pantalla: 5,99 pulgadas 18:9, 2.160 x 1.080 píxeles, 403 ppp, Brillo 450 nits
- Procesador: Snapdragon 625 Octa-core 2 GHz, Adreno 506
- RAM: 4 GB
- Capacidad: 64 GB
- Software: Android 8.1.0 + MIUI Global 10.2
- Cámara trasera: 12 MP, f/2.2, sensor 1.25µm, flash LED doble tono, PDAF, vídeos FullHD
- Cámara frontal: 5 MP, LED tipo antorcha, modo belleza, vídeos FullHD
- Conectividad: LTE Dual SIM, WiFi, Bluetooth 4.2, GPS-Glonass-Beidou, minijack, MicroUSB
- Batería: 4.000 mAh
- Costo: \$200

FUENTE: Xataka (2019).

3.5.3. Equipo de desarrollo

El equipo en el que se desarrollaron los prototipos y se llevó a cabo el análisis estadístico tiene las siguientes características:

- PC: Intel® Core(™) i5-7500 CPU @3.4GHz
- Memoria: RAM 16GB
- Tarjeta gráfica: NVIDIA GeForce GTX 1050 Ti
- Disco duro: SSD 256, HD 1TB
- Sistema operativo: Windows 10 Professional
- Sistema operativo: Linux Ubuntu 18.08

3.5.4. Entorno integrado de desarrollo

Para el desarrollo del primer prototipo se usó Python sobre la plataforma Jupyter Notebook (Anaconda). Anaconda es una distribución de código abierto para

computación científica para los lenguajes Python / R en Linux, Windows y Mac OS X, que tiene como objetivo simplificar la administración y el despliegue de paquetes (Anaconda Inc., 2019). Junto a Jupyter permite el desarrollo interactivo a través de una aplicación web en la que se visualizan los resultados inmediatamente en “notebooks” (Jupyter, 2019).

Para el desarrollo del segundo y tercer prototipo se usó Android Studio. Android Studio es el entorno de desarrollo integrado oficial para el sistema operativo Android de Google, construido sobre el software IntelliJ IDEA de JetBrains y diseñado específicamente para el desarrollo de Android (Android Studio, 2019).

3.5.5. OpenCV

Para la implementación del flujo de procesamiento de imágenes se usó la librería OpenCV. OpenCV es una librería optimizada para aplicaciones de visión computacional y de aprendizaje automático en tiempo real. OpenCV es de código abierto y es publicada bajo la Licencia BSD 3-Clause. Es gratis para uso comercial. OpenCV está escrito de forma nativa en C++ y tiene interfaces para C++, Python, Java y MATLAB y es compatible con Windows, Linux, Android y Mac OS. (OpenCV, 2019)

3.5.6. Software NBIS del NIST

Para la extracción de características de las imágenes de huellas dactilares se usó el software NBIS del NIST. El Software de Imagen Biométrica (NBIS) es desarrollado por el Instituto Nacional de Estándares y Tecnología de Estados Unidos (NIST), y puede ser distribuido libremente sin requisitos de licencia y se considera dominio público. (NIST NBIS, 2019). Para compilar estos paquetes en Windows se usaron MinGW y MSYS.

3.5.7. Software de análisis estadístico

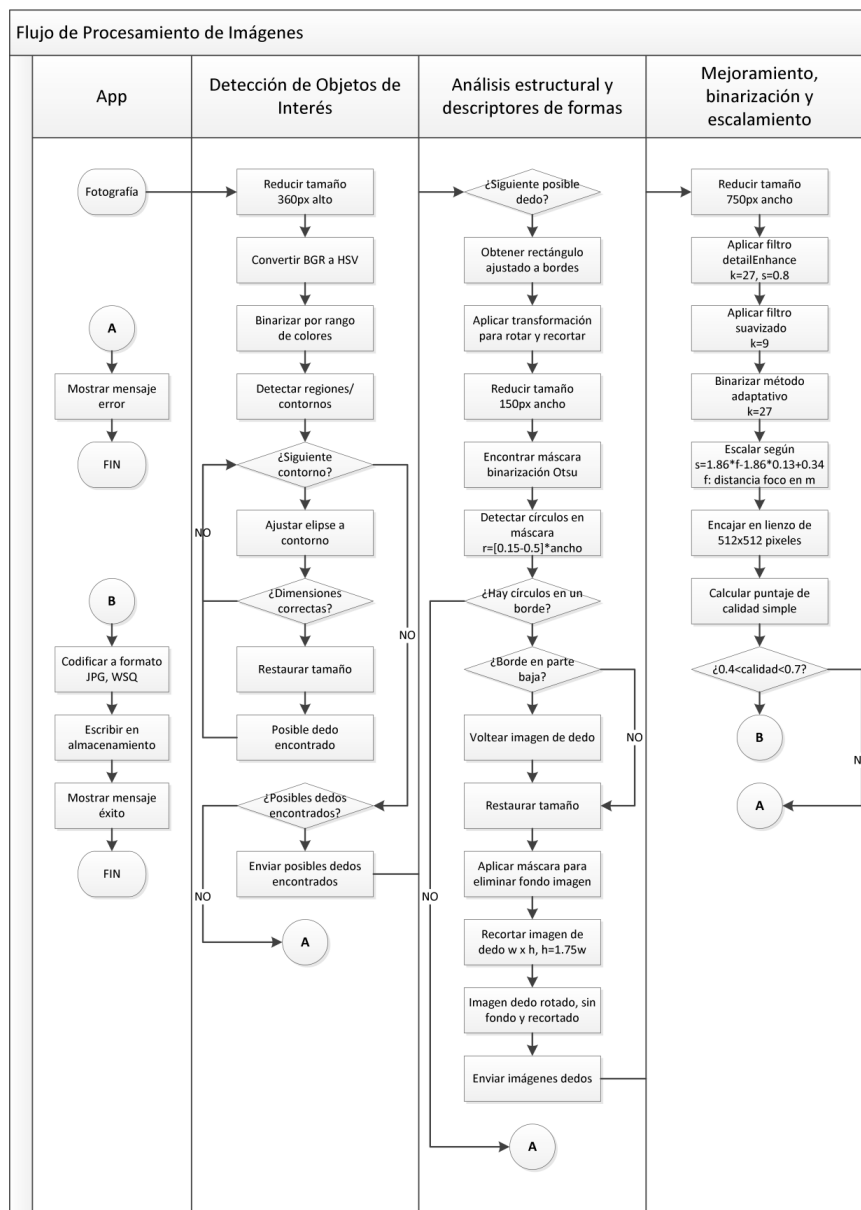
Para el análisis estadístico se usó el lenguaje R sobre la plataforma Jupyter Notebook (Anaconda). R es un lenguaje y entorno para computación estadística y para gráficos que proporciona una amplia variedad de técnicas estadísticas (modelado lineal y no lineal, pruebas estadísticas clásicas, análisis de series de tiempo, clasificación, agrupación, etc.) y técnicas gráficas. R está disponible como Software Libre bajo los términos de la Licencia Pública General GNU de la Free Software Foundation en forma de código fuente. Se compila y se ejecuta en una amplia variedad de plataformas UNIX y sistemas similares (incluyendo FreeBSD y Linux), Windows y MacOS. (R, 2019).

CAPÍTULO IV. RESULTADOS Y DISCUSIÓN

4.1. FLUJO DE PROCESAMIENTO DE IMÁGENES

El flujo de procesamiento de imágenes propuesto se aplica a todas las imágenes capturadas por la cámara, en esta investigación el término “método de captura por cámara” implica que la imagen capturada ha sido procesada según el flujo detallado en la Figura N° 4.1. El ANEXO B contiene mayor detalle de este flujo.

Figura N° 4.1: Flujo de procesamiento de imágenes propuesto.



Elaboración propia.

4.1.1. Detección del objeto de interés

El primer paso consiste en seleccionar el objeto de interés (dedo); es decir, detectar que la imagen de entrada contenga el objeto que se pretende procesar, para este propósito se suelen usar técnicas de detección de piel mediante al análisis de colores o el análisis de patrones (Sankaran, Malhotra, Mittal, Vatsa & Singh, 2015; Stein, Bouatou & Busch, 2013; Aoyama, Ito, Aoki & Ota, 2013; Kaur, Jain & Mittal, 2012), también se podrían usar técnicas de visión profunda como las redes completamente convolucionales para segmentación semántica o de instancia (Ronneberger, Fischer & Brox, 2015; Milletari, Navab & Ahmadi, 2016; Shelhamer, Long & Darrell, 2017; Chen, Zhu, Papandreou, Schroff & Adam, 2018).

Inicialmente se entrenó un modelo completamente convolucional basado en la arquitectura U-net (Ronneberger et al., 2015) para obtener las máscaras de segmentación que se usarían para detectar el objeto de interés. Se obtuvieron resultados prometedores; sin embargo, los requisitos para ejecutar inferencias sobre este modelo dificultan su implementación y uso en tiempo real (uso de memoria, tiempo en CPU, espacio en disco). En el ANEXO C se encuentra el detalle de este modelo, su arquitectura, el entrenamiento, algunas inferencias, así como el conjunto de datos de 40,138 imágenes generadas por el investigador para este propósito.

A fin de garantizar su uso en tiempo real se usaron técnicas de detección de piel mediante el análisis de colores. Se construyó un conjunto de datos de 5,388 histogramas de colores de imágenes de palmas, sobre el que se realizó un análisis de conglomerados a fin de determinar el número de grupos (perfiles de colores de piel). Luego, se usan los percentiles 5 y 95 de cada grupo para determinar el rango que se podría considerar como color de piel. Finalmente, la imagen binarizada se determina usando la ecuación representada en la Figura N° 4.2. En el ANEXO D se detalla el análisis de

conglomerados y el proceso de construcción del conjunto de datos de 5,388 histogramas de imágenes de palmas.

Figura N° 4.2: Fórmula para binarizar la imagen usando el color de la piel.

$$B = \begin{cases} 255, & 1 < h < 17 \text{ y } 18 < s < 103 \text{ y } 206 < v < 251, \text{ o} \\ & 3 < h < 17 \text{ y } 36 < s < 119 \text{ y } 165 < v < 228, \text{ o} \\ & 2 < h < 18 \text{ y } 55 < s < 159 \text{ y } 129 < v < 195, \text{ o} \\ & 0 < h < 13 \text{ y } 20 < s < 111 \text{ y } 184 < v < 250 \\ 0, & \text{otro caso} \end{cases}$$

Elaboración propia.

La fórmula presentada en la Figura N° 4.2 se aplica sobre la imagen original reducida a un alto de 360 pixeles mediante la función `inRange` de OpenCV. A la imagen resultado de este proceso se le realizan operaciones morfológicas (dilación y erosión, $k=7$ y $k=3$) para eliminar regiones pequeñas. La Figura N° 4.3 muestra la imagen de entrada y la imagen binarizada encontrada mediante este proceso.

Figura N° 4.3: Imagen de entrada e imagen binarizada luego del análisis de color.



Elaboración propia.

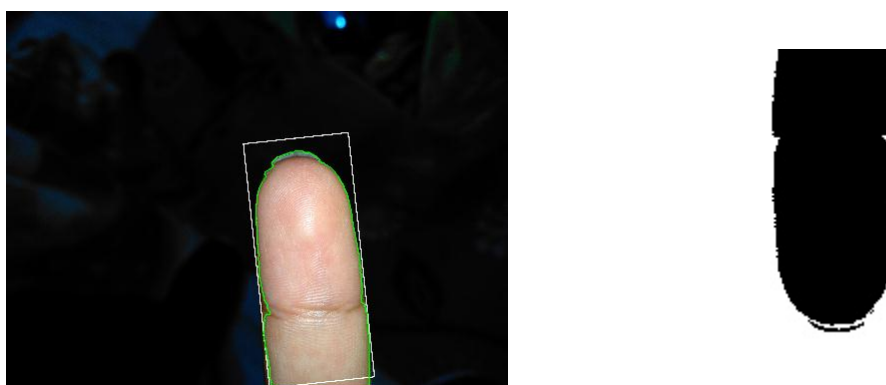
A la imagen binarizada se le realiza una detección de contornos mediante la función `findContours` (Suzuki, 1985) implementada en OpenCV. A cada contorno encontrado se le ajusta una elipse mediante la función `fitEllipse` (Fitzgibbon & Fisher, 1995), las dimensiones de esta elipse son evaluadas en sus características estructurales:

ratio, área y dimensiones. Cada elipse que, por sus características estructurales, se asemeje a un dedo, es considerada como un “posible dedo”.

4.1.2. Análisis estructural y descriptores de formas

Cada región que se considera un “posible dedo” es ajustada y rotada para que termine de forma vertical. Esta imagen rotada es reducida temporalmente a un ancho de 150 píxeles con el objetivo de encontrar la máscara de segmentación y de detectar la punta del dedo. La máscara de segmentación se encuentra realizando una binarización mediante el algoritmo de Otsu, 1979. La detección de la punta del dedo se realiza ajustando al menos dos círculos de distintos radios (de 0.15 a 0.5 del ancho del dedo) a la imagen binarizada, se confirma la punta del dedo si los círculos encontrados son cercanos y se encuentran en uno de los bordes. La detección de círculos se realiza usando la transformada de Hough mediante la función `HoughCircles` de OpenCV, esta operación es computacionalmente intensiva así que es adecuado trabajar sobre la máscara de segmentación en tamaño reducido.

Figura N° 4.4: Imagen con región del dedo resaltada y rotada.



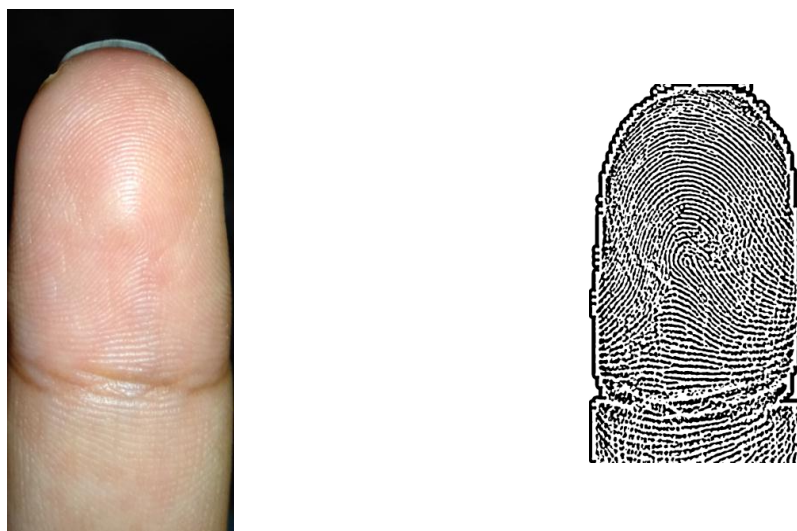
Elaboración propia.

La imagen es rotada si la punta del dedo se encuentra en la parte inferior. Luego, el fondo de la imagen es eliminado usando la máscara de segmentación encontrada. Al finalizar este proceso, se cuenta con la imagen del dedo rotada, recortada y sin fondo.

4.1.3. Recorte, mejoramiento y binarización

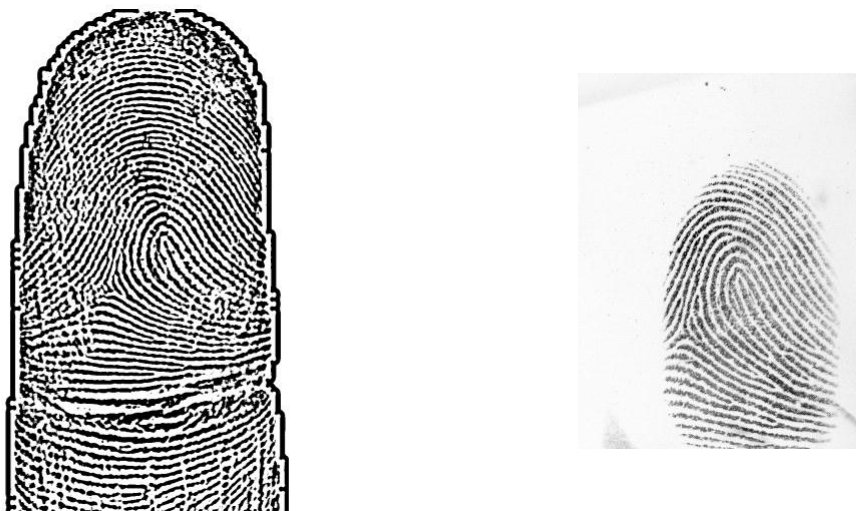
La imagen del dedo es escalada a un ancho de 750 píxeles para luego ser mejorada y binarizada. El mejoramiento se realiza usando un filtro en el dominio del espacio que preserve los bordes y realce los detalles, para lo cual se usa la función `detailEnhance` (Gastal & Oliveira, 2011) de OpenCV ($k=27$, $s=0.8$), a este resultado se le pasa un filtro de suavizado ($k=9$). A continuación, la imagen resultante se pasa por un algoritmo de binarización adaptativa con una ventana de 27 píxeles.

Figura N° 4.5: Imagen con región del dedo recortada y mejorada.



Elaboración propia.

La imagen resultante debe ser escalada a una resolución uniforme, ya que el mismo dedo puede tener diferentes tamaños cuanto más lejos o cerca se encuentre del foco de la cámara. El escalamiento a una resolución uniforme es necesario ya que es un requisito fundamental de los comparadores basados en minucias que la imagen de entrada tenga una resolución de 500dpi. A fin de determinar la resolución adecuada para la imagen final se ajustó un modelo lineal que determina el ratio de escalamiento aproximado que se debe aplicar según la distancia del foco de la cámara al dedo, esta distancia es calculada por el dispositivo.

Figura N° 4.6: Captura por cámara y captura por sensor (ratio 1:2).

Elaboración propia.

La imagen resultante es analizada para determinar su calidad, la cual está representada como el resultado de la fórmula de la Figura N° 4.7. La imagen se acepta sólo si está en el rango de 0.4 a 0.7. Finalmente, si la imagen cumple con el criterio de calidad, ésta es centrada en un lienzo de 512x512 píxeles con fondo blanco.

Figura N° 4.7: Fórmula para calcular un puntaje de calidad de una huella capturada por cámara.

$$score = \frac{total\ de\ pixeles\ en\ negro}{area}$$

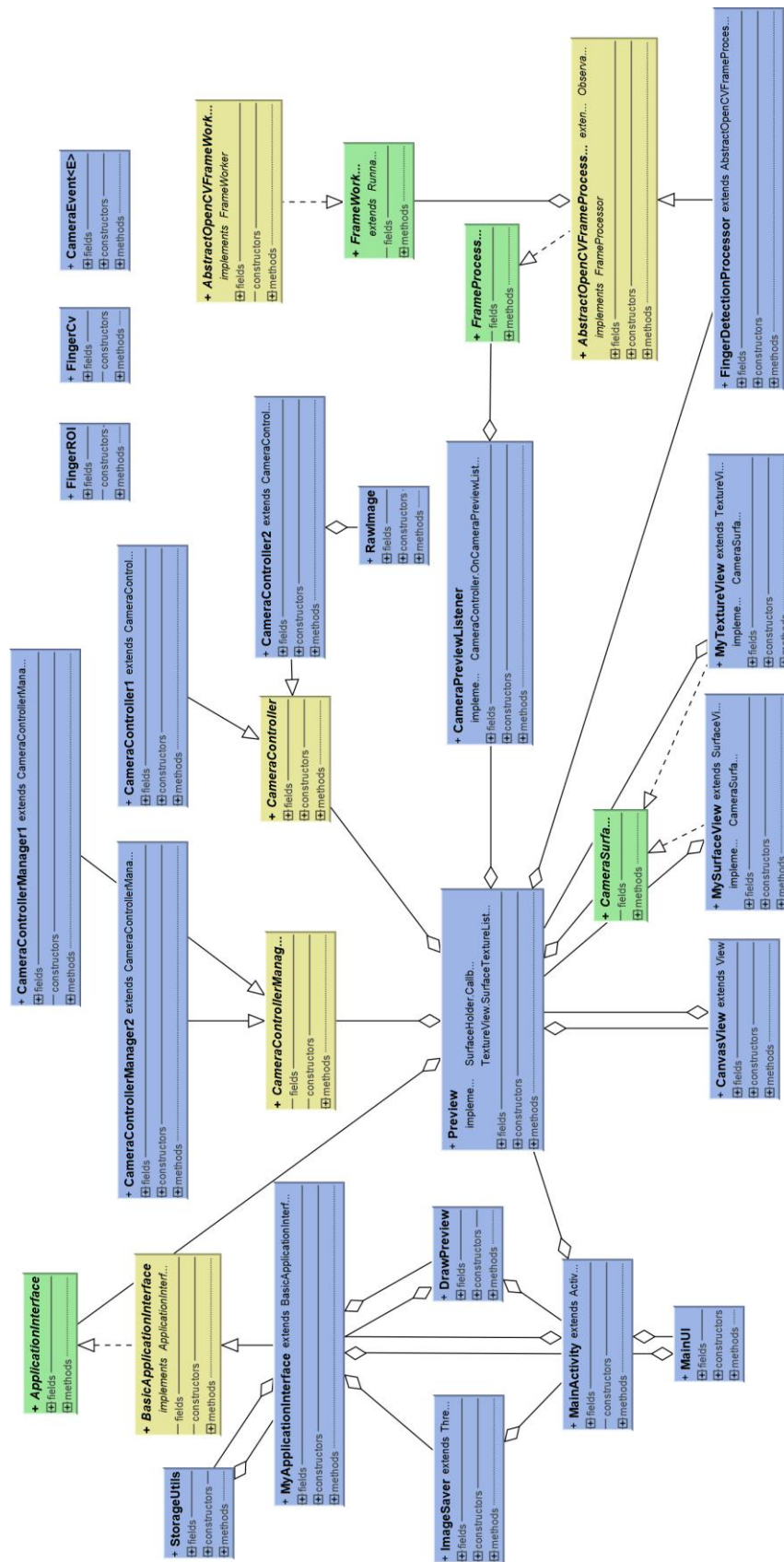
Elaboración propia.

4.2. IMPLEMENTACIÓN DE LA APLICACIÓN MÓVIL

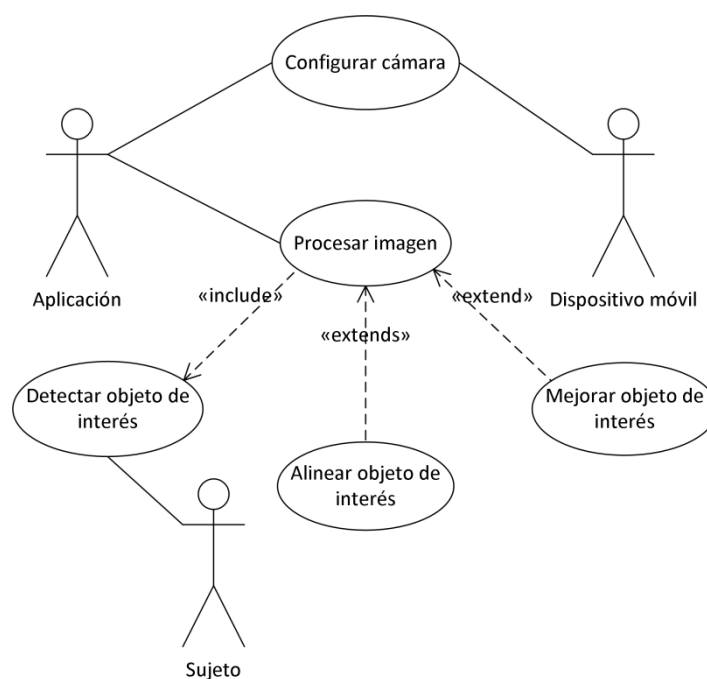
4.2.1. Especificación de requerimientos de software

El proceso de implementación de la aplicación móvil se realizó usando el modelo de desarrollo evolutivo (Pressman & Maxim, 2014), el cual se caracteriza por ser iterativo y por construir versiones cada vez más completas. La Especificación de Requerimientos de Software (ERS) se encuentra detallada en el ANEXO A. A continuación se muestra los diagramas de casos de uso y de clases de la aplicación:

Figura N° 4.8: Diagrama de clases de diseño de la aplicación móvil.



Elaboración propia.

Figura N° 4.9: Diagrama de casos de uso de la aplicación móvil.

Elaboración propia.

Los requisitos y la descripción de casos de uso, así como otros diagramas se encuentran en la Especificación de Requerimientos de Software detallada en el ANEXO A.

4.2.2. Prototipos de la aplicación móvil

El primer prototipo fue desarrollado usando OpenCV en Python en un cuaderno de Jupyter Notebook que lee la imagen desde un archivo, realiza el procesamiento y retorna una imagen procesada del mismo tamaño.

El segundo prototipo fue desarrollado usando OpenCV en Java para Android a partir del proyecto de código abierto CAMdroid (Hschott, 2019). Este prototipo evalúa cada fotograma para detectar el dedo y enfocarlo. El fotograma es procesado, y la imagen del dedo es recortada, rotada, escalada y binarizada a un tamaño predefinido.

El tercer prototipo fue desarrollado usando OpenCV en Java para Android a partir del proyecto de código abierto OpenCamera (2019). Este prototipo evalúa cada

fotograma para detectar el dedo, enfocarlo y tomar una fotografía. La fotografía es procesada y la imagen del dedo es recortada, rotada y escalada según el ratio calculado por el modelo lineal en base a la distancia del foco al dedo. Esta imagen recortada es mejorada y binarizada. El flujo de procesamiento de imágenes usado por esta aplicación se detalla en el punto 4.1 de esta investigación.

Figura N° 4.10: Captura de pantalla de la aplicación móvil.



Elaboración propia.

4.3. DIFERENCIAS EN LAS CARACTERÍSTICAS DE LA IMAGEN DE LA HUELLA OBTENIDA POR AMBOS MÉTODOS DE CAPTURA

4.3.1. Diferencias en el número de minucias

Para este análisis se construyó un conjunto de datos de 505 registros, cada registro es obtenido de una imagen y tiene los campos: “minutiae”, que representa el número de minucias, y “group”, que representa el método de captura, esto se aprecia en la Tabla N° 4.1. El ANEXO E detalla el proceso de elaboración de este conjunto de datos.

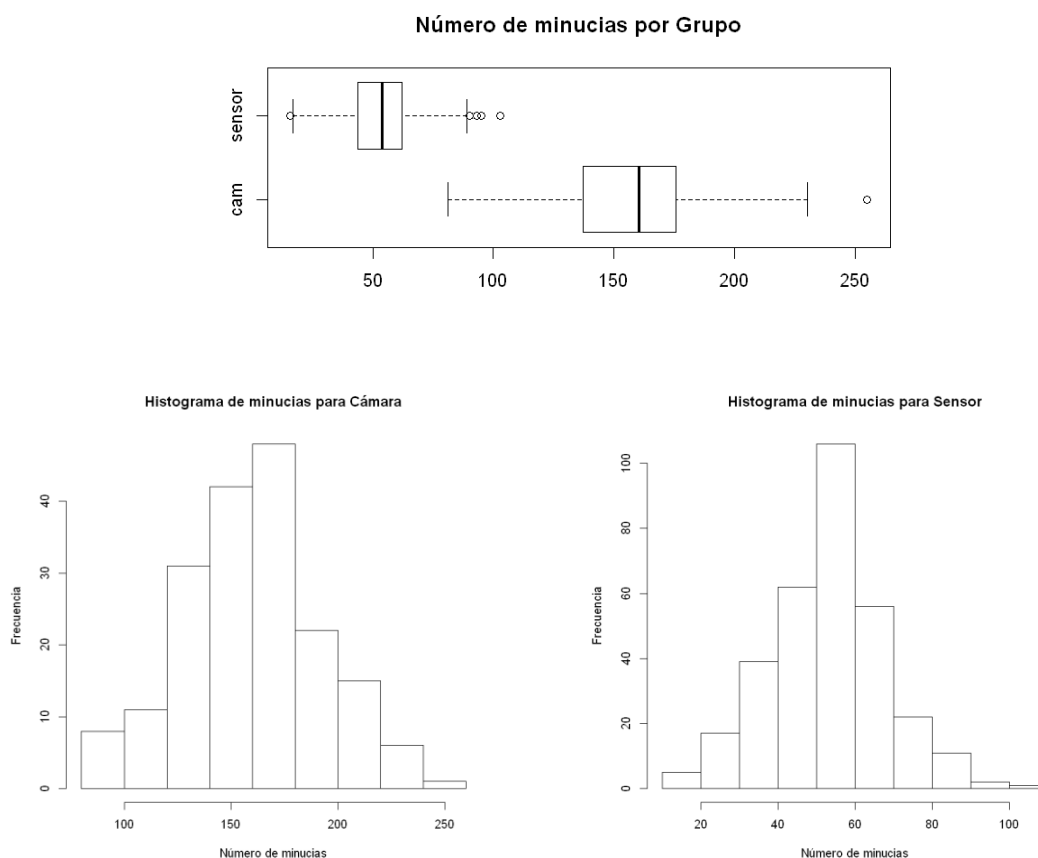
Tabla N° 4.1: Estructura del conjunto de datos de números de minucias.

group	minutiae
cam	151
...	...

Elaboración propia.

Los estadísticos descriptivos de este conjunto de datos se muestran en la Tabla N° 4.2, se aprecia que la media del número de minucias de las imágenes capturadas con la cámara es aproximadamente 3 veces mayor a la media del número de minucias de las imágenes capturadas con el sensor. Esto se confirma en el gráfico de cajas por método de captura y en los histogramas que se muestran en la Figura N° 4.11.

Figura N° 4.11: Diagrama de cajas del número de minucias agrupado por método de captura. Histogramas para el número de minucias por cada método de captura.



Elaboración propia.

Tabla N° 4.2: Estadísticos descriptivos conjunto de datos de número de minucias.

Grupo	Número	Media	Mediana	Desviación Estándar
cam	184	159.59	160.5	32.61
sensor	321	53.49	54.0	15.02

Elaboración propia.

Se plantea la pregunta *¿La media del número de minucias del grupo Sensor (m_s) es igual a la media del número de minucias del grupo Cámara (m_c)?*. Entonces, las hipótesis estadísticas correspondientes son:

$$H_0: m_s = m_c$$

$$H_a: m_s \neq m_c$$

Antes de realizar la prueba ANOVA se verifican los supuestos de esta prueba respondiendo a las siguientes preguntas:

1. ¿Proviene las muestras de grupos independientes?

Sí.

2. ¿Los datos de cada uno de los dos grupos siguen una distribución normal? Para verificar el supuesto de normalidad se plantea la hipótesis:

H_0 : Los datos están distribuidos de forma normal.

H_a : Los datos no están distribuidos de forma normal.

Ya que $n > 30$ se usa la prueba estadística LKS (Lilliefors-Kolmogorov-Smirnov) para cada grupo de muestras.

Figura N° 4.12: Resultados de la prueba de normalidad LKS para el número de minucias del grupo cámara.

Lilliefors (Kolmogorov-Smirnov) normality test data: norm D = 0.072179, p-value = 0.0207
--

Elaboración propia.

Figura N° 4.13: Resultados de la prueba de normalidad LKS para el número de minucias del grupo sensor.

```
Lilliefors (Kolmogorov-Smirnov) normality test
data: norm
D = 0.050842, p-value = 0.04434
```

Elaboración propia.

En la Figura N° 4.12 (método de captura usando cámara) y la Figura N° 4.13 (método de captura usando sensor) se puede apreciar que el p-valor es menor que el nivel de significancia 0.05 por lo que se rechaza la hipótesis nula y se acepta la hipótesis alterna: los datos no siguen una distribución normal.

3. ¿Las poblaciones tienen la misma varianza?

Para verificar el supuesto de homogeneidad de varianzas se usa la prueba F:

Figura N° 4.14: Resultados de la prueba F de homogeneidad de varianzas para el número de minucias de los grupos: cámara y sensor.

```
F test to compare two variances
data: minutiae by group
F = 4.7141, num df = 183, denom df = 320, p-value < 2.2e-16
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 3.660402 6.126846
sample estimates:
ratio of variances
 4.714072
```

Elaboración propia.

En la Figura N° 4.14 se puede apreciar que el p-valor es menor que el nivel de significancia 0.05. Se concluye que existen diferencias significativas entre las varianzas de los dos conjuntos de datos.

Dado que no se cumplen los supuestos de normalidad y homocedasticidad para la prueba ANOVA, se procede a usar la prueba no paramétrica de Wilcoxon-Mann-Whitney.

Figura N° 4.15: Resultados de la prueba bilateral no paramétrica de Wilcoxon-Mann-Whitney para el número de minucias de los grupos: cámara y sensor.

```

Wilcoxon rank sum test with continuity correction

data: minutiae by group
W = 59019, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0
  
```

Elaboración propia.

Interpretación. Como se puede apreciar en la Figura N° 4.15 la prueba devuelve un p-valor que es menor al nivel de significancia de 0.05, entonces se rechaza la hipótesis nula y se concluye que: la media del número de minucias del grupo Sensor es significativamente diferente a la media del número de minucias del grupo Cámara.

A fin de determinar si la media del número de minucias del grupo Sensor es mayor a la media del número de minucias del grupo Cámara se realiza la prueba:

Figura N° 4.16: Resultados de la prueba unilateral no paramétrica de Wilcoxon-Mann-Whitney para el número de minucias de los grupos: cámara y sensor.

```

Wilcoxon rank sum test with continuity correction

data: minutiae by group
W = 59019, p-value = 1
alternative hypothesis: true location shift is less than 0
  
```

Elaboración propia.

Interpretación. Como se puede apreciar en la Figura N° 4.16 la prueba devuelve un p-valor que es mayor al nivel de significancia de 0.05, entonces se acepta la hipótesis nula y se concluye que: la media del número de minucias del grupo Cámara es mayor a la media del número de minucias del grupo Sensor.

4.3.2. Diferencias en el índice de calidad

Para este análisis se construyó un conjunto de datos de 505 registros, cada registro es obtenido de una imagen y tiene los campos: “nfiq”, que representa el índice de calidad, y “group”, que representa el método de captura, esto se aprecia en la Tabla N° 4.3. El ANEXO F detalla el proceso de elaboración de este conjunto de datos.

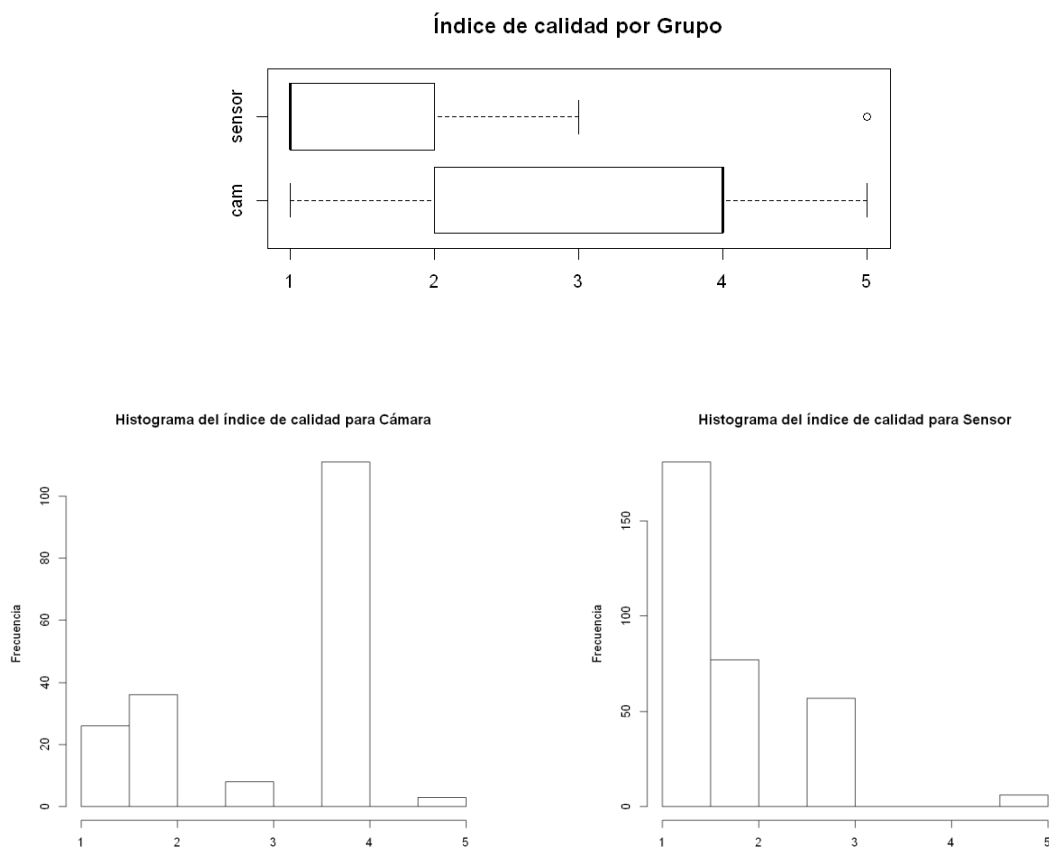
Tabla N° 4.3: Estructura del conjunto de datos de índices de calidad.

group	nfiq
cam	2
...	...

Elaboración propia.

Los estadísticos descriptivos de este conjunto de datos se muestran en la Tabla N° 4.4, se aprecia que la media del índice de calidad de las imágenes capturadas con la cámara es mayor a la media del índice de calidad de las imágenes capturadas con el sensor. Esto se confirma en el gráfico de cajas por método de captura y en los histogramas que se muestran en la Figura N° 4.17.

Figura N° 4.17: Diagrama de cajas del índice de calidad agrupado por método de captura. Histogramas para el índice de calidad por cada método de captura.



Elaboración propia.

Tabla N° 4.4: Estadísticos descriptivos del conjunto de datos de índices de calidad.

Grupo	Número	Media	Mediana	Desviación Estándar
cam	184	3.16	4	1.19
sensor	321	1.67	1	0.90

Elaboración propia.

Se plantea la pregunta *¿La media del índice de calidad del grupo Sensor (m_s) es igual a la media del índice de calidad del grupo Cámara (m_c)?*. Entonces, las hipótesis estadísticas correspondientes son:

$$H_0: m_s = m_c$$

$$H_a: m_s \neq m_c$$

Antes de realizar la prueba ANOVA se verifican los supuestos de esta prueba respondiendo a las siguientes preguntas:

1. ¿Proviene la muestra de grupos independientes?

Sí.

2. ¿Los datos de cada uno de los dos grupos siguen una distribución normal? Para verificar el supuesto de normalidad se plantea la hipótesis:

$$H_0: \text{Los datos están distribuidos de forma normal.}$$

$$H_a: \text{Los datos no están distribuidos de forma normal.}$$

Ya que $n > 30$ se usa la prueba estadística LKS (Lilliefors-Kolmogorov-Smirnov) para cada grupo de muestras.

Figura N° 4.18: Resultados de la prueba de normalidad LKS para el índice de calidad del grupo cámara.

<p>Lilliefors (Kolmogorov-Smirnov) normality test</p> <p>data: norm</p> <p>D = 0.38036, p-value < 2.2e-16</p>
--

Elaboración propia.

Figura N° 4.19: Resultados de la prueba de normalidad LKS para el índice de calidad del grupo sensor.

```
Lilliefors (Kolmogorov-Smirnov) normality test
data: norm
D = 0.33645, p-value < 2.2e-16
```

Elaboración propia.

En la Figura N° 4.18 (método de captura usando cámara) y la Figura N° 4.19 (método de captura usando sensor) se puede apreciar que el p-valor es menor que el nivel de significancia 0.05 por lo que se rechaza la hipótesis nula y se acepta la hipótesis alterna: los datos no siguen una distribución normal.

3. ¿Las poblaciones tienen la misma varianza?

Para verificar el supuesto de homogeneidad de varianzas se usa la prueba F:

Figura N° 4.20: Resultados de la prueba F de homogeneidad de varianzas para el índice de calidad de los grupos: cámara y sensor.

```
F test to compare two variances
data: minutiae by group
F = 4.7141, num df = 183, denom df = 320, p-value < 2.2e-16
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 3.660402 6.126846
sample estimates:
ratio of variances
 4.714072
```

Elaboración propia.

En la Figura N° 4.20 se puede apreciar que el p-valor es menor que el nivel de significancia 0.05. Se concluye que existen diferencias significativas entre las varianzas de los dos conjuntos de datos.

Dado que no se cumplen las suposiciones de normalidad y homocedasticidad para la prueba ANOVA. Se procede a usar la prueba no paramétrica de Wilcoxon-Mann-Whitney.

Figura N° 4.21: Resultados de la prueba bilateral no paramétrica de Wilcoxon-Mann-Whitney para el índice de calidad de los grupos: cámara y sensor.

```

Wilcoxon rank sum test with continuity correction

data:  nfiq by group
W = 48466, p-value < 2.2e-16
alternative hypothesis: true location shift is not equal to 0

```

Elaboración propia.

Interpretación. Como se puede apreciar en la Figura N° 4.21 la prueba devuelve un p-valor que es menor al nivel de significancia de 0.05, entonces se rechaza la hipótesis nula y se concluye que: la media del índice de calidad del grupo Sensor es significativamente diferente a la media del índice de calidad del grupo Cámara.

A fin de determinar si la media del índice de calidad del grupo Cámara *es mayor* a la media del índice de calidad del grupo Sensor se realiza la prueba:

Figura N° 4.22: Resultados de la prueba unilateral no paramétrica de Wilcoxon-Mann-Whitney para el índice de calidad de los grupos: cámara y sensor.

```

Wilcoxon rank sum test with continuity correction

data:  nfiq by group
W = 48466, p-value = 1
alternative hypothesis: true location shift is less than 0

```

Elaboración propia.

Interpretación. Como se puede apreciar en la Figura N° 4.22 la prueba devuelve un p-valor que es mayor al nivel de significancia de 0.05, entonces se acepta la hipótesis nula y se concluye que: la media del índice de calidad del grupo Cámara es mayor a la media del índice de calidad del grupo Sensor. **Es necesario indicar que en el índice de calidad NFIQ, valores más grandes indican menor calidad de imagen.**

4.4. RENDIMIENTO DE UN SISTEMA BIOMÉTRICO

Para este análisis se construyó un conjunto de datos de 195,961 registros, cada registro es obtenido de la comparación de dos imágenes de huellas y tiene los campos: “group”, que representa el método de captura de cada huella (cámara-cámara, cámara-

sensor, sensor-sensor), “truth_n”, donde 1 significa que ambas huellas son de la misma persona y 0 que son de personas distintas, “score”, que es el puntaje de comparación en crudo, y, “score_norm2”, que representa el puntaje de comparación normalizado en el rango [0-1] según las ecuaciones de la Figura N° 3.6 y Figura N° 3.7. El ANEXO G detalla el proceso de elaboración de este conjunto de datos.

Tabla N° 4.5: Estructura del conjunto de datos de puntajes de comparación.

group	truth_n	score	score_norm2
cam-sensor	1	36	0.94
...

Elaboración propia.

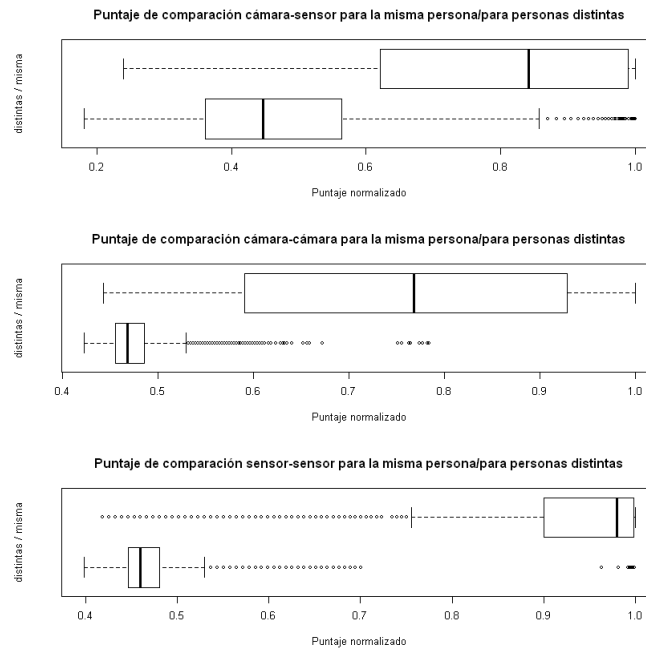
Los estadísticos descriptivos de este conjunto de datos, cuando se agrupan por huellas que pertenecen a la misma persona y huellas que pertenecen a distintas personas, se muestran en la Tabla N° 4.6, se aprecia que la media del puntaje de comparación es mayor en el grupo cámara-cámara (aún cuando pertenecen a personas distintas). La Figura N° 4.23 muestra los gráficos de cajas de los puntajes de comparación normalizados por grupos en la que se puede apreciar que los puntajes genuinos (ambas huellas provienen de la misma persona) e impostores (las huellas provienen de distintas personas) para el grupo cámara-sensor son menos discriminables que para el grupo sensor-sensor.

Tabla N° 4.6: Estadísticos descriptivos de los puntajes de comparación cuando las dos huellas pertenecen a la misma persona o a personas distintas.

Persona	Grupo	Número	Media	Mediana	Desv. Estd.
Distintas personas	cam-cam	32,644	28.05	26	9.81
	cam-sensor	57,632	12.17	11	5.52
	sensor-sensor	100,390	10.21	9	5.35
Mismas personas	cam-cam	1,212	320.53	156	427.92
	cam-sensor	1,432	39.50	27	31.86
	sensor-sensor	2,651	185.80	154	134.81

Elaboración propia.

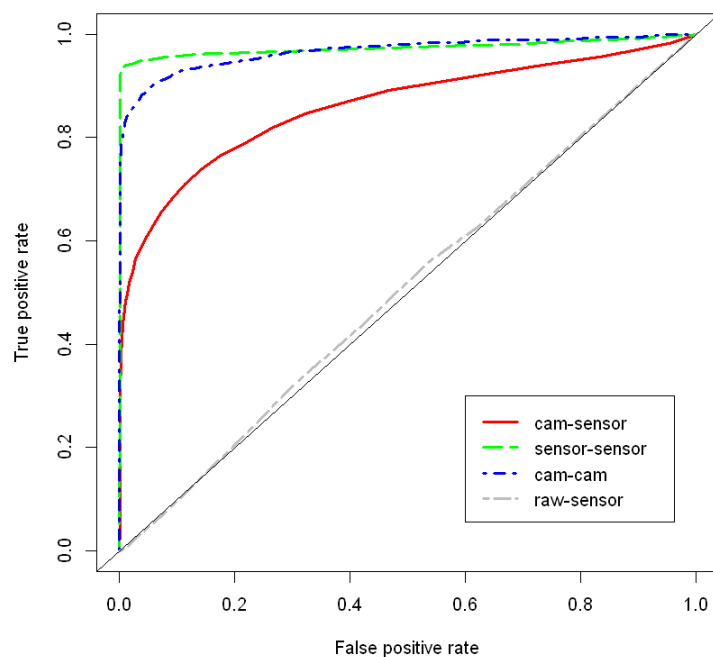
Figura N° 4.23: Diagramas de cajas del puntaje de comparación para los grupos: cámara-sensor, sensor-cámara, sensor-sensor cuando las dos huellas pertenecen a la misma persona o a personas distintas.



Elaboración propia.

A fin de determinar el rendimiento del sistema biométrico cuando se usan diferentes métodos de captura de imágenes (cámara-cámara, cámara-sensor, sensor-sensor) se realiza un análisis de las curvas ROC de los clasificadores de los puntajes de comparación para cada par de métodos. Asimismo, se incluye el método de captura “raw” que agrupa las fotografías sin ningún tipo de procesamiento (sólo conversión a escala de grises). En la Figura N° 4.24 se aprecia que el rendimiento del clasificador para los puntajes de comparación de imágenes capturadas con el mismo método (cámara-cámara, sensor-sensor) es mucho mejor que el clasificador para los puntajes de comparación para imágenes capturadas con distintos métodos (cámara-sensor). También se aprecia que el clasificador para los puntajes de comparación en los que una de las imágenes no tienen ningún tipo de procesamiento (raw-sensor) es casi el de un valor aleatorio, se encuentra cerca a la línea diagonal.

Figura N° 4.24: Curvas ROC cuando se comparan imágenes capturadas por distintos métodos: cámara-sensor (cam-sensor), sensor-sensor (sensor-sensor), cámara-cámara (cam-cam) e imagen sin procesamiento-sensor (raw-sensor).



Elaboración propia.

Como se ha visto en el marco teórico el rendimiento de un clasificador dependerá del umbral que se seleccione para determinar una coincidencia. Para aplicaciones críticas se preferirá mantener el ratio de falsos positivos (FPR) bajo, para aplicaciones civiles este umbral puede incrementarse. Seleccionando un ratio de falsos positivos del 2%, es decir, que 2 de cada 100 sujetos que no son quien dicen ser podrían ser aceptados por el sistema, obtenemos un ratio de verdaderos positivos del 51.75% cuando se comparan imágenes capturadas por la cámara contra imágenes capturadas por el sensor, lo que significa que 51 de cada 100 sujetos que son quien dicen ser serían aceptados por el sistema. Con el mismo FPR se obtiene un TPR de 94.19% para las comparaciones entre imágenes capturadas por sensor-sensor y 85.39% para las comparaciones entre imágenes capturadas por cámara-cámara.

CONCLUSIONES

PRIMERA

Sí existen diferencias en las características de la imagen de la huella obtenida por un nuevo método de captura de huellas dactilares a diferencia de la imagen de la huella obtenida por un sensor biométrico. El número de minucias de una imagen de huella obtenida por el método de captura propuesto es mayor, ya que el nuevo método genera una mayor superficie de captura; sin embargo, la calidad de la imagen disminuye.

SEGUNDA

Se ha implementado un flujo de procesamiento de imágenes que es aplicado en tiempo real a las imágenes capturadas mediante la cámara de un dispositivo móvil para obtener una imagen de huella dactilar compatible con un sistema biométrico de reconocimiento dactilar basado en minucias, este flujo debe considerar las restricciones de memoria y procesamiento de estos dispositivos y consta de tres etapas: en la primera etapa se usa el análisis del dominio de los colores para la detección del objeto de interés, en la segunda etapa, se realiza un análisis estructural para recortar y girar la imagen del dedo, en la última etapa, se mejora y binariza la imagen para finalmente ser escalada a una resolución uniforme.

TERCERA

Se ha implementado una aplicación móvil que usa el flujo de procesamiento de imágenes propuesto para obtener la imagen de la huella dactilar a partir del dedo enfocado por la cámara del dispositivo. La metodología de desarrollo permite una construcción iterativa e incremental y ha sido dividida en varios prototipos, el primer prototipo fue desarrollado en Python y sólo implementó el flujo de procesamiento inicial, el segundo prototipo es la primera versión de la aplicación móvil y está basada

en el proyecto de código abierto CAMdroid, y el prototipo final es la aplicación móvil actual y está basada en el proyecto de código abierto OpenCamera.

CUARTA

El rendimiento de un sistema biométrico de reconocimiento dactilar cuando se comparan imágenes de las huellas obtenidas por un nuevo método de captura contra las imágenes de las huellas obtenidas por un sensor biométrico (cámara-sensor) medido a través del ratio de verdadero positivo (TPR) es del 51.75%; cuando ambas imágenes se obtienen de la misma fuente el TPR es del 94.19% (sensor-sensor) y del 85.39% (cámara-cámara); en todos los casos se usa un ratio de falso positivo (FPR) del 2%, este umbral puede determinarse usando un análisis del clasificador mediante curvas ROC, y debería ajustarse según el tipo de aplicación. Por motivos de comparación se obtiene un TPR del 1.81% cuando se comparan imágenes de las huellas sin ningún tipo de procesamiento contra las imágenes de las huellas obtenidas por un sensor biométrico (raw-sensor).

RECOMENDACIONES

PRIMERA

La aplicación de técnicas de aprendizaje supervisado ayudaría a mejorar las características de la imagen de la huella obtenida por el método de captura propuesto en esta investigación. Así mismo, estas técnicas nos darían la posibilidad de generar máscaras de segmentación, predecir la ubicación de las minucias, predecir las regiones de interés, predecir el dedo usado, y el género, edad y ocupación de la persona.

SEGUNDA

El flujo de procesamiento de imágenes propuesto en esta investigación sólo permite la captura de una huella dactilar por fotografía; sin embargo, se podrían capturar hasta 4 imágenes de huellas dactilares a la vez. Así mismo, este flujo podría ser usado para etiquetar un conjunto de datos que serviría como entrada para la aplicación de técnicas de aprendizaje supervisado.

TERCERA

En dispositivos móviles con cámaras más sofisticadas se podría explorar la captura de características de nivel 3 a partir de las imágenes de los dedos. También se podría explorar la aplicación de técnicas de captura o generación de modelos tridimensionales de los dedos.

CUARTA

El ratio de verdadero positivo (TPR) encontrado en esta investigación puede mejorarse sustancialmente usando una fusión de puntajes de comparación, es decir, usando 4 huellas a la vez para obtener un solo puntaje de comparación. A este puntaje de comparación también se le podría fusionar el puntaje de comparación de la biometría de la palma y de la geometría de la mano.

REFERENCIAS BIBLIOGRÁFICAS

- Afifi, M. (2019). 11K Hands: gender recognition and biometric identification using a large dataset of hand images. *Multimedia Tools and Applications*.
doi:10.1007/s11042-019-7424-8
- Anaconda Inc. (2019). *Anaconda Distribution. The World's Most Popular Python/R Data Science Platform*. Recuperado el 19 de Junio del 2019 de <https://www.anaconda.com/distribution/>
- Android Studio (2019). *Features overview for Android Studio*. Recuperado el 19 de Junio del 2019 de <https://developer.android.com/studio/features>
- ANSI NISTC (2006). *Biometrics Glossary (BG)*. Recuperado el 19 de Junio del 2019 de <https://www.hSDL.org/?view&did=464490>.
- Aoyama, S., Ito, K., Aoki, T. & Ota, H. (2013). A Contactless Palmprint Recognition Algorithm for Mobile Phones. *International Workshop on Advanced Image Technology*, 409-413.
- Buciu, I. & Gacsadi, A. (2016). Biometrics Systems and Technologies: A survey. *International Journal of Computers, Communications & Control*, 11(3), 315-330.
- CERIASPURDUE (10 de Mayo 2012). *2008-09-17 CERIAS Fingerprint Sensor Interoperability: Analysis of Error Rates for Fingerprint*. [Archivo de Video]. Recuperado el 19 de Junio del 2019 de <https://youtu.be/BBSmZ2Z4s2E>.
- Chambi Mamani, E. W. (2016). *Reconocimiento y detección biométrica basada en imágenes de huellas digitales*. (Tesis para optar el grado académico de Magister Scientiae en Informática). Universidad Nacional del Altiplano, Puno, Perú.
- Chen, L., Zhu, Y., Papandreou, G., Schroff, F. & Adam, H. (2018). Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *CoRR*, 2018, abs/1802.02611.

- Deb, D., Chugh, T., Engelsma, J. J., Cao, K., Nain, N., Kendall, J. & Jain, A. K. (2018). Matching Fingerphotos to Slap Fingerprint Images. *CoRR*, 2018, abs/1804.08122.
- Derawi, M. O., Yang, B. & Busch, C. (2012). Fingerprint Recognition with Embedded Cameras on Mobile Phones. *Security and Communications Networks*, 2012, 1-12. doi:10.1002/sec
- Dongjae, L., Kyoungtaek, C., Heeseung, C. & Jaihie, K. (2008). Recognizable-Image Selection for Fingerprint Recognition with a Mobile-Device Camera. *IEEE Transactions on Systems, Man & Cybernetics: Part B*, 38(1), 233-243. doi:10.1109/TSMCB.2007.908869
- Faulds, H. (2015). *Dactylography or the Study of Fingerprints*. Recuperado el 19 de Junio del 2019 de <https://www.gutenberg.org/ebooks/47911>. (Trabajo original publicado en 1912)
- Fitzgibbon, A. W. & Fisher, R. B. (1995). A Buyer's Guide to Conic Fitting. *Proc. 5th British Machine Vision Conference, 1995*, 513-522.
- Galton, F. (2011). *Finger Prints*. Recuperado el 19 de Junio del 2019 de <https://www.gutenberg.org/ebooks/36979>. (Trabajo original publicado en 1892)
- Gashi, I., Mason, S., Lugini, L., Marasco, E. & Cukic, B. (2014). Interoperability between Fingerprint Biometric Systems: An Empirical Study. *IEEE International Conference on Dependable Systems and Networks, 2014*.
- Gastal, E. & Oliveira, M. (2011). Domain transform for edge-aware image and video processing. *ACM Transactions on Graphics*, 30 (2011).
- Gonzales, R. & Woods, R. (2008). *Digital Image Processing*. 3a ed. United States of America: Pearson Education Inc.

- Google Colaboratory (2019). *Google Colaboratory a free Jupyter notebook environment*. Recuperado el 19 de Junio del 2019 de <https://colab.research.google.com/>
- Gupta, S., Anand, S. & Rai, A. (2017). Fingerprint Extraction Using Smartphone Camera. *CoRR*, 2017, *abs/1708.00884*.
- Haupt, G. & Mozer, T. (2015). Assessing biometric authentication: a holistic approach to accuracy. *Biometric Technology Today*, 2015(3), 5-8.
- Hawthorne, M. R. (2009). *Fingerprints: analysis and understanding*. United States of America: CRC Press
- Heeseung, C., Kyoungtaek, C. & Jaihie, K. (2010). Mosaicing Touchless and Mirror-Reflected Fingerprint Images. *IEEE Transactions on Information Forensics & Security*, 5(1), 52. doi:10.1109/TIFS.2009.2038758
- Hernández, S. R. (2006). *Metodología de la Investigación*. 5a ed. México: McGraw Hill.
- Hschott (2019). *CAndroid: Showcase of Android camera control and OpenCV image processing*. Recuperado el 19 de Junio del 2019 de <https://github.com/hschott/Candroid>.
- IEEE CBP (2012). *IEEE CBP Learning System. Module 2: Biometrics Modalities*. United States of America: IEEE CBP Service Center.
- Isecus (2019). Fingerprint ZKTeco Scanner SLK20R (Live20R). Recuperado el 19 de Junio del 2019 de <https://www.isecus.com/product/fingerprint-scanner-slk20r-live20r/>
- Jupyter (2019). *About Project Jupyter*. Recuperado el 19 de Junio del 2019 de <https://jupyter.org/about>

- Kaur, P., Jain, A. & Mittal, S. (2012). Touch-less Fingerprint Analysis - A Review and Comparison. *Intelligent Systems and Applications*, 2012(6), 46-52.
- Khalil, M. S., Kurniawan, F., & Saleem, K. (2013). Authentication of Fingerprint Biometrics Acquired using a Cellphone Camera: A review. *International Journal of Wavelets, Multiresolution & Information Processing*, 11(5), 1-17.
doi:10.1142/S0219691313500331
- Kingma, D. P. & Ba, J. (2014). Adam: A Method for Stochastic Optimization. *CoRR*, 2014, abs/1412.6980.
- Krasin, I., Duerig, T., Alldrin, N., Ferrari, V., Abu-El-Haija, S., Kuznetsova, A., Rom, H., Uijlings, J., Popov, S., Kamali, S., Mallocci, M., Pont-Tuset, J., Veit, A., Belongie, S., Gomes, V., Gupta, A., Sun, C., Chechik, G., Cai, D., Feng, Z., Narayanan, D. & Murphy, K. (2017). OpenImages: A public dataset for large-scale multi-label and multi-class image classification. Recuperado el 19 de Junio del 2019 de <https://storage.googleapis.com/openimages/web/index.html>
- Komarinski, P. (2005). *Automated Fingerprint Identification Systems (AFIS)*. United States of America: Elsevier Academic Press.
- Labati, R. D., Piuri, V. & Scotti, F. (2016). *Touchless Fingerprint Biometrics*. United States of America: CRC Press.
- Laganière, R. (2014). *OpenCV Computer Vision Application Programming Cookbook*. 2a ed. United States of America: Packt Publishing.
- Maddala, S. (2010). *Implementation and Evaluation of NIST Biometric Image Software for Fingerprint Recognition*. (Thesis for Master of Technology and Master of Science in Electrical Engineering with emphasis on Signal Processing) Blekinge Institute of Technology, Sweden.

- Maltoni, D., Maio, D., Jain, A. K. & Prabhakar, S. (2009). *Handbook of Fingerprint Recognition*. United States of America: Springer.
- Milletari, F., Navab, N. & Ahmadi, S. (2016). V-Net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation. *CoRR*, 2016, *abs/1606.04797*.
- Modi, S. (2008). *Analysis of Fingerprint Sensor Interoperability on System Performance*. (Thesis for Doctor of Philosophy, Technology specialization in Computational Sciences & Engineering). Purdue University, West Lafayette, United States of America.
- Modi, S., Kim, H., & Elliott, S. (2009). Statistical analysis of fingerprint sensor interoperability performance. *IEEE 3Rd International Conference on Biometrics: Theory, Applications and Systems, 2009*, 1-6. doi:10.1109/BTAS.2009.5339063
- Moore, Z. E., Elliott, S. J. & O'Connor, K. J. (2015). Interoperability and the Stability Score Index. *IT in Industry*, 3(4), 88-91.
- NIST NBIS (2019). *NIST Biometric Image Software (NBIS)*. Recuperado el 19 de Junio del 2019 de <https://www.nist.gov/services-resources/software/nist-biometric-image-software-nbis>
- NIST NBIS Userguide (2019). *NIST NBIS User's Guide to NIST Biometric Image Software (NBIS)*. Recuperado el 19 de Junio del 2019 de <https://www.nist.gov/publications/users-guide-nist-biometric-image-software-nbis>
- NIST NBIS-EC Userguide (2019). *Users Guide to Export Controlled Distribution of NIST Biometric Image Software (NBIS-EC)*. Recuperado el 19 de Junio del 2019 de <https://www.nist.gov/publications/users-guide-export-controlled-distribution-nist-biometric-image-software-nbis-ec>

- OpenCamera (2019). *OpenCamera for Android*. Recuperado el 19 de Junio del 2019 de <https://opencamera.sourceforge.io/>.
- OpenCV (2019). *About OpenCV*. Recuperado el 19 de Junio del 2019 de <https://opencv.org/about/>
- Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Trans. Sys., Man., Cyber*, 9 (1), 62–66. doi:10.1109/TSMC.1979.4310076
- Oude Elfenrink, W. (2014). *Fingerprint Acquisition with a Smartphone Camera*. (Bachelor Assignment). University of Twente, Enschede, Netherlands.
- Pressman, R. S. & Maxim, B. R. (2014). *Software Engineering. A practitioner's guide approach*. 8a ed. United States of America: Mc Graw Hill.
- Ronneberger, O., Fischer, P. & Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. *CoRR*, 2015, [abs/1505.04597](https://arxiv.org/abs/1505.04597).
- Rother, C., Kolmogorov, V. & Blake, A. (2004). "GrabCut": interactive foreground extraction using iterated graph cuts. *ACM SIGGRAPH*, 2004, 309-314. doi:10.1145/1186562.1015720
- Sankaran, A., Malhotra, A., Mittal, A., Vatsa, M. & Singh, R. (2015). On smartphone camera based fingerphoto authentication. *IEEE 7th International Conference on Biometrics Theory, Applications and Systems*, 2015. doi:10.1109/BTAS.2015.7358782
- Shelhamer, E., Long, J. & Darrell, T. (2017). Fully Convolutional Networks for Semantic Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(4), 640-651. doi:10.1109/TPAMI.2016.2572683
- Smith, S. (2011). *The Scientist and Engineer's Guide to Digital Signal Processing*. Recuperado el 19 de Junio del 2019 de <http://www.dspguide.com/CH1.PDF>
- Sommerville, I. (2011). *Ingeniería de Software*. Mexico: Pearson Education.

- Stein, C., Bouatou, V. & Busch, C. (2013). Video-based Fingerphoto Recognition with Anti-spoofing Techniques with Smartphone Cameras. *International Conference of the BIOSIG Special Interest Group, 2013*.
- Suzuki, S., & Abe, K. (1985) Topological Structural Analysis of Digitized Binary Images by Border Following. *CVGIP, 30(1)*, 32-46
- The R Foundation (2019). *About R*. Recuperado el 19 de Junio del 2019 de <https://www.r-project.org/about.html>
- Veliz-Capuñay, C. (2017). *Notas del curso de Análisis Multivariado*. Pontificia Universidad Católica del Perú, Lima, Perú.
- Xataka (2019). *Xiaomi Redmi 5 Plus, análisis*. Recuperado de <https://www.xatakandroid.com/analisis/xiaomi-redmi-5-plus-analisis-caracteristicas-precio-especificaciones>
- Xiaomi (2019). *Xiaomi Redmi 5 Plus*. Recuperado el 19 de Junio del 2019 de <https://www.mi.com/global/redmi-5-plus/>

ANEXOS

ANEXO A. Especificación de requerimientos de software para la aplicación móvil

A.1. Especificación de requerimientos de software

A.1.1. Introducción

En este anexo se definen las Especificaciones de Requisitos de Software para la Aplicación Móvil de Captura de Imágenes de Huellas Dactilares basados en las especificaciones del estándar IEEE 830.

A.1.1.1. Propósito

Este anexo tiene como objetivo presentar una descripción detallada de la aplicación.

A.1.1.2. Ámbito del sistema

La Aplicación Móvil de Captura de Imágenes de Huellas Dactilares será usada por otras aplicaciones que requieran obtener imágenes de huellas dactilares usando la cámara del dispositivo, típicamente como entrada a un sistema biométrico dactilar.

A.1.1.3. Definiciones, acrónimos y abreviaturas

Definiciones

- Sistema biométrico. Sistema biométrico de reconocimiento dactilar que puede realizar operaciones de autenticación e identificación.

Acrónimos

- ERS. Especificación de Requisitos de Software

A.1.1.4. Referencias

IEEE 830 - Recommended Practice for Software Requirements Specification

A.1.1.5. Visión general de la ERS

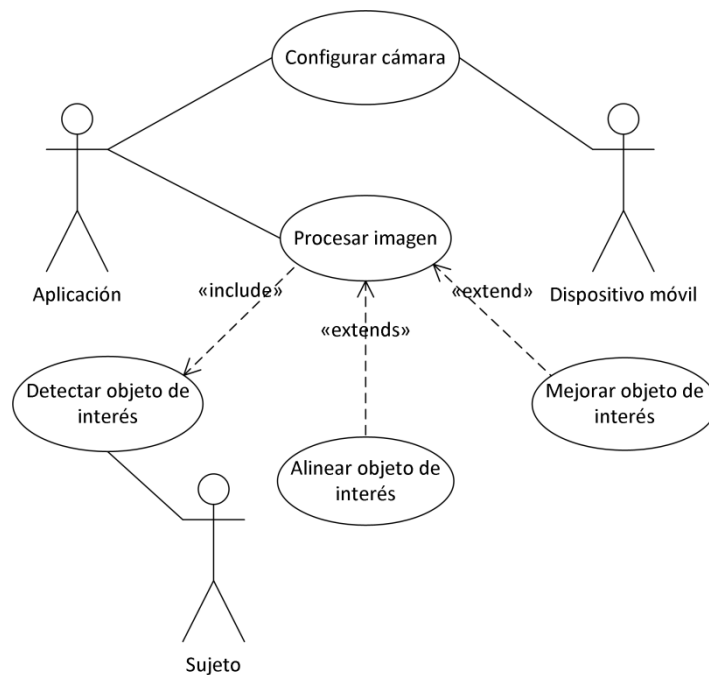
La Descripción general proporciona información general de la aplicación y detalla sus casos de uso. Los Requisitos específicos detallan los requerimientos para la aplicación.

A.1.2. Descripción general

La Aplicación Móvil de Captura de Imágenes de Huellas Dactilares es una aplicación para dispositivos Android que implementa un flujo de procesamiento de imágenes que usa las fotografías tomadas por la cámara del dispositivo para obtener imágenes de huellas dactilares que podrían ser usadas por sistemas biométricos dactilares tradicionales.

A.1.2.1. Modelo de casos de uso

Figura N° A.1: Modelo de casos de uso.



Elaboración propia.

Autor	Descripción
Aplicación	La aplicación propiamente dicha.
Dispositivo móvil	Teléfono inteligente con sistema Android 5.1 o superior con funcionalidad de cámara.
Sujeto	Usuario que captura su huella.

A.1.2.1.1. Casos de uso por paquete

Esta aplicación sólo está conformada por un paquete el cual contiene los casos de uso: Configurar cámara, Procesar imagen, Detectar objeto de interés, Alinear objeto de interés y Mejorar objeto de interés.

Configurar cámara

Administra la cámara del dispositivo y permite verificar y establecer las características requeridas.

Procesar imagen

Aplica el flujo de procesamiento de imágenes sobre la imagen (fotograma o fotografía).

Detectar objeto de interés

Detecta si la imagen contiene un dedo y genera una máscara de segmentación de la región que lo contiene.

Alinear objeto de interés

Recorta sólo la región que contiene el dedo y la gira a posición vertical.

Mejorar objeto de interés

Mejora la imagen a fin de resaltar los surcos y valles.

A.1.2.2. Características de los usuarios

Los usuarios son las aplicaciones que requieran obtener imágenes de huellas dactilares usando la cámara del dispositivo, típicamente como entrada a un sistema biométrico dactilar.

A.1.2.3. Suposiciones y dependencias

A.1.2.3.1. Suposiciones

- La aplicación será desarrollada sólo para el sistema Android.

- El dispositivo debe contar con una cámara móvil que soporte enfoque, cálculo de distancia de foco, modo macro y modo antorcha.
- El dedo será ubicado a una distancia de entre 7 a 15 centímetros de la cámara, preferentemente en posición horizontal y paralela al dispositivo.
- El dedo debe estar limpio y no contener objetos extraños que modifiquen su apariencia o color.
- El dedo debe estar inmóvil y extendido de manera que las crestas y valles puedan distinguirse en la imagen.
- Sólo se realizará la captura de un dedo a la vez.
- La imagen final será escalada a una resolución uniforme y un tamaño variable según las dimensiones del dedo.

A.1.2.3.2. Dependencias

- Dispositivo móvil con sistema Android 5.1 o superior.
- Cámara móvil con resolución mínima de 10MP.
- Cámara móvil que soporte enfoque, cálculo de distancia de foco, modo macro y modo antorcha.

A.1.3. Requisitos específicos

A continuación se detallan los requisitos por cada caso de uso.

A.1.3.1. Especificación de casos de uso

Configurar cámara en A.2.

Procesar imagen en A.3.

Detectar objeto de interés en A.4.

Alinear objeto de interés en A.5.

Mejorar objeto de interés en A.6.

A.1.3.2. Requisitos de interfaces externas

A.1.3.2.1. Interfaces de usuario

- La aplicación muestra una previsualización en tiempo real de la cámara.
- La aplicación muestra mensajes informativos o de error al usuario.

A.1.3.2.2. Interfaces de hardware

- La aplicación responde al toque del usuario sobre la pantalla enfocando en la región seleccionada.
- La aplicación administra la comunicación con la cámara del dispositivo.

A.1.3.2.3. Interfaces de software

- La aplicación escribe la imagen de la huella dactilar en un directorio en formato JPG o PNG o WSQ.

A.1.3.2.4. Interfaces de comunicación

- La aplicación se comunica con otras aplicaciones mediante la escritura de imágenes en disco.

A.1.3.3. Requisitos de rendimiento

- La aplicación funciona y responde en tiempo real.

A.1.3.4. Requisitos de desarrollo

A.1.3.4.1. Software

- Windows 10 Pro.
- JDK 8.
- Android Studio 3.4.1.
- Android SDK.
- Graddle.
- Librería OpenCV 3.43 para Android.
- USB OEM driver.

A.1.3.4.2. Hardware

- Sistema con procesador Intel Core i5 3.4GHz, 16GB RAM, GPU 4GB RAM, 500GB SSD, 1TB HD.
- Pantalla, teclado y mouse.
- Dispositivo Xiaomi Redmi 5 Plus y cable USB.
- Dispositivo Xiaomi en modo desarrollador configurado y con instalación por USB configurado.

A.1.3.5. Requisitos tecnológicos**A.1.3.5.1. Software**

- Android 8.1.0.

A.1.3.5.2. Hardware

- Xiaomi Redmi 5 Plus o dispositivo compatible.
- Cámara móvil con resolución mínima de 10MP.
- Cámara móvil que soporte enfoque, cálculo de distancia de foco, modo macro y modo antorcha.

A.1.3.6. Atributos de software**A.1.3.6.1. Seguridad**

- Ninguna.

A.1.3.6.2. Portabilidad

- Compatible con Android 5.1 o superior.
- Compatible con cámaras que cumplan las características mínimas requeridas.

A.1.3.6.3. Mantenibilidad

- Código fuente versionado.

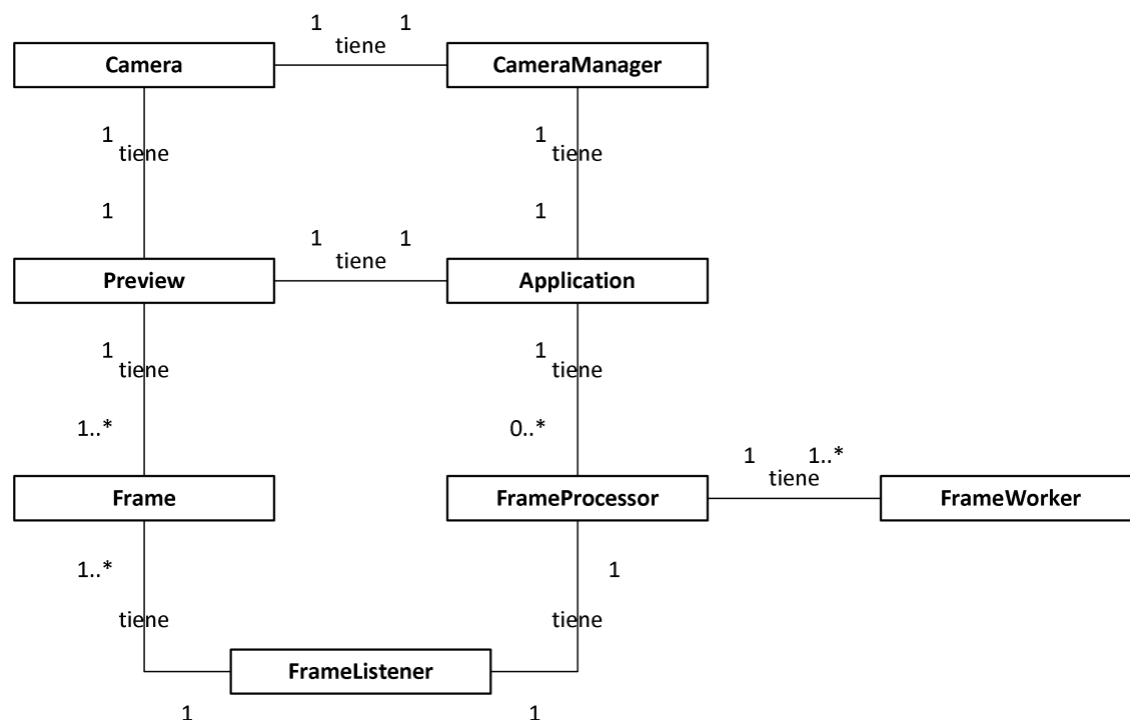
A.1.3.6.4. Fiabilidad

- Verificar las características requeridas para la cámara del dispositivo al iniciar la aplicación.
- Mostrar mensajes de error y éxito al usuario.
- Restringir la captura de objetos que no se encuentren a una distancia de 7cm a 15cm del foco de la cámara.
- Procesar imagen sólo cuando se haya detectado un posible dedo.
- Capturar huella sólo si la imagen procesada tiene una calidad aceptable.
- Liberar los recursos de hardware que no estén en uso.

A.1.4. Análisis

A.1.4.1. Diagrama de clases de análisis

Figura N° A.2: Diagrama de clases de análisis.



Elaboración propia.

A.1.5. Diseño

A.1.5.1. Diagrama de clases de diseño

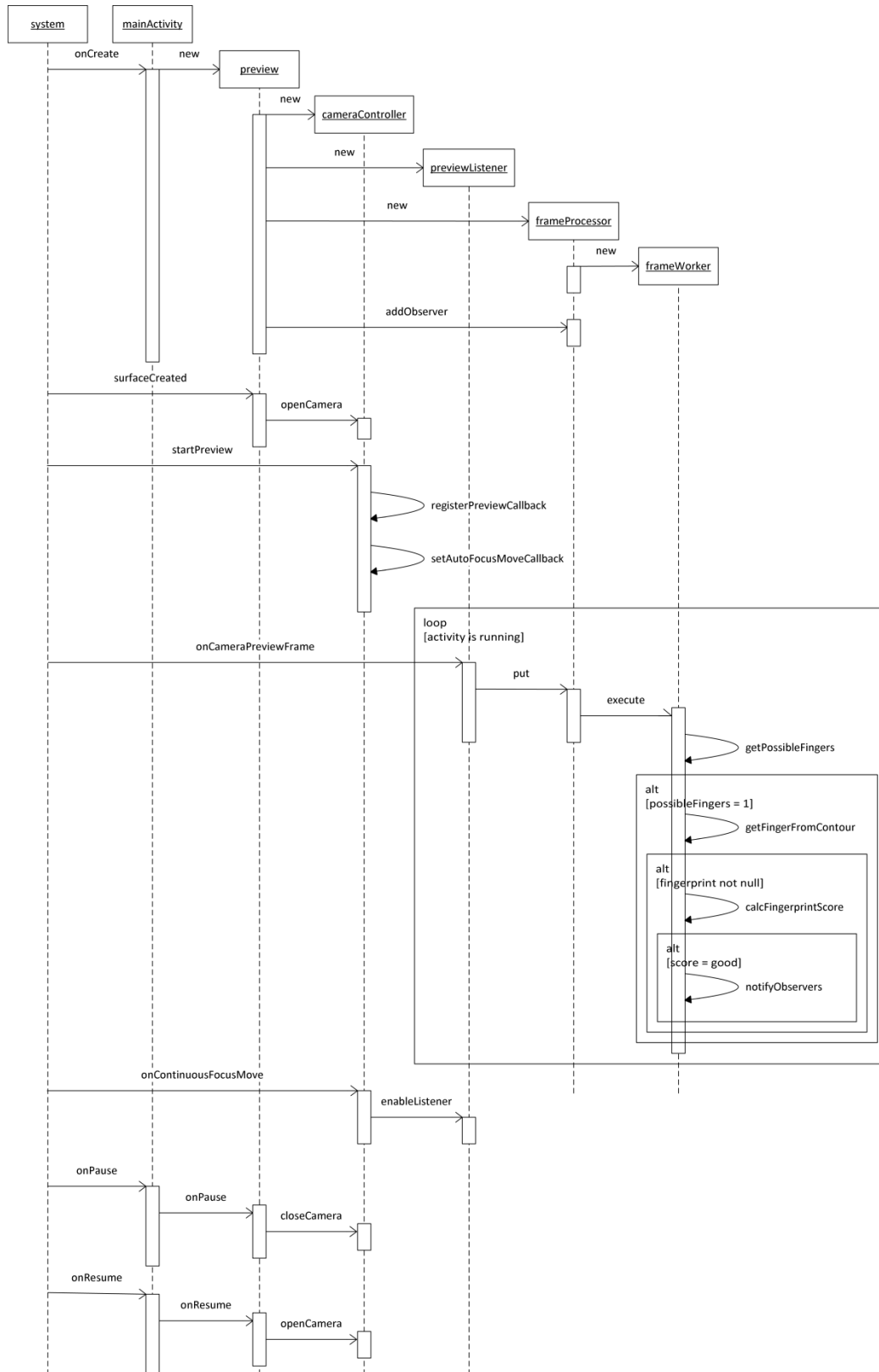
Figura N° A.3: Diagrama de clases de diseño.

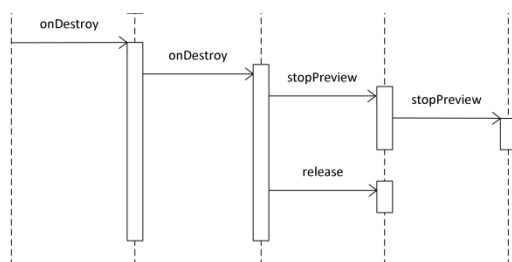


Elaboración propia.

A.1.5.2. Diagrama de secuencia

Figura N° A.4: Diagrama de secuencia.





Elaboración propia.

A.1.6. Artefactos

- APK.
- Código fuente.

A.1.7. Apéndices

A.2. Especificación del caso de uso: configurar cámara

A.2.1. Nombre del caso de uso

Breve descripción

Este caso de uso administra la cámara del dispositivo.

Actores

- Aplicación.
- Dispositivo móvil.

A.2.2. Flujo básico de eventos

	APLICACIÓN	DISPOSITIVO
1	Aplicación inicia.	
2	Aplicación verifica características de cámara.	
3	Aplicación inicializa cámara.	
4		Dispositivo inicializa cámara.

A.2.3. Flujos alternativos

A.2.3.1. Cámara no tiene las características requeridas

1	En 2, si la cámara no tiene las características requeridas la aplicación finaliza con un mensaje de error.	
---	--	--

A.2.4. Subflujos

	APLICACIÓN	DISPOSITIVO
1	Aplicación recibe evento de cierre.	
2	Aplicación detiene cámara.	
3		Dispositivo finaliza cámara y libera recursos.
4	Aplicación finaliza.	

A.2.5. Escenarios clave

La cámara es inicializada y configurada.

A.2.6. Pre condiciones

No aplica.

A.2.7. Post condiciones

No aplica.

A.2.8. Puntos de extensión

No aplica.

A.2.9. Requerimientos especiales

Dispositivo debe cumplir requisitos mínimos de hardware.

A.2.10. Información adicional

No aplica.

A.3. Especificación del caso de uso: procesar imagen

A.3.1. Nombre del caso de uso

Breve descripción

Este caso de uso ejecuta el flujo de procesamiento de imágenes para cada fotograma o fotografía.

Actores

- Aplicación.

A.3.2. Flujo básico de eventos

	APLICACIÓN
1	Aplicación y cámara configurada e inicializada.
2	Aplicación ha recibido un fotograma o fotografía de la cámara y la muestra en la interfaz.
3	Aplicación verifica que la distancia del foco al objeto enfocado se encuentre en el rango de 7cm a 15cm.
4	Aplicación inicia caso de uso Detectar objeto de interés .
5	Si objeto de interés fue detectado, Aplicación inicia caso de uso Alinear objeto de interés .
6	Si objeto de interés fue alineado, Aplicación inicia caso de uso Mejorar objeto de interés .
7	Aplicación verifica calidad de objeto de interés final.
8	Si la calidad es adecuada e: - Imagen procesada era un fotograma Aplicación se genera evento para tomar fotografía. - Imagen procesada era una fotografía se guarda archivo y se envía mensaje "Huella dactilar capturada".
9	Aplicación repite caso de uso hasta que recibe evento de cierre de usuario.

A.3.3. Flujos alternativos

A.3.3.1. Distancia de foco a objeto enfocado no está en rango aceptable

1	En 3, si la distancia de foco a objeto enfocado no está en rango aceptable la aplicación envía mensaje “Coloque su dedo de 7 a 15cm de la cámara” y continúa en paso 9.
---	---

A.3.3.2. No se detectó objeto de interés

1	En 5, si no se detectó objeto de interés aplicación continúa en paso 9.
---	---

A.3.3.3. No se alineó objeto de interés

1	En 5, si no se alineó objeto de interés aplicación continúa en paso 9.
---	--

A.3.3.4. Objeto de interés final no tiene calidad adecuada

1	En 8, si objeto de interés final no tiene calidad adecuada aplicación continúa en paso 9.
---	---

A.3.4. Subflujos

No aplica.

A.3.5. Escenarios clave

No aplica.

A.3.6. Pre condiciones

Caso de uso **Configurar cámara** ejecutado previamente.

A.3.7. Post condiciones

No aplica.

A.3.8. Puntos de extensión

En 4 se inicia caso de uso **Detectar región de interés**.

En 5 se inicia caso de uso **Alinear región de interés**.

En 6 se inicia caso de uso **Mejorar región de interés**.

A.3.9. Requerimientos especiales

No aplica.

A.3.10. Información adicional

No aplica.

A.4. Especificación del caso de uso: detectar objeto de interés**A.4.1. Nombre del caso de uso****Breve descripción**

Este caso de uso realiza la detección de un posible dedo en la imagen procesada (objeto de interés).

Actores

- Aplicación.
- Sujeto.

A.4.2. Flujo básico de eventos

	SUJETO	APLICACIÓN
1	Sujeto muestra su dedo a una distancia de 7cm a 15cm del foco de la cámara.	
2		Aplicación analiza el dominio de los colores de la imagen para encontrar las regiones que podrían contener un dedo.
3		Aplicación verifica las características estructurales de las regiones que posiblemente contengan un dedo y selecciona aquellas que parezcan un dedo.

A.4.3. Flujos alternativos

No aplica.

A.4.4. Subflujos

	SUJETO	APLICACIÓN
1	Sujeto toca la pantalla.	
2		Aplicación genera evento de enfoque en la posición tocada.
3		Aplicación ejecuta flujo básico de eventos al terminar el enfoque.

A.4.5. Escenarios clave

Regiones que parecen dedos son encontradas.

A.4.6. Pre condiciones

Caso de uso **Procesar imagen** en ejecución.

A.4.7. Post condiciones

No aplica.

A.4.8. Puntos de extensión

No aplica.

A.4.9. Requerimientos especiales

No aplica.

A.4.10. Información adicional

No aplica.

A.5. Especificación del caso de uso: alinear región de interés**A.5.1. Nombre del caso de uso****Breve descripción**

Este caso de uso alinea la región de interés de manera que finalice en una posición vertical con la punta del dedo apuntando hacia arriba.

Actores

- Sistema.

A.5.2. Flujo básico de eventos

	SISTEMA
1	Aplicación recorta región de interés de imagen original.
2	Aplicación gira región de interés recortada a posición vertical.
3	Aplicación detecta punta del dedo y gira región de interés si punta del dedo se encuentra en la parte baja de la imagen.

A.5.3. Flujos alternativos

No aplica.

A.5.4. Subflujos

No aplica.

A.5.5. Escenarios clave

Región de interés rotada a posición vertical con la punta del dedo apuntando hacia arriba.

A.5.6. Pre condiciones

Caso de uso **Procesar imagen** en ejecución.

A.5.7. Post condiciones

No aplica.

A.5.8. Puntos de extensión

No aplica.

A.5.9. Requerimientos especiales

No aplica.

A.5.10. Información adicional

No aplica.

A.6. Especificación del caso de uso: mejorar región de interés**A.6.1. Nombre del caso de uso****Breve descripción**

Este caso de uso mejora la región de interés seleccionada y la escala a una resolución uniforme.

Actores

- Sistema.

A.6.2. Flujo básico de eventos

	SISTEMA
1	Aplicación mejora bordes (surcos y valles) presentes en región de interés.
2	Aplicación convierte región de interés a blancos y negros.
3	Aplicación escala región de interés a resolución uniforme de 500ppi.
4	Aplicación recorta región de interés para quedarse con la parte central de la huella.

A.6.3. Flujos alternativos

No aplica.

A.6.4. Subflujos

No aplica.

A.6.5. Escenarios clave

Región de interés mejorada y escalada.

A.6.6. Pre condiciones

Caso de uso **Procesar imagen** en ejecución.

A.6.7. Post condiciones

No aplica.

A.6.8. Puntos de extensión

No aplica.

A.6.9. Requerimientos especiales

No aplica.

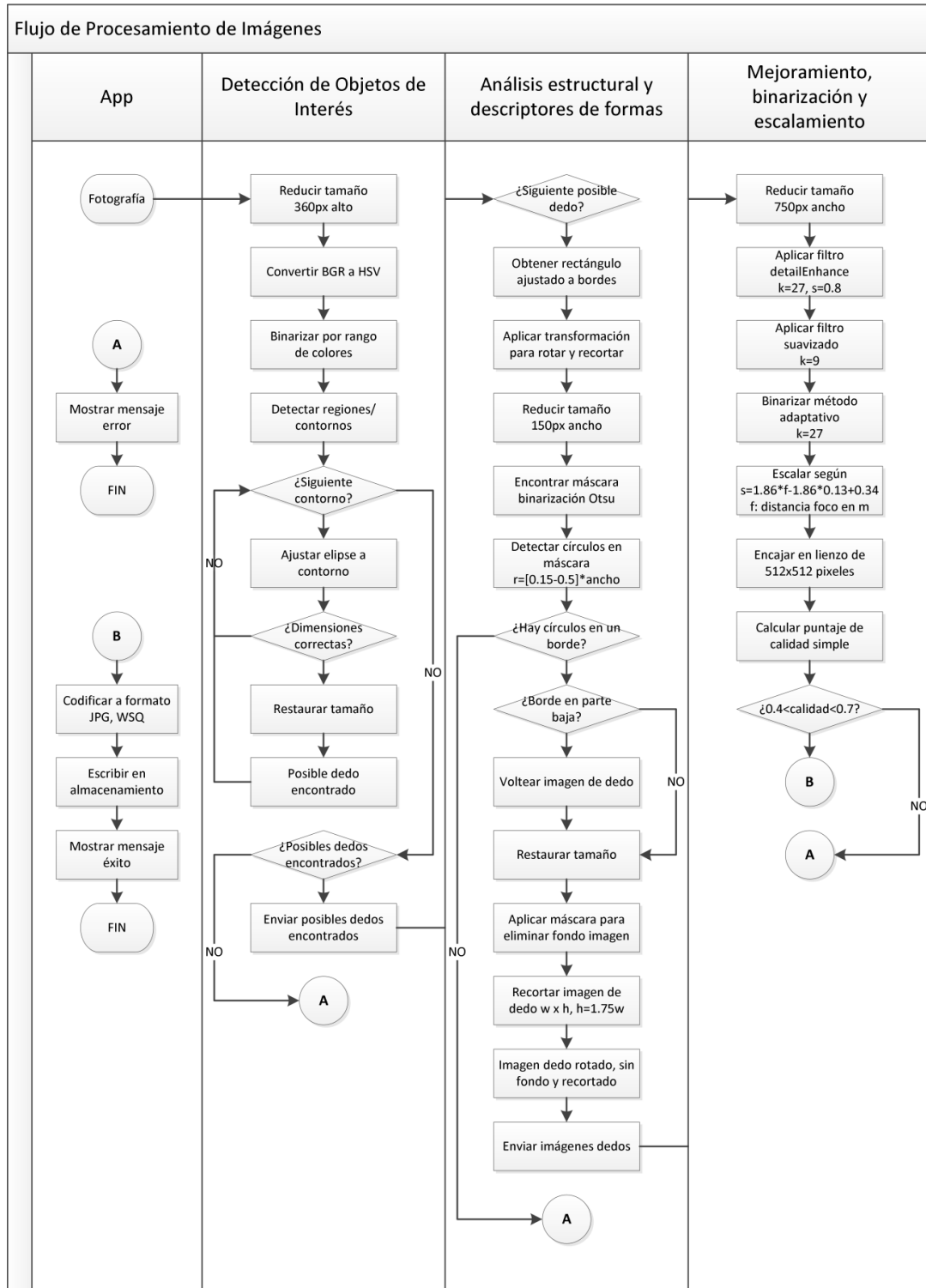
A.6.10. Información adicional

No aplica.

ANEXO B. Detalle del flujo de procesamiento de imágenes propuesto

B.1. Flujo de procesamiento de imágenes propuesto

Figura N° B.1: Flujo de procesamiento de imágenes propuesto.



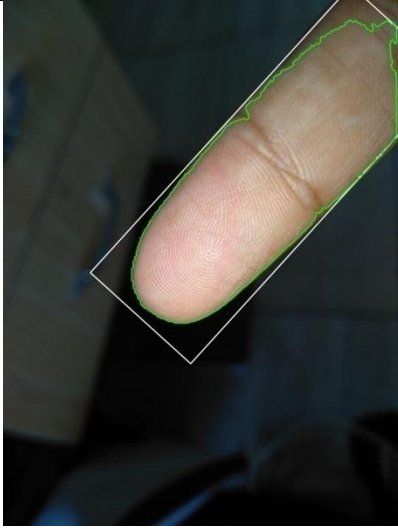


Elaboración propia.

B.2. Detección del objeto de interés

La Figura N° B.2 muestra las imágenes más importantes del flujo de procesamiento de esta etapa.

Figura N° B.2: Imágenes de la etapa de detección del objeto de interés.

Inicio	Reducir tamaño, convertir a HSV y binarizar	Detectar regiones, ajustar elipses y encontrar posibles dedos
 <p>3000x4000 pixeles (escala 1:20)</p>	 <p>270x360 pixeles (escala 1:2)</p>	 <p>3000x4000 pixeles (escala 1:20)</p>

Elaboración propia.




B.3. Análisis estructural y descriptores de formas

La Figura N° B.3 muestra las imágenes más importantes del flujo de procesamiento de esta etapa.

B.4. Mejoramiento, binarización y escalamiento




La Figura N° B.4 muestra las imágenes más importantes del flujo de procesamiento de esta etapa.

Figura N° B.3: Imágenes de la etapa de análisis estructural.

Ajustar rectángulo, transformación y recorte	Reducir, binarizar y detectar círculos	Voltar, restaurar tamaño y eliminar fondo
 <p>1029x2343 pixeles (escala 1:10)</p>	 <p>con círculos 65x150 pixeles (escala 1:2)</p>	 <p>1028x1800 pixeles (escala 1:10)</p>

Elaboración propia.

Figura N° B.4: Imágenes de la etapa de mejoramiento.

Reducir tamaño y aplicar filtros	Binarizar	Escarar y encajar en lienzo de 512x512
 <p>750x1313 pixeles (escala 1:5)</p>	 <p>750x1313 pixeles (escala 1:5)</p>	 <p>512x512 pixeles (escala 1:5)</p>

Elaboración propia.

ANEXO C. Modelo de aprendizaje profundo para segmentación de palmas basado en una CNN

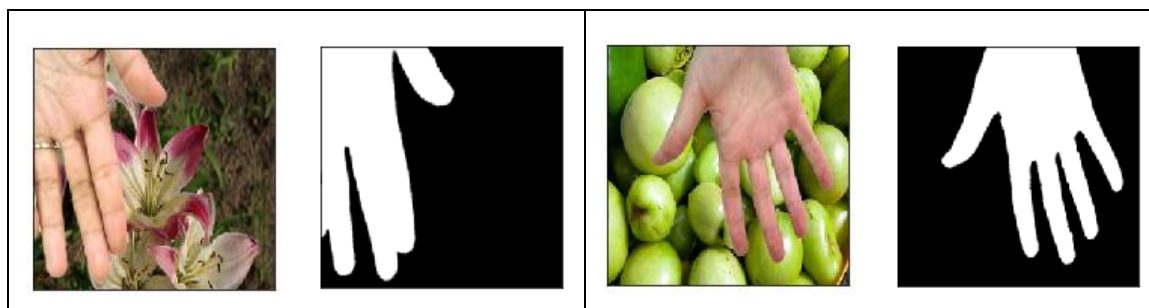
Avances recientes en la visión computacional y el reconocimiento de patrones han destacado la capacidad de las redes neuronales convolucionales (CNN) para resolver tareas complejas como la clasificación, la segmentación y la detección de objetos. Las redes neuronales convolucionales (CNN) tienen la capacidad de “aprender” una representación jerárquica de datos de entrada que no necesita haber sido procesada ni tener características creadas a mano. (Milletari et al., 2016)

A continuación se detalla el proceso de entrenamiento de un modelo completamente convolucional basado en U-net (Ronneberger et al., 2015) sobre un conjunto de datos de 40,138 imágenes generadas por el investigador.

C.1. Generación del conjunto de datos de 40,138 imágenes

El conjunto de datos fue generado a partir de 5,395 palmas del conjunto de datos 11k hands (Afifi, 2017), a cada una de estas imágenes se le generó una máscara de segmentación de forma manual mediante la herramienta Interactive Image Segmentation (2019). A partir de las imágenes de palmas y las máscaras de segmentación se construye el conjunto de datos final, para lo cual se usa la técnica de aumentación de datos mediante una matriz de transformación aleatoria (traslación, rotación, escalamiento) combinando el resultado con un fondo aleatorio del conjunto de datos OpenImages (Krasin et al, 2017). En la Figura N° C.1 se muestra imágenes aleatorias del conjunto de datos final:

Figura N° C.1: Muestra del conjunto de datos de 40,138 imágenes generadas por el investigador.

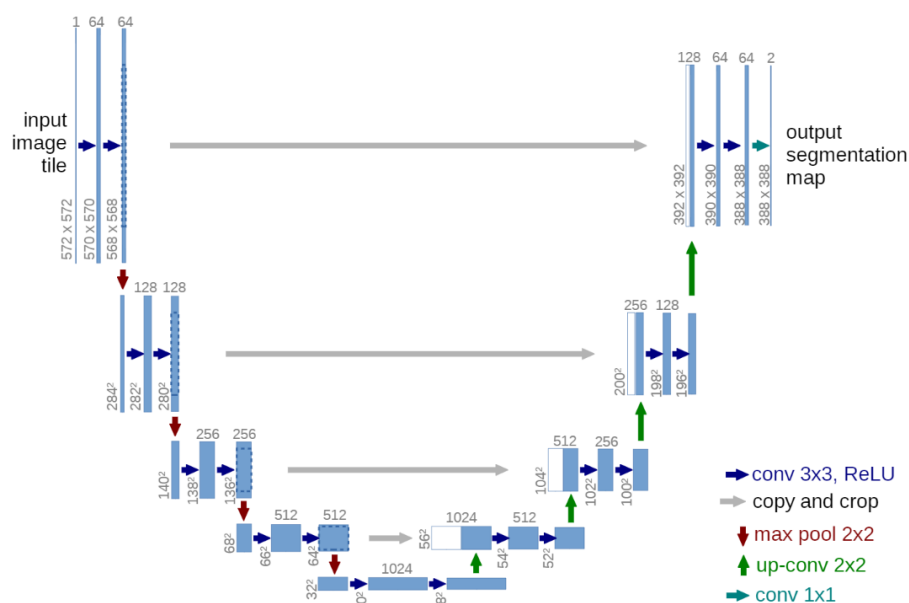


Elaboración propia.

C.2. Arquitectura del modelo

La arquitectura del modelo se muestra en la Figura N° C.2 y está basada en el modelo U-net propuesto por Ronneberger et al. (2015). El modelo contiene capas que van reduciéndose en la izquierda e incrementándose en la derecha. El lado izquierdo de la arquitectura sigue el modelo de una red completamente convolucional, convoluciones de 3x3, activaciones ReLU y max-pooling de 2x2 con saltos de 2. El lado de la derecha usa “up-convolution” de 2x2, convoluciones de 3x3, y activaciones ReLU. La última capa usa una convolución de 1x1. (Ronnerberger et al., 2015).

Figura N° C.2: Arquitectura del modelo completamente convolucional.

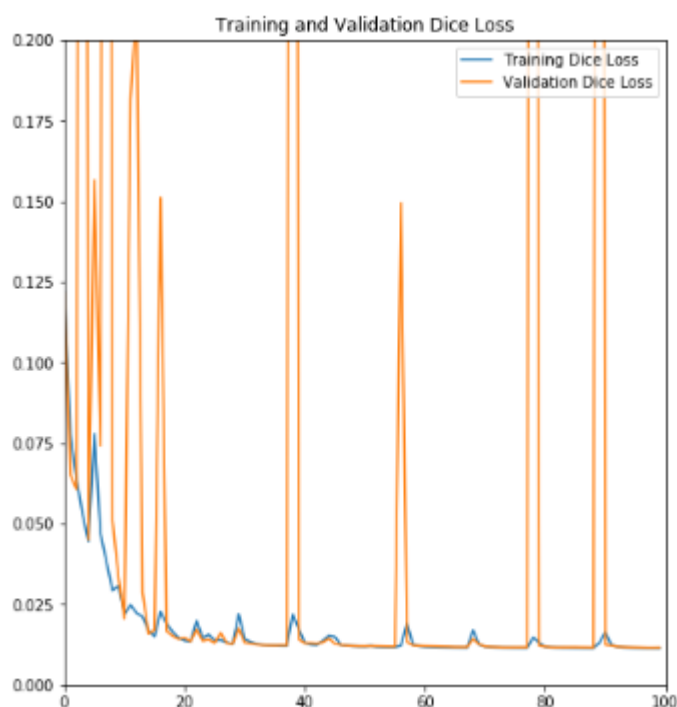


Elaboración propia.

C.3. Entrenamiento del modelo

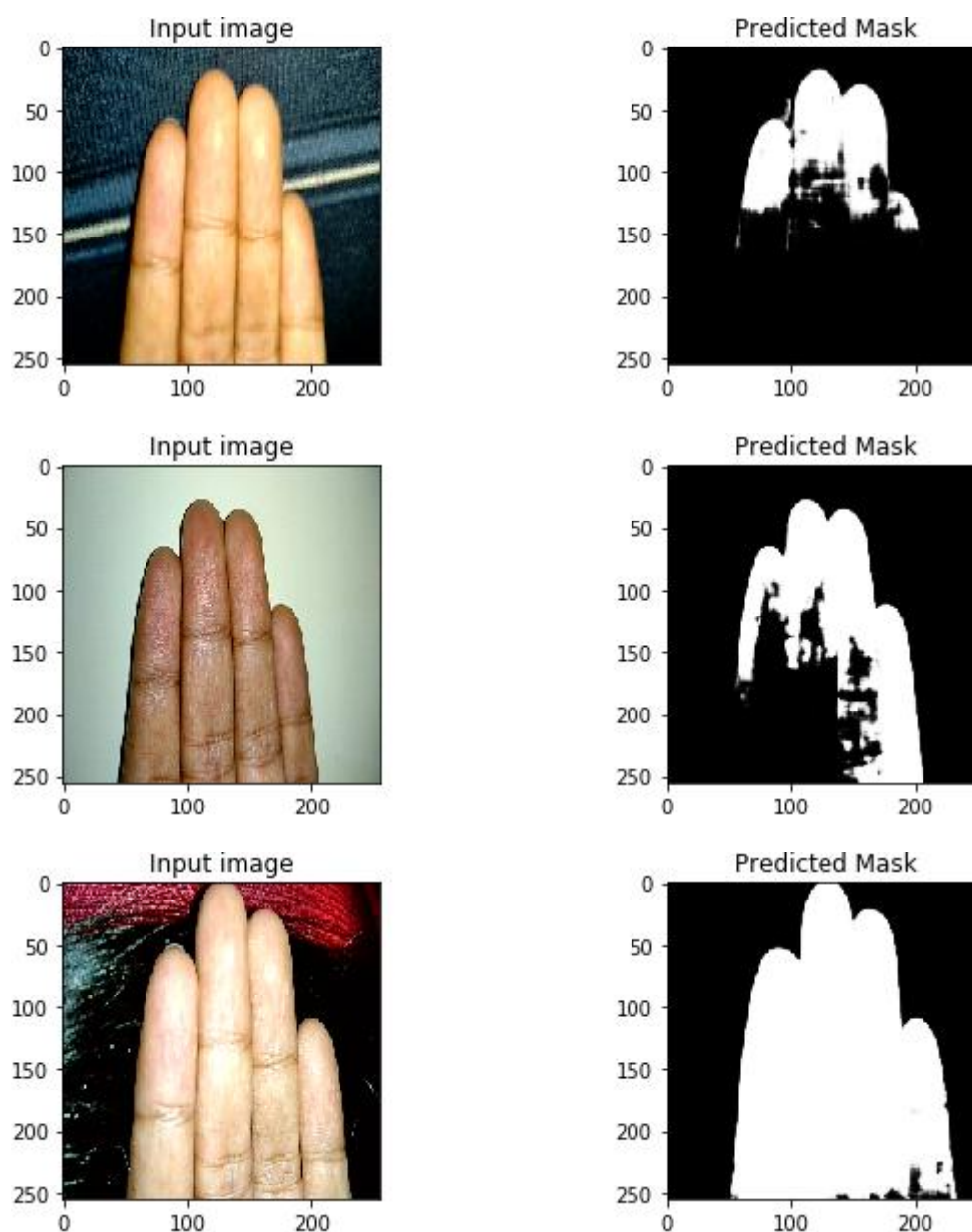
El proceso de entrenamiento se realizó sobre la plataforma Google Colaboratory (2019). Se usaron las funciones de pérdida de Dice (Milletari et al., 2016) y binary cross entropy. El conjunto de datos de entrenamiento inicial fue de 3,500 imágenes y de 100 épocas en batches de 8 con optimizador Adam (Kingma & Ba, 2017). Los resultados del error de validación se muestran en la Figura N° C.3.

Figura N° C.3: Gráfico de pérdida del conjunto de entrenamiento y validación.



Elaboración propia.

Luego de 100 épocas de entrenamiento, el modelo es capaz de inferir las máscaras de segmentación para nuevas imágenes las cuales se muestran en la Figura N° C.4.

Figura N° C.4: Máscaras de segmentación inferidas en imágenes de prueba.

Elaboración propia.

Los resultados son prometedores; sin embargo, el proceso de inferencia toma alrededor de 1 segundo en CPU (aproximadamente 100ms en GPU). Además el modelo entrenado supera los 300MB de tamaño. Estas restricciones limitan su uso en aplicaciones móviles en tiempo real.

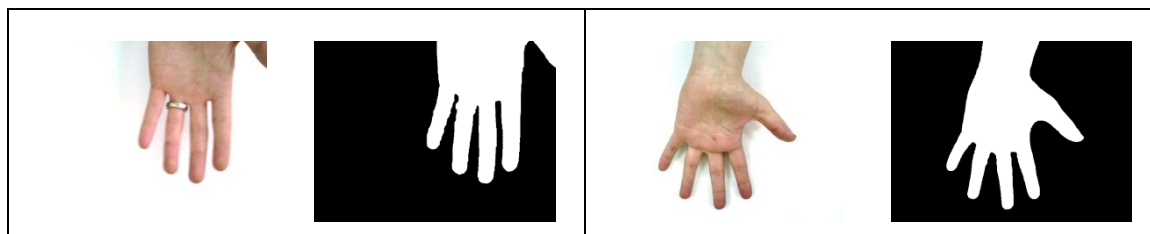
ANEXO D. Análisis de colores HSV para la etapa de detección de objeto de interés del flujo propuesto de procesamiento de imágenes

D.1. Construcción del conjunto de datos de 5,388 histogramas HSV

El conjunto de datos de 5,388 imágenes de máscaras de palmas fue construido a partir del conjunto de datos 11K hands (Afifi, 2017). El conjunto de datos 11K hands contiene 11,076 imágenes de manos (1600 x 1200 píxeles) de 190 sujetos de entre 18 y 75 años de edad. Cada sujeto tiene varias imágenes con los dedos abiertos y cerrados de sus manos derecha e izquierda. Cada mano se fotografió desde ambos lados: dorsal y palmar con un fondo blanco uniforme y se colocó aproximadamente a la misma distancia de la cámara. (Afifi, 2017)

En un primer paso se seleccionaron sólo las imágenes que corresponden a 5,395 imágenes de palmas. Para encontrar la máscara de segmentación, estas imágenes seleccionadas fueron preprocesadas usando la implementación del algoritmo GrabCut (Rother, Kolmogorov & Blake, 2004) de la librería OpenCV, este proceso fue realizado de forma manual y semi-automatizada a partir de la modificación de una herramienta basada en Python (Interactive Image Segmentation, 2019). Finalizando este proceso se obtuvieron 5,388 imágenes de palmas con sus respectivas máscaras de segmentación.

Figura N° D.1: Muestra del conjunto de datos de 5,388 imágenes de máscaras de segmentación construidas por el investigador.



Elaboración propia.

En un siguiente paso cada imagen de este conjunto de datos resultante es convertida al modelo de colores HSV para calcular el histograma de cada canal

considerando sólo la palma. Para cada histograma se calculan la media, mediana, moda y desviación estándar, tal y como se muestra en la Tabla N° D.1.

Tabla N° D.1: Estructura del conjunto de datos de estadísticas de los canales HSV de cada imagen.

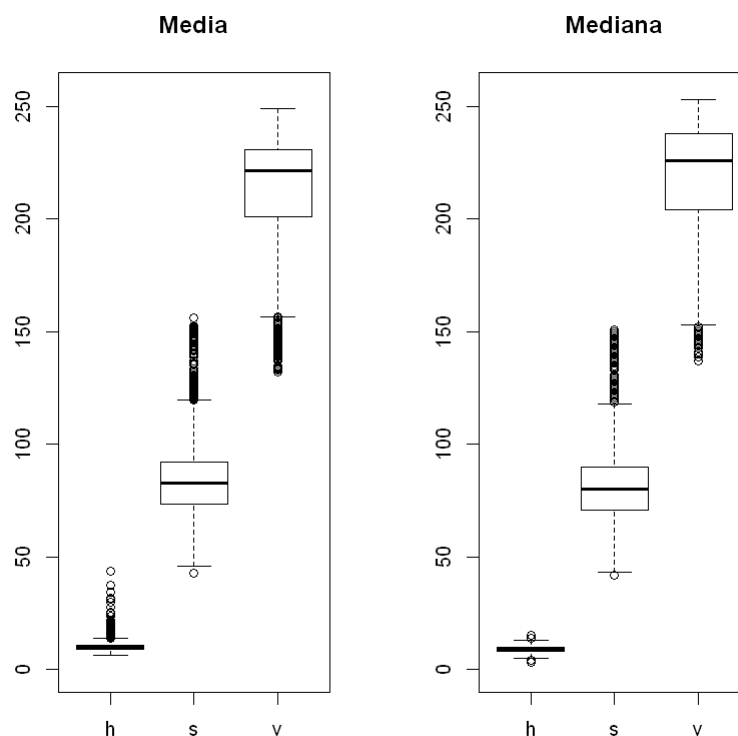
file	mean_h	mode_h	media_n_h	std_h	mean_s	mode_s	media_n_s	std_s	mean_v	mode_v	media_n_v	std_v
Hand_0000038.jpg	10.18	9	9	16.07	85.96	85	85	17.70	198.74	203	199	26.87
Hand_0000039.jpg	8.59	9	8	8.30	83.71	81	83	18.10	207.59	255	209	26.45
Hand_0000040.jpg	8.76	9	8	7.71	85.15	85	84	17.47	207.07	255	208	26.78
...

Elaboración propia.

D.2. Análisis estadístico exploratorio del conjunto de datos

Al realizar el análisis exploratorio del conjunto de datos generado se encuentra una mayor cantidad de valores atípicos cuando se usa la media de los colores como medida de posición; tal y como se muestra en la Figura N° D.2. Por ejemplo; en el componente Matiz (Hue), una inspección más detallada de los valores atípicos muestra que las máscaras para las imágenes que representan estos valores incluyen parte del atuendo del sujeto (imágenes: 0000038, 0002205, 0004153). Sin embargo, al usar la mediana se puede apreciar que aún existen valores atípicos en los componentes Saturación (Saturation) y Valor/Brillo (Value). En ambos componentes los valores atípicos más extremos se corresponden con imágenes de sujetos de raza negra (imágenes: 0009338, 0009430, 0009492, 0011582, 0009392, 0009433).

Figura N° D.2: Diagrama de cajas de la media y la mediana de los canales HSV de las regiones de las manos.



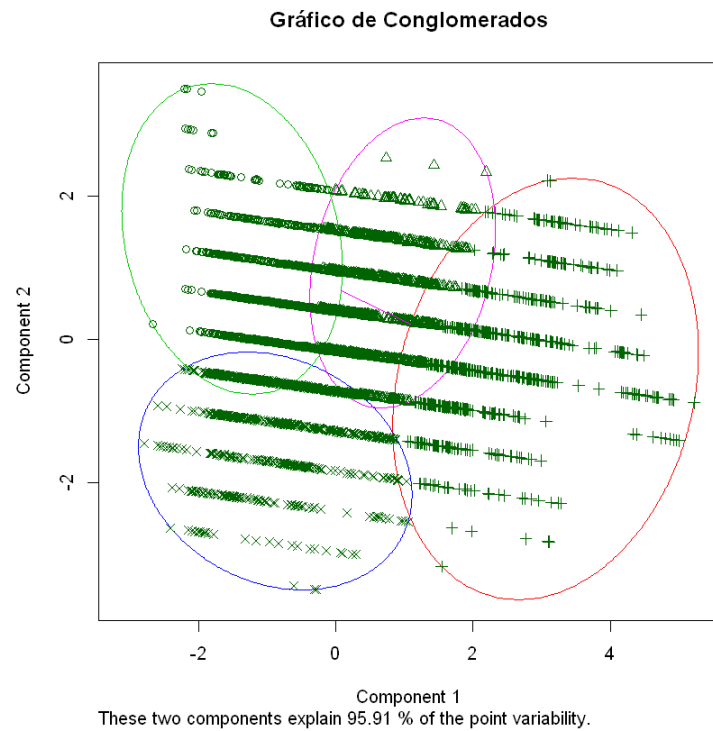
Elaboración propia.

D.3. Análisis de conglomerados para encontrar perfiles de colores de piel

A fin de determinar los rangos de colores que podrían considerarse como piel, primero se debe encontrar los perfiles de colores o grupos. Para este propósito se usaron técnicas de aprendizaje no-supervisado (análisis de conglomerados usando k-medias) con el objetivo de encontrar cada clúster y su respectivo centroide, para lo cual se aplicó el algoritmo K-medias con 4 grupos sobre la mediana de los componentes HSV de cada imagen, obteniéndose el resultado de la Figura N° D.3. Un análisis de clusterización jerárquica confirma la existencia de 4 grupos¹, tal y como lo muestra la Figura N° D.4.

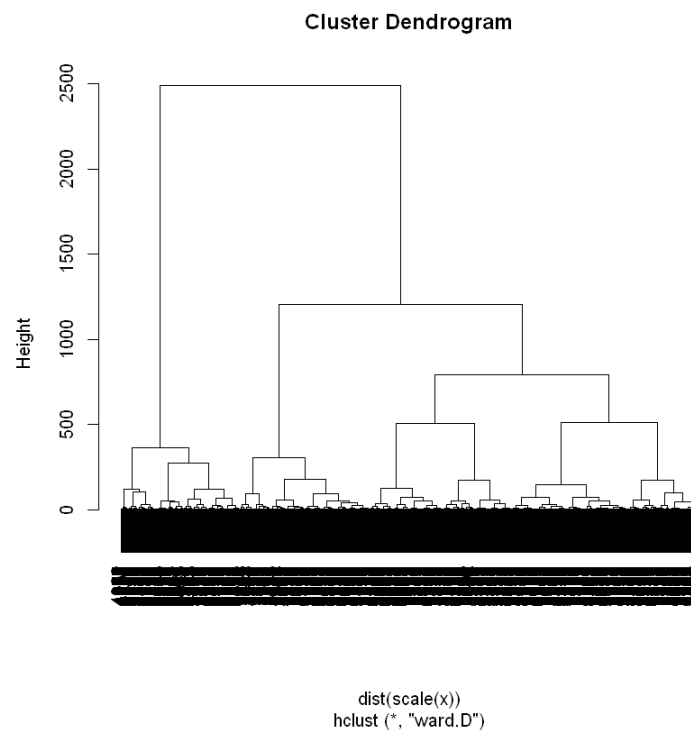
¹ Según la información demográfica del conjunto de datos 11k-hands (Afifi, 2017), el color de la piel está compuesto por 4 categorías: muy claro, claro, medio, y oscuro

Figura N° D.3: Diagrama de conglomerados de la mediana de los canales HSV de las regiones de las manos.



Elaboración propia.

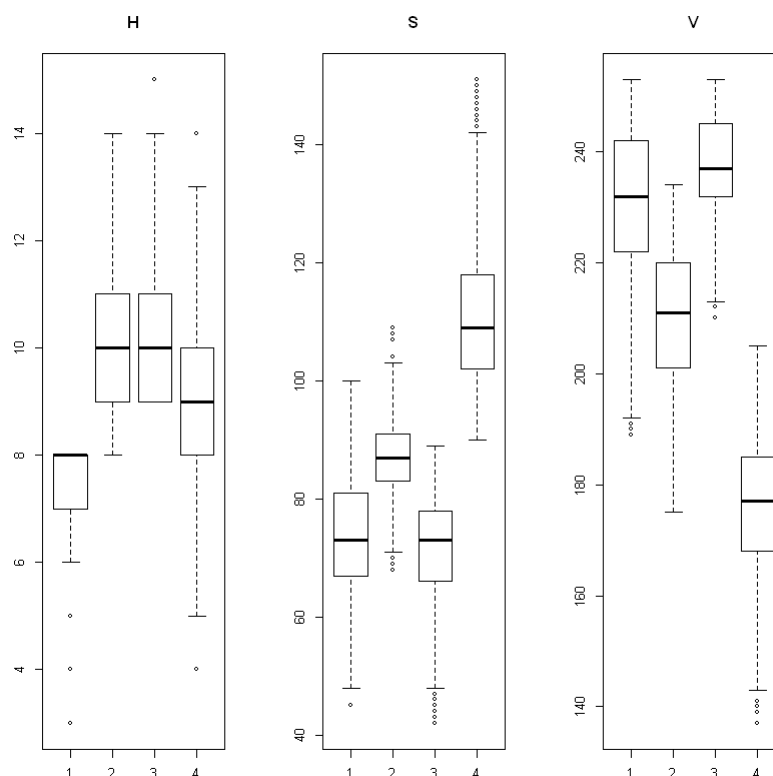
Figura N° D.4: Dendrograma que confirma la existencia de 4 grupos.



Elaboración propia.

Finalmente, se usa esta información para tomar los percentiles 5 y 95 como rangos de color que se considerarían color de piel para cada perfil o grupo encontrado. La Tabla N° D.2 muestra los percentiles para cada grupo.

Figura N° D.5: Diagrama de cajas de los canales HSV de los 4 grupos encontrados.



Elaboración propia.

Tabla N° D.2: Percentiles 5 y 95 para cada grupo encontrado.

	median_h_q5	median_h_q95	median_s_q5	median_s_q95	median_v_q5	median_v_q95
1	5	8	60.0	91	204.0	250
2	8	12	76.0	99	185.3	228
3	9	12	58.0	83	226.0	251
4	7	13	95.3	139	149.0	195

Elaboración propia.

ANEXO E. Elaboración del conjunto de datos de número de minucias

Este conjunto de datos se construye a partir de las 184 imágenes capturadas con cámara y 321 imágenes capturadas con sensor, y contiene el número de características de nivel 2 (posición, ángulo y tipo) encontradas en cada imagen de huella dactilar capturada. Las características de nivel 2 minucias, se pueden clasificar en bifurcaciones y terminaciones, para mayor información sobre las características de nivel 2 el lector puede referirse al marco teórico de esta investigación (Biometría dactilar).

E.1. Preparación de las imágenes capturadas

Todas las imágenes capturadas son colocadas en un directorio, organizadas por método de captura.

E.2. Extracción de minucias de cada imagen

Se usó la aplicación mindtct del paquete NBIS con los argumentos mostrados en la Figura N° 3.3. La salida de esta aplicación es un archivo XYT (o plantilla XYT) formateada según el estándar ANSI INCITS 378-2004. La Figura N° E.1 muestra las primeras filas de una plantilla XYT, el primer valor representa la coordenada x, el segundo valor la coordenada y, el tercer valor la dirección y el ángulo, y el último valor la calidad de la minucia. Para mayor información el lector se puede dirigir a NIST NBIS Userguide (2019).

Figura N° E.1: Plantilla de minucias obtenidas al ejecutar la aplicación mindtct.

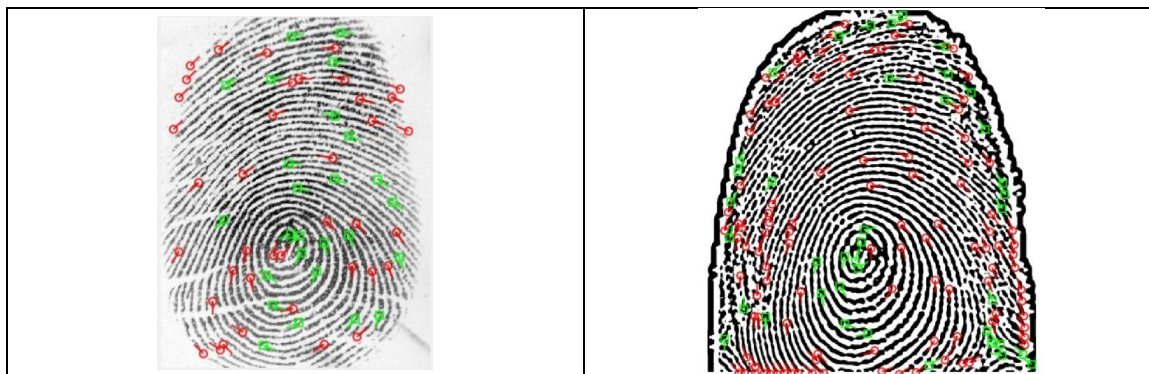
```
./mindtct -m1 fingerprint.jpg fingerprint
```

```
15 259 39 9  
16 186 45 9  
16 359 157 19  
...
```

Elaboración propia.

La Figura E.2 presenta visualmente las minucias encontradas en una imagen capturada por sensor y otra imagen capturada por la cámara.

Figura N° E.2: Minucias de una imagen capturada por sensor y de una imagen capturada por cámara.



Elaboración propia.

Para obtener la plantilla XYT de las imágenes del grupo “sensor” se usó el script mostrado en la Figura N° E.3, el mismo script con pequeñas variaciones es usado para las imágenes del grupo “cam”.

Figura N° E.3: Script bash para extraer las minucias de las imágenes de un directorio.

```
#!/bin/bash
for f in sensor /*.jpg ;
do echo "$f" ;
./mindtct -m1 ./ $f ./ $f ;
mv ./ $f.xyt sensor_min ;
done
```

Elaboración propia.

E.3. Conjunto de datos: minutiae.csv

El conjunto de datos minutiae.csv se genera mediante el script mostrado en la Figura N° E.4 luego de que todas las plantillas XYT se copiaran a un mismo directorio.

Figura N° E.4: Script bash para generar el conjunto de datos minutiae.csv.

```
#!/bin/bash
for file in *.xyt ;
do echo "$file" ;
echo -n "$file," >> minutiae.csv ;
wc -l < $file >> minutiae.csv ;
done
```

Elaboración propia.

El método de captura de cada huella puede extraerse a partir del nombre del archivo. Cada nombre de archivo sigue el formato: <método-de-captura>-<sujeito>-<código-dedo>-<otros-datos>.jpg.xyt. La Figura N° E.5 muestra las primeras filas del conjunto de datos en formato CSV.

Figura N° E.5: Estructura del conjunto de datos minutiae.csv.

```
cam-P1-F10-...jpg.xyt,151
cam-P1-F10-...jpg.xyt,165
...
```

Elaboración propia.

E.4. Post-procesamiento del conjunto de datos

El conjunto de datos final contiene los campos: “minutiae”, que representa el número de minucias, y “group”, que representa el método de captura de la huella. Estos campos son generados ejecutando el script mostrado en la Figura N° E.6.

Figura N° E.6: Script en R para generar el conjunto de datos de números de minucias.

```
csv = read.csv(file="minutiae.csv", header=FALSE, sep=",")
data = data.frame(
  group=str_extract(csv$V1, '[a-z]+'),
  minutiae=csv$V2)
```

Elaboración propia.

Finalmente, el conjunto de datos final tiene la estructura mostrada en la Tabla N°

E.1.

Tabla N° E.1: Estructura del conjunto de datos de números de minucias.

group	Minutiae
cam	151
...	...

Elaboración propia.

ANEXO F. Elaboración del conjunto de datos de índices de calidad

Este conjunto de datos se construye a partir de las 184 imágenes capturadas con cámara y 321 imágenes capturadas con sensor, y contiene el índice de calidad según el algoritmo NFIQ del NIST calculado para cada imagen de huella dactilar capturada. El índice de calidad NFIQ representa una estimación de la calidad de una imagen, hay cinco niveles de calidad, siendo 1 la calidad más alta y 5 la calidad más baja, para mayor información el lector puede referirse a NIST NBIS Userguide (2019).

F.1. Preparación de las imágenes capturadas

Todas las imágenes capturadas son colocadas en un directorio, organizadas por método de captura.

F.2. Cálculo del índice de calidad de cada imagen

Se usó la aplicación nfiq del paquete NBIS con los argumentos mostrados en la Figura N° 3.4. La salida de esta aplicación es un número del 1 al 5, que representa la calificación de calidad según el algoritmo NFIQ del NIST (NIST NBIS Userguide, 2019). La Figura N° F.1 muestra el resultado para una imagen.

Figura N° F.1: Índice de calidad obtenido al ejecutar la aplicación nfiq.

```
./nfiq fingerprint.jpg
```

1

Elaboración propia.

F.3. Conjunto de datos: nfiq.csv

El conjunto de datos nfiq.csv se genera mediante el script mostrado en la Figura N° F.2 luego de que todas las imágenes se copiaran a un directorio.

El método de captura de cada huella puede extraerse a partir del nombre del archivo. Cada nombre de archivo sigue el formato: <método-de-captura>-<sujeito>-<código-dedo>-<otros-datos>.jpg. La Figura N° F.3 muestra las primeras filas del conjunto de datos en formato CSV.

Figura N° F.2: Script bash para calcular los índices de calidad de las imágenes de un directorio.

```
#!/bin/bash
for file in *.jpg ;
do echo "$file" ;
echo -n "$file," >> nfiq.csv ;
./nfiq $file >> nfiq.csv ;
done
```

Elaboración propia.

Figura N° F.3: Estructura del conjunto de datos nfiq.csv.

```
sensor-P1-F10-...jpg,2
...
```

Elaboración propia.

F.4. Post-procesamiento del conjunto de datos

El conjunto de datos final contiene los campos: “nfiq”, que representa el índice de calidad; y, “group”, que representa el método de captura de la huella. Estos campos son generados ejecutando el script mostrado en la Figura N° F.4.

Figura N° F.4: Script en R para generar el conjunto de datos de índices de calidad.

```
csv = read.csv(file="nfiq.csv", header=FALSE, sep=",")

data = data.frame(
  group=str_extract(csv$V1, '[a-z]+'),
  nfiq=csv$V2)
```

Elaboración propia.

Finalmente, el conjunto de datos final tiene la estructura de la Tabla N° F.1.

Tabla N° F.1: Estructura del conjunto de datos de índices de calidad.

group		nfiq
sensor		1
...		...

Elaboración propia.

ANEXO G. Elaboración del conjunto de datos de puntajes de comparación

Este conjunto de datos se construye a partir de las 195,961 comparaciones realizadas entre las 184 imágenes capturadas con cámara y 321 imágenes capturadas con sensor, en la que cada comparación es agrupada según la proveniencia de las dos imágenes: cámara-cámara, cámara-sensor, y sensor-sensor. De esta forma cada grupo contendría el número de puntajes de comparación que se muestra en la Tabla N° G.1. El puntaje de comparación representa el número de detalles que podrían coincidir y que es obtenido por el algoritmo Bozorth del NIST (NIST NBIS-EC Userguide, 2019).

Tabla N° G.1: Número de puntajes de comparación agrupados según la fuente de las imágenes: cámara-cámara, cámara-sensor, y sensor-sensor.

Grupo	# puntajes	
cámara-cámara	33,856	184 x 184 puntajes de comparación
cámara-sensor	59,064	184 x 321 puntajes de comparación
sensor-sensor	103,041	321 x 321 puntajes de comparación

Elaboración propia.

G.1. Preparación de las imágenes capturadas

Las plantillas XYT de todas las imágenes capturadas son extraídas (ANEXO E) y colocadas en un directorio.

G.2. Cálculo del puntaje de comparación de cada par de imágenes

Se usó la aplicación bozorth3 del paquete NBIS con los argumentos mostrados en la Figura N° 3.5. La salida de esta aplicación es un número o, puntaje de comparación crudo, que representa la cantidad de detalles que pueden coincidir. La Figura N° G.1 muestra el resultado para dos plantillas de la misma persona.

Figura N° G.1: Puntaje de comparación obtenido al ejecutar la aplicación bozorth3.

```
./bozorth3 -m1 -b template1.xyt template2.xyt
```

```
124
```

Elaboración propia.

G.3. Conjunto de datos: results.csv

El conjunto de datos results.csv se genera ejecutando el script mostrado en la Figura N° G.2 sobre el directorio en el que se encuentran todas las plantillas XYT.

Figura N° G.2: Script bash para calcular los puntajes de comparación de las plantillas XYT de un directorio.

```
#!/bin/bash
for t1 in *.xyt ;
do for t2 in *.xyt ;
do echo "$t1 vs $t2" ;
echo -n "$t1,$t2," >> results.csv ;
./bozorth3 -m1 -b $t1 $t2 >> results.csv ;
done ;
done
```

Elaboración propia.

El método de captura de cada huella puede extraerse a partir del nombre del archivo. Cada nombre de archivo sigue el formato: <método-de-captura>-<sujeto>-<código-dedo>-<otros-datos>.jpg.xyt. La Figura N° G.3 muestra las primeras filas del conjunto de datos en formato CSV.

Figura N° G.3: Estructura del conjunto de datos results.csv.

```
cam-P1-F10-...jpg.xyt,cam-P1-F10-...jpg.xyt,1611
cam-P1-F10-...jpg.xyt,cam-P1-F10-...jpg.xyt,109
cam-P1-F10-...jpg.xyt,cam-P1-F10-...jpg.xyt,124
...
```

Elaboración propia.

G.4. Post-procesamiento del conjunto de datos

El conjunto de datos final contiene los campos: “group” que representa el método de captura de cada huella (cámara-cámara, cámara-sensor, sensor-sensor);

“truth_n”, donde 1 significa que ambas huellas son de la misma persona y 0 que son de personas distintas; “score”, que es el puntaje de comparación en crudo; y, “score_norm2”, que representa el puntaje de comparación normalizado en el rango [0-1] según las ecuaciones de la Figura N° 3.6 y Figura N° 3.7. Estos campos son generados ejecutando el script mostrado en la Figura N° G.4.

Figura N° G.4: Script en R para generar el conjunto de datos de puntajes de comparación.

```

csv = read.csv(file="results.csv", header=FALSE, sep=",")

data = data.frame(
  type1=str_extract(csv$V1, '[a-z]+'),
  subject1=str_extract(csv$V1, 'P[0-9]+-F[0-9]+'),
  type2=str_extract(csv$V2, '[a-z]+'),
  subject2=str_extract(csv$V2, 'P[0-9]+-F[0-9]+'),
  samefile=(csv$V1==csv$V2),
  score=csv$V3)

data$truth_n=ifelse(data$subject1==data$subject2, 1, 0)
data$group = apply(as.matrix(data[,c("type1","type2")]), 1, paste,
collapse="-")

dataset = data[!data$samefile, c("group","truth_n","score")]

softmax = function(x) (
  return(ifelse(x==0, 0.5, 1/(1+exp(-x))))
)

cam_cam = dataset[dataset$group=="cam-cam",c("group","truth_n","score")]
cam_cam$score_norm = (cam_cam$score-mean(cam_cam$score))/sd(cam_cam$score)
cam_cam$score_norm2 = sapply(cam_cam$score_norm, softmax)

# y así para los conjuntos de datos: cam-sensor, sensor-sensor, raw-sensor
    
```

Elaboración propia.

Finalmente, el conjunto de datos final tiene la estructura de la Tabla N° G.2.

Tabla N° G.2: Estructura del conjunto de datos de puntajes de comparación.

group	truth_n	score	score_norm2
cam-sensor	1	36	0.9388
cam-sensor	1	37	0.9452
...

Elaboración propia.

ANEXO H. Costos de sensores biométricos a julio del 2019

A continuación se listan los costos de algunos dispositivos biométricos homologados por el FBI como PIV SINGLE FINGER CAPTURE DEVICES². Estos dispositivos fueron probados y cumplen con las especificaciones de calidad de imagen Next Generation Identificación (IQS). Los costos fueron obtenidos de los principales proveedores de internet, están en su moneda original y no incluyen gastos de envío.

Tabla N° H.1: Costos de sensores biométricos a Julio del 2019.

Actualizado al 04/07/2019	MercadoLibre	Amazon	AliExpress
CrossMatch Verifier 300 LC 2.0		\$ 497.99	
CrossMatch EF200	S/. 400.00		
Futronic FS10	S/. 982.00		
Futronic FS88	S/. 690.00		
Futronic FS88H	S/. 524.00	\$ 69.50	
Futronic FS88HS	S/. 895.00		
Blucoil Secugen Hamster Pro 20		\$ 72.99	
Safran Morpho MSO Finger VP		\$ 450.00	
Safran Morpho MSO 1300 E3		\$ 112.00	
Sagem MSO1350	S/. 680.00		
Sagem MSO301	S/. 2,700.00		
Sagem MSO300	S/. 800.00		
SecuGen Hamster Pro 20	S/. 494.00	\$ 77.99	
Digital Persona U.are.U 4500	S/. 236.00	\$ 72.50	\$ 70.00
Suprema BioMini Slim 2		\$ 155.00	
Suprema BioMini Plus 2		\$ 279.00	
SecuGen Hamster Plus	S/. 544.00	\$ 84.99	
SecuGen Hamster Pro	S/. 289.00		
EikonTouch 700		\$ 68.99	

Elaboración propia.

² Lista de dispositivos homologados <https://www.fbibiospecs.cjis.gov/Certifications>