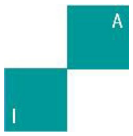
Inteligencia Artificial 26(71), 75-113
doi: 10.4114/intartif.vol26iss71pp75-113



INTELIGENCIA ARTIFICIAL

http://journal.iberamia.org/

A-Survey: Identification and Classification of Fingerprints via the Extreme Learning Machine Algorithm

David Zabala-Blanco [1, A], Diego Martinez-Pereira [1], Marco Flores-Calero [2], Jayanta Datta [3], Ali Dehghan Firoozabadi [4]
[1] Universidad Católica del Maule, Talca, Chile
[A] dzabala@ucm.cl
[2] Universidad de la Fuerzas Armadas-ESPE, Sangolquí, Ecuador
[3] Universidad de Chile, Santiago de Chile, Chile
[4] Universidad Tecnológica Metropolitana, Santiago de Chile, Chile

Abstract The fingerprint comes to be the most popular and utilized biometric for identifying persons owing to its bio-invariant characteristic, precision, as well as easy acquisition. A sub-system of an identification system is the classification stage in order to diminish the penetration rate and computational complexity. Actually, there are many formal investigations regarding techniques by exploiting convolutional neural networks (CNN) together with fingerprints images, which have superior performance metrics at the cost of large training times even employing high performance computing, which is not feasible in the standard word. In our manuscript, researches about identify and classify fingerprint databases by recurring to extreme learning machines (ELM) will be extensively reported and discussed for the first time. The diverse methodologies (ELM plus feature extractors) given by the authors will be studied and contrasted considering performance analysis. Consequently, academic papers with diverse version of ELMs are developed to observe the pros and cons that they exhibit with each other and to probe how they may help for minimizing the penetration rate of fingerprint databases. In fact, this issue is very relevant because by enhancing the penetration rate means shorting search times and computational complexity in fingerprints.

Keywords: Extreme learning machines, Fingerprint databases, Identification system, Classification subsystem

Resumen La huella dactilar es uno de los rasgos biométricos más populares y empleados en la identificación de personas, debido a su característica bioinvariante, precisión y por su fácil adquisición. Una de las etapas en la identificación de huellas dactilares es la clasificación, esta tiene el objetivo de reducir los tiempos de búsquedas y el costo computacional en las bases de datos. Actualmente, existen varias publicaciones académicas con métodos basados en redes neuronales convolucionales profundas (CNN) utilizando imágenes de huellas dactilares como entradas, los cuales tienen un excelente desempeño en cuanto a la clasificación, sin embargo, estos tienen un costo computacional muy elevado, y requieren tecnología de ultimo nivel, la cual no es accesible para cualquier persona. En este trabajo se revisarán propuestas de identificadores y clasificadores de huellas dactilares empleando máquinas de aprendizaje extremo (ELM) por primera vez. Se analizarán los métodos propuestos por los autores, para luego comparar el rendimiento con los distintos clasificadores considerados por los autores en sus respectivos trabajos. Se consideran trabajos con distintos tipos de ELM, con el objetivo de ver las ventajas y desventajas que presentan unas con otras, y verificar como estas pueden aportar en la reducción de la tasa de penetración de bases de datos de huellas dactilares. Esto último es importante, debido que mejorar la tasa de penetración, implica reducir los tiempos de búsqueda y complejidad computacional en las bases de datos de huellas dactilares.

Palabras Clave: Máquinas de aprendizaje extremo, bases de datos de huellas digitales, sistema de identificación, sistema de clasificación

ISSN: 1137-3601 (print), 1988-3064 (on-line)

1 Introducción

La biometría son las medidas biológicas, o características físicas, que se pueden utilizar para identificar a las personas. Una de las técnicas para identificar a una persona es la huella dactilar, principalmente por su característica bioinvariante, es decir, posee información que no cambia durante el periodo de vida de un individuo.
La huella dactilar ha llamado la atención frente a otros tipos de técnicas biométricas (rostro, voz, sangre, ojos, entre otras) debido a su bajo costo de obtención y por su alta precisión. Las aplicaciones del reconocimiento de huellas dactilares se enfocan en la seguridad, como por ejemplo en el registro civil y forense, del mismo modo es una forma para la autentificación del usuario que se suele emplear en alarmas, patrones de smartphones, claves de seguridad, etc. La huella dactilar puede identificarse o verificarse, para esto se requiere una base de datos en donde si el usuario desea identificar una huella dactilar, esta se comparará con todas las huellas dactilares presentes en la base de datos, mientras que, si se desea verificar, la huella dactilar se compara con la misma huella presente en la base de datos.

El procesamiento para identificar una huella dactilar puede ser muy costoso a nivel computacional y tomar un tiempo considerable, por lo tanto, se han adoptado diferentes categorías según las características de los patrones de cresta de la huella dactilar. Las huellas digitales se clasifican en 5 tipos según la morfología de los relieves de la superficie del dedo: arco, bucle izquierdo, bucle derecho, arco carpa y espiral [1], ver Figura 1. Como se puede ver, las clases de huellas digitales no son uniformes, siendo 3 muy ocurrentes y dos muy escasas, lo que dificulta el problema de clasificación en particular. La clasificación de huellas se utiliza en sistemas de reconocimiento para agrupar muestras de la misma clase y disminuir el tiempo búsqueda en la base de datos biométrica.

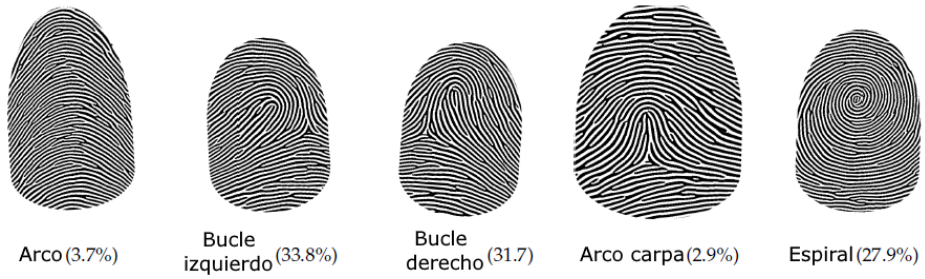


Figura 1. Muestras de imágenes de huellas dactilares, que representan las cinco categorías de clasificación incluyendo la frecuencia de aparición en la población total [1].

Durante las últimas dos décadas múltiples métodos para la clasificación o identificación se han publicado, siendo los más recientes métodos basados en redes convolucionales profundas (CNN, por sus siglas en inglés, forma de abreviación empleada a lo largo del trabajo), las cuales han logrado excelentes resultados en términos de desempeño, pero con la gran desventaja de que estos requieren un costo computacional excesivo y necesitan invertir mucho tiempo, incluso si se utiliza computadores de alta prestancia. Una de las alternativas que se presenta en la actualidad para la clasificación/identificación de huellas dactilares es aplicando algoritmos basados en máquinas de aprendizaje extremo (ELM). Las ELM son algoritmos para la red neuronal de una sola capa oculta (SHLNN), donde los pesos de la capa de entrada y los sesgos de la capa oculta se generan aleatoriamente. Es así que los pesos de la capa de salida se calculan resolviendo un sistema lineal gracias a la matriz pseudoinversa de Moore-Penrose. Se ha demostrado que las ELM son competentes en términos de exactitud, y se destacan por su proceso de entrenamiento rápido y estable, además de que su implementación es relativamente accesible para todo público.

A pesar de que las ELM son una excelente alternativa para enfrentar el problema de velocidad y costo computacional en la clasificación de huellas dactilares, la realidad es que la distribución de clases huellas dactilares no es uniforme (existen bases de datos ideales, pero no se deberían considerar para demostrar resultados reales), y esto representa un problema para las ELM, debido a que puede mostrar una tendencia hacia las clases mayoritaria en los resultados. Una solución para este problema es emplear las ELM ponderadas (W-ELM), ya que esta incorpora pesos de correlación de ELM en función de los errores de entrenamiento en el mismo proceso, y esta se ha comprobado previamente que este tipo de ELM tiene mejores resultados cuando los datos de entrada no son uniformes como es el caso de las huellas dactilares, sin embargo, al ser un algoritmo más robusto, el costo computacional aumenta levemente.

Un reconocimiento automático de personas basado en huellas dactilares requiere la coincidencia de una huella digital de entrada con una gran cantidad de huellas digitales en una base de datos. Sin embargo, la base de datos puede ser enorme (por ejemplo, la base de datos del FBI contiene más de 70 millones de huellas dactilares), tal tarea

puede ser muy costosa en términos de carga computacional y tiempo. La clasificación de huellas dactilares es el enfoque más común para reducir la tasa de penetración de la base de datos de un sistema de identificación de huellas dactilares. En este trabajo se busca revisar artículos académicos relacionados a la clasificación e identificación de huellas dactilares, mediante el uso de ELM, con el objetivo de comprobar cómo estas influyen en la reducción de la tasa de penetración en las bases de datos de huellas dactilares.

2 Marco Teórico

En esta sección se describirán los tipos de ELM reportados en la literatura que se ha revisado, vale decir, la ELM básica, la ELM regularizada, la ELM ponderada, la ELM ponderada por decaimiento y la ELM variacional bayesiana. De igual forma se describirán los extractores de características y los métodos que permiten obtener información de huella dactilar empleados en los trabajos revisados. Toda esta sección está particionada por las investigaciones que fueron encontradas en el estado de arte respecto al problema de identificación y clasificación de personas por huellas digitales, considerando diferentes bases de datos (SPRINGER, SCOPUS, IEEE, MPDI, otras) y artículos tanto de conferencias como de revista.

2.1 Extractores de características

Se describirán los extractores de características empleados en cada trabajo revisado. El extractor de características o descriptor tiene la función de extraer características de las imágenes de huellas dactilares. El objetivo final que tienen es representar la huella digital como una entrada para el clasificador.

2.1.1 Clasificación de huellas digitales a través de ELM estándar y desbalanceada [1]

Según Feng y Jain [2], hay tres categorías de representación de características de huellas dactilares: global, local y de gran detalle. Aunque, solo se utilizan descriptores de características globales porque las clases de huellas se definen por estas [3]. Las características de enfoque son las orientaciones de las crestas y las representaciones de puntos singulares. Las orientaciones de las crestas representan un mapa de orientación (OM), que representa el flujo de las crestas. Las ubicaciones de con cambios de flujo de crestas se seleccionan como puntos singulares, siendo núcleos y deltas los tipos más conocidos. Cada clase de huella se puede definir en función de la distribución de las orientaciones de sus crestas y los puntos singulares [3].

La extracción de OM es el primer paso en cualquier sistema de clasificación de huellas dactilares basado en características. Las representaciones basadas en OM se obtienen como una descripción de flujo de cresta local para cada bloque en huella digital. El OM de una muestra de huella de $U \times V$ pixeles es una matriz de $\frac{U}{u} \times \frac{V}{v}$ calculado para bloques de orientación de $u \times v$. La matriz OM almacena los ángulos expresados en radianes en un rango de $[0, \pi)$ o $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right)$. Cuando se obtiene el OM, se usa para detectar puntos singulares analizando el comportamiento de las crestas [4].

Galar y col. [3] presentan una taxonomía refinada de los métodos de extracción de características. Clasificaron los extractores en cuatro categorías: imagen de orientación, puntos singulares, flujo de línea de cresta y respuestas de filtro de Gabor. Para complementar el método propuesto, se utilizan los mejores extractores de características globales reportados en la literatura [5, 6].

- **Capelli02** [7], basado en OM de la huella dactilar. El enfoque registra el punto central mediante el método de Poincaré [8]. Luego la huella se representa mediante un vector de cinco posiciones, calculado aplicando un conjunto de máscaras dinámicas derivadas directamente de cada clase. Este vector de característica almacena las orientaciones.
- **Hong08** [9], este descriptor mejora el vector de características FingerCode [10] mediante la inclusión de información de seguimiento de crestas y puntos singulares. También codifica la posición y la distancia entre los puntos finales de la pseudo-cresta en relación con el punto del núcleo principal.
- **Liu10** [11], representa la huella dactilar mediante la construcción de un vector de características basado en las medidas relativas entre los puntos singulares. Los puntos singulares se detectan calculando respuestas de filtro complejas a múltiples escalas [12]. Un vector de características consta de la posición relativa, la dirección y las certezas de cada punto singular para cada escala.

2.1.2 Clasificación de huellas dactilares naturales usando el histograma de gradiente como descriptor [13]

En este articulo académico se considera el histograma de gradientes orientados (HOG), un extractor de características que codifica patrones de texturas y se utiliza para representar imágenes en muchas aplicaciones [14]. Para esto, HOG divide en bloques la huella dactilar y calcula el histograma del campo de orientación de cada bloque. HOG utiliza el gradiente basado en puntos para calcular el campo de orientación de cada bloque, que no es robusto contra el ruido y no se adapta a los patrones de las crestas que la huella dactilar posee. El rendimiento de los sistemas de identificación se puede ver afectado por la orientación de las crestas, por este motivo, el autor [13] propone una mejora del HOG para capturar el campo de orientación de la cresta de mejor forma y esta forma favorecer la identificación de la huella dactilar.

El autor del artículo realizó una mejora del extractor de características respecto al original. Las modificaciones implican que el HOG sea más resistente al ruido y a la calidad de las huellas dactilares, además de mejorar la representación de imágenes de huellas dactilares dañadas en comparación al HOG original. En la figura 2, se puede apreciar el aporte realizado por el autor en el descriptor, captura el campo de orientación de la huella digital y crea un descriptor robusto, mientras que en la figura 3 se puede comparar una huella dactilar con el HOG original y el HOG modificado, en donde el campo de orientación del HOG modificado claramente enfoca y representa de mejor forma una huella dactilar.

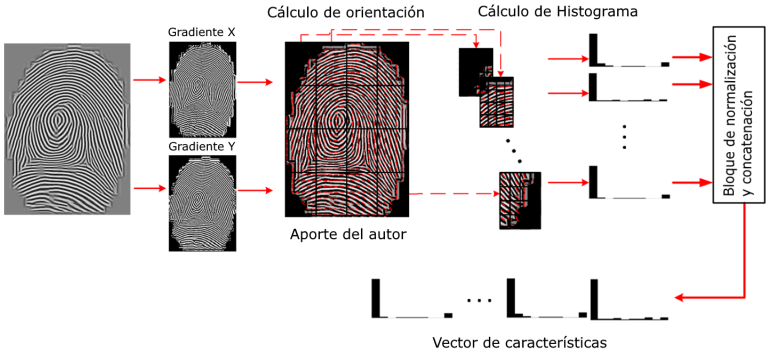


Figura 2: Esquema del funcionamiento del descriptor HOG [13].

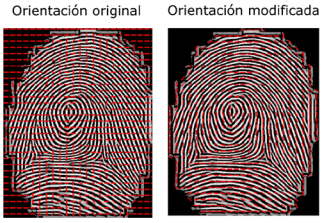


Figura 3: Comparación de huella dactilar con HOG original y HOG mejorado [13].

2.1.3 Identificación de huellas digitales basado en ELM [15]

En este trabajo se consideraron esencialmente dos extractores de características, los momentos geométricos y los momentos Zernike. Para representar una huella dactilar se extraen tres conjuntos de características de los momentos geométricos y Zernike, creando vectores con características combinadas de 3×17.

- **Análisis de momentos geométricos**, las características del momento geométrico (momento Hu) pueden proporcionar las propiedades de invariancia a escala, posición y rotación [16]. Esté análisis se utiliza para extraer características invariantes de celdas teseladas en una región de interés (ROI) de la huella dactilar. Para una función continua de dos dimensiones $f(x, y)$, el momento de orden $(p + q)$ se define como:

$$m_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q f(x, y) dx\, dy \;\; para\; p, q = 0,1,2 \ldots \qquad (1)$$

Un teorema de unicidad establece que si $f(x, y)$ es continuo por partes y tiene valores distintos de cero solo en una parte finita del plano xy, entonces existen momentos de todos los órdenes, y la secuencia de momentos (m_{pq}) esta determinada únicamente por $f(x, y)$.

Los momentos centrales se definen como:

$$\mu_{pq} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}(x-\bar{x})^p(y-\bar{y})^q f(x,y)dxdy \qquad (2)$$

donde $\bar{x} = m_{10}/m_{00}$ y $\bar{y} = m_{01}/m_{00}$. Si $f(x,y)$ es una imagen digital, entonces la ecuación (2) se convierte en:

$$\mu_{pq} = \sum\sum(x-\bar{x})^p(y-\bar{y})^q f(x,y) \qquad (3)$$

Y los momentos centrales normalizados, denotados por η_{pq}, donde:

$$\eta_{pq} = \mu_{10}/\mu_{00}^{\gamma}$$
$$\gamma = (p+q)/2 + 1 \text{ para } p+q = 2,3 \ldots \qquad (4)$$

Se puede derivar un conjunto de siete momentos invariantes del segundo y tercer momento [16]. El conjunto consta de grupo de expresiones de momento centralizadas no lineales y es un conjunto de invariantes de momento ortogonales absolutos que se pueden usar para la identificación de patrones invariantes a escala, posición y rotación de la siguiente forma:

$$\psi_1 = \eta_{20} + \eta_{02}$$
$$\psi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$
$$\psi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2$$
$$\psi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - 3\eta_{03})^2$$
$$\psi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$
$$\psi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} - \eta_{03})^2]$$
$$+ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2]$$
$$\psi_6 = (\eta_{20} - \eta_{02})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03})$$
$$\psi_7 = (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2] + (3\eta_{12} - \eta_{30})(\eta_{21} +$$
$$\eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \qquad (5)$$

- **Análisis de momentos Zernike**, las características del momento de Zernike pueden proporcionar las propiedades de invariancia a escala, posición y rotación [17]. Esté se utiliza para extraer características invariantes de múltiples ROI. Las magnitudes de los momentos de Zernike se han tratado como características invariantes de rotación, además tienen propiedades invariantes de traslación y escala mediante transformaciones geométricas simples [17]. Los polinomios radiales de Zernike de orden n con repetición m, $V_{nm}(x,y)$ puede escribirse como:

$$V_{nm}(x,y) = R_{nm}(x,y)e^{jm\theta} \qquad (6)$$

con:

$$j = \sqrt{-1}, \quad \theta = \tan^{-1}(y/x) \qquad (7)$$

y

$$R_{nm}(x,y) = \sum_{s=0}^{(n-|m|)/2} \times \frac{(-1)^S(x^2+y^2)^{(n-2s)/2}(n-s)!}{s!\,((n+|m|-2s)/2)!\,((n-|m|-2s)/2)!} \qquad (8)$$

donde $s = 0,1, \ldots, (n-|m|)/2$, $n \geq 0$, $|m| < n$, y $n - |m|$ es par. El ángulo θ se encuentra entre *0* y 2π y se mide con respecto al eje x en sentido antihorario. El origen del esquema de la coordenada es el centro de la imagen. Para una imagen digital, el momento Zernike de orden n y repetición m están dados por:

$$A_{nm} = \frac{n+1}{\pi}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\,V_{nm}^*(x,y) \qquad (9)$$

$V_{nm}^*(x,y)$ es el complejo conjugado de $V_{nm}(x,y)$. Una de las principales propiedades de los momentos de Zernike es que la imagen se puede reconstruir usando la transformada inversa:

$$\hat{f}(x,y) = \sum_{n=0}^{n_{max}}\sum_{m=-n}^{n} A_{nm}\,V_{nm}(x,y) \qquad (10)$$

donde n_{max} es el orden máximo de los momentos de Zernike considerados para una aplicación particular. Las magnitudes de los momentos de Zernike $|A_{nm}|$ son invariantes en rotación, igual pueden ser invariantes en traslación y escala. El momento Zernike $|A_{nm}|$ es de orden n con repetición m, donde n es un número no negativo y m es un numero entero sujeto a restricciones $n - |m| = par$, $|m| \leq n$. Se han encontrado que los momentos de Zernike de bajo orden son estables bajo transformaciones lineales, mientras que los

de orden alto tienen grandes variaciones. Por lo tanto, el orden n fue elegido para ser menos de cinco, y los primeros diez momentos de Zernike $(n \leq 5)$ pueden definirse como: $A_{0,0}, A_{1,1}, A_{2,0}, A_{2,2}, A_{3,1}, A_{3,3}, A_{4,0}, A_{4,2}, A_{4,4}, A_{5,1}$.

2.1.4 Reconocimiento biométrico multimodal basado en características visuales de fusión local y máquina de aprendizaje extremo bayesiano variacional [18]

En este artículo se utilizan tres extractores de características para obtener la información de la huella dactilar. Se considera el filtro de Gabor, los momentos de Zernike y la función de patrón binario local. Los momentos de Zernike se describieron en la sección previa, por ende, solo se describirán los otros dos extractores de características, los cuales posee una definición breve, debido a que el autor especifica que los extractores solo complementan un modelo fusión de características propio de la metodología del trabajo.

- **Filtro de Gabor,** se puede utilizar para aplicar el análisis de tiempo-frecuencia de la imagen de una huella dactilar, debido a su buena selectividad de dirección y selectividad de frecuencia. Para la extracción de las características de textura de la imagen que quiere analizar, en este caso se consideran varios bloques que dividen una imagen de huella dactilar. Para cada bloque de la imagen de la huella dactilar B_i el vector de características se denota como $Ft_G^i = [G_{0.0}, G_{0.1}, \dots, G_{2.1}]$.

- **Función de patrón binario local,** el operador de patrón binario local (LBP) posee características que lo hacen resistente a la luz y la expresión de la imagen. La huella dactilar se divide en bloques, y para cada bloque se extraen las características LBP utilizando un patrón uniforme con un vector de tamaño 59 [19]. El vector de características de la huella dactilar B_i se denota como Ft_L^i.

2.2 Extreme learning machines

Se describirán los clasificadores que fueron empleados en los diferentes trabajos revisados, tal como lo sugiere el título de este trabajo, todos los algoritmos se basan en ELMs. Por lo tanto, se comenzará describiendo el modelo básico, para luego continuar con las mejoras que se consideraron en los artículos revisados.

2.2.1 ELM original

La ELM es un algoritmo para SLFN de una sola capa oculta, destacando por su velocidad de aprendizaje y excelente rendimiento en generalización. Las ELM superan a las redes neuronales artificiales basadas en gradientes y máquinas de vectores de soportes en términos de predicción [20]. Dado un conjunto de entrenamiento con L muestras, la ELM mapea entradas (muestras de datos) y salidas (etiquetas) empleando solo una capa oculta compuesta por N nodos. Matemáticamente se representa [21, 22]:

$$H\beta = T$$

$$\begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \cdots & g(w_L \cdot x_1 + b_N) \\ \vdots & \ddots & \vdots \\ g(w_1 \cdot x_L + b_1) & \cdots & g(w_L \cdot x_L + b_N) \end{bmatrix} \begin{bmatrix} \beta_1^T \\ \vdots \\ \beta_N^T \end{bmatrix} = \begin{bmatrix} t_1^T \\ \vdots \\ t_L^T \end{bmatrix} \qquad (11)$$

donde H es la matriz de salida de la capa oculta, β denota la matriz de pesos de salida entre la capa oculta y capa de salida, T representa los resultados de salida de destino de la capa de salida, $g(\cdot)$ se refiere a una función continua por partes no lineal, como la función sigmoide, w_j es el vector de pesos de entrada entre en el nodo de entrada y el j-esimo nodo oculto, $x_i \in \mathbb{R}^n$ se refiere a los i-esimos datos de entrada donde n significa la dimensión de la capa de entrada, b_j representa el sesgo del j-esimo nodo oculto, β_j denota el vector de peso de salida entre los j-esimos nodos de salidas y neuronas ocultas, y $t_i \in \mathbb{R}^m$ es el vector objetivo m-dimensional originado por x_i. Además, w_j y b_j es el resultado de cualquier distribución de probabilidad continua, como la distribución rectangular, de esta forma es casi todo de forma automática. El termino $w_j \cdot x_i$ viene a ser el producto interno de w_j y x_i. En la figura 4 se puede observar la arquitectura global de una ELM con fines de compresión de las ecuaciones matemáticas.

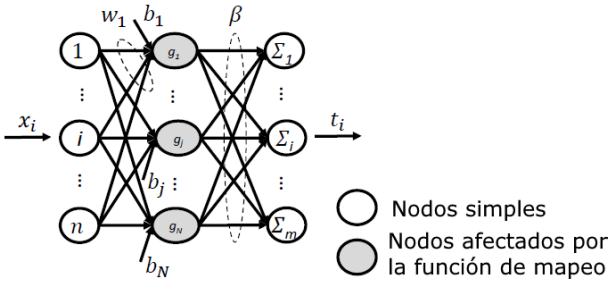


Figura 4: Arquitectura global de una ELM [1].

Se ha comprobado teóricamente que las SLFN con nodos ocultos aleatorios poseen la capacidad de aproximación universal y que los nodos ocultos se pueden generar aleatoriamente, sin importar los datos de entrenamiento. Por lo tanto, una vez que los nodos ocultos se hayan generado aleatoriamente y se proporcionan los datos de entrenamiento, la matriz de salida de la capa oculta H es conocida, y no necesita intervención. Entonces, para realizar el entrenamiento de la ELM equivale a obtener la solución de un sistema de lineal de los pesos de salida β.

Según la teoría de Barlett [23] para redes neuronales sin retroalimentación, para obtener un mejor rendimiento de generalización, la ELM debe intentar obtener el error de entrenamiento más pequeño $||\varepsilon||$, el cual esta sujeto a la siguiente ecuación:

$$\varepsilon = H\beta - T \qquad (12)$$

Entonces, bajo esa restricción, obtener la solución del sistema propuesta por Huang et. al. [20] es la siguiente:

$$\beta = H^{\dagger}T \qquad (13)$$

donde $H^{\dagger}$ es la inversa generalizada de Moore-Penrose de la matriz de la salida de la capa oculta H. Si los puntos de entrenamiento L son distintos, H es de rango de columna completo con probabilidad de 1 cuando $N \leq L$. Sin embargo, en aplicaciones reales el número de nodos ocultos siempre es menor que el número de puntos de datos de entrenamiento $N < L$. Por lo tanto:

$$H^{\dagger} = (H^T H)^{-1} H^T \qquad (14)$$

Finalmente, cabe mencionar que la función de activación o mapeo que afecta a los nodos debe ser escogida por el usuario, usualmente las que se utilizan son la sigmoide**,** umbral, coseno, seno, entre otras.
A continuación, se presentarán algunos tipos de ELM mejoradas, las cuales derivan del modelo básico, generalmente la variación se encuentra en el cálculo de β.

2.2.2 ELM regularizada

El riesgo real de predicción del aprendizaje consiste en el riesgo empírico y el riesgo estructural [24]. Un modelo con una buena capacidad de generalización tiene mejor compensación en ambos riesgos. Por ende, el riesgo real se puede representar mediante la suma ponderada los dos tipos de riesgos, y la proporción de estos se puede regularizar con un factor de ponderación C por riesgo empírico, este se representa por la suma del cuadrado del error $\|\varepsilon\|^2$, y el riesgo estructural representado por $\|\beta\|^2$, esto deriva de maximizar la distancia de las clases de separación de Margen [24]. Por lo tanto, el modelo matemático del algoritmo de la ELM regularizada (R-ELM) se puede describir:

$$\text{Minimizar: } \left\{\frac{C}{2}\|\varepsilon\|^2 + \frac{1}{2}\|\beta\|^2\right\} \qquad (15)$$

Sujeto a la ecuación (12), donde $\varepsilon = \varepsilon_1, \varepsilon_2, \ldots, \varepsilon_L$ son el error para L muestras, y es un parámetro de constante usado para ajustar el balance del riesgo estructural y riesgo empírico. Regulando C, se puede ajustar la proporción de riesgo empírico y riesgo estructural. El multiplicador lagrangiano de la expresión (15) se puede obtener de la siguiente forma [24, 25]:

$$\begin{aligned}\mathcal{L}(\beta, \varepsilon, \alpha) &= \frac{C}{2}\|H\beta - T\|^2 + \frac{1}{2}\|\beta\|^2 - \sum_{j=1}^{L}\alpha_j\left(\sum_{i=1}^{N}\beta_i K(\mathrm{a_i}x_j + \mathrm{b_i}) - t_j - \varepsilon_j\right) \\ &= \frac{C}{2}\|H\beta - T\|^2 + \frac{1}{2}\|\beta\|^2 - \alpha(\beta - T - \varepsilon) \qquad (16)\end{aligned}$$

donde $\alpha_j \epsilon R(j = 1,2, \ldots, N)$ es el multiplicador de Lagrange. Establecer los gradientes de este lagrangiano con respeto a $(\beta, \varepsilon, \alpha)$ igual a cero da las siguiente condiciones de optimalidad:

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial \beta} \rightarrow \beta^T = \alpha H \\ \dfrac{\partial \mathcal{L}}{\partial \varepsilon} \rightarrow C\varepsilon^T + \alpha = 0 \\ \dfrac{\partial \mathcal{L}}{\partial \alpha} \rightarrow H\beta - T - \varepsilon = 0 \end{cases} \quad (17)$$

Entonces la solución de la expresión (17) es la siguiente:

$$\beta = \left(\frac{I}{C} + H^T H\right)^{-1} H^T T \quad (18)$$

Esta expresión (18) solo involucra la inversión de una matriz de orden $N \times N$, donde $N \ll L$. Por lo tanto, la velocidad de cálculo de β es super rápida.

2.2.3 ELM ponderada

La mayoría de los algoritmos de aprendizaje se ven afectados por la distribución de clases, esto incluye a la ELM básica, por ende, esta se ve afectada por la distribución de clases [26]. El problema surge porque en muchas ocasiones se consideran entornos ideales en donde los datos están equilibrados, pero en casos donde los datos no son ideales y están desequilibrados existe la probabilidad de inclinación hacia las clases mayoritarias. La W-ELM se propone como una solución al problema mencionado.

Las muestras con errores de entrenamientos altos deben relacionarse con pesos pequeños y viceversa en el algoritmo ELM [27]. Según el teorema de Karush-Kuhn-Tucker, la solución para β toma la siguiente forma [28]:

$$\beta = \begin{cases} \left(H^T W H + \dfrac{I}{C}\right)^{-1} H^T W T & si\ L > N \\ H^T \left(W H H^T + \dfrac{I}{C}\right)^{-1} W T & De\ lo\ contrario \end{cases} \quad (19)$$

donde W es la matriz de costos de clasificación errónea. Es una matriz diagonal de $L \times L$ de acuerdo con la distribución de clases de la siguiente forma [26]:

$$Weighted\ ELM1: \quad W_{ii} = \frac{1}{N(t_i)}, \quad (20)$$

$$Weighted\ ELM2: \quad W_{ii} = \begin{cases} \dfrac{0.618}{N(t_i)} & si\ N(t_i) > media[N(t_i)] \\ \dfrac{1}{N(t_i)} & De\ lo\ contrario \end{cases}, \quad (21)$$

en donde $N(t_i)$ se refiere al numero de muestras en la clase t_i. En la ELM1 ponderada, los conjuntos de datos desequilibrados alcanzan un equilibrio cardinal. Para disminuir aún más los pesos de los datos de la clase mayoritaria, la ELM2 ponderada es la más adecuada, porque considera la proporción áurea. La compensación entre la ELM y la W-ELM1 viene dada por la W-ELM2.

2.2.4 ELM ponderada por decaimiento

Numerosas técnicas se han desarrollado para resolver la clasificación de datos desequilibrados en las ELM, como la W-ELM mejorada [29], la W-ELM neutrosófica mejorada [30], ELM basada en funciones de activación dual [31], W-ELM regularizada [26], entre otras. En términos de simplicidad y mejora, se destaca la W-ELM por decaimiento (DW-ELM) [27]. Para el aprendizaje de equilibrio y optimización, se agrega un grado extra de libertad en la W-ELM, que se conoce como parámetro de decadencia d. La matriz ponderada se puede escribir como [27]:

$$Decay\ weighted\ ELM: \quad W_{ii} = \frac{\sqrt[d]{\dfrac{N(t_i)}{\max[N(t_i)]}}}{N(t_i)} \quad (22)$$

A medida que el parámetro de decadencia aumenta, la clase minoritaria es más relevante que la clase mayoritaria. Lo que quiere decir, que mientras se varía el parámetro d se pueden obtener mejores posiciones de los limites en el clasificador. Si $d = 1$, la ELM no sufre cambios y se mantiene básica. También es necesario tener en mente que entre más compleja sea la ELM, aumenta el costo computacional como se puede apreciar al comparar las ecuaciones (18) y (19).

2.2.5 Algoritmo de generalización para las ELM

Las ELM descritas en las secciones 2.2.1-2.2.4 se pueden generalizar con el siguiente algoritmo.

Algoritmo 1: *Procesamiento de aprendizaje para la ELM.*

Dado un conjunto de entrenamiento $\Omega = \{(x_i, t_i) \mid i = 1, \dots, L\}$, la función de activación (mapeo) $g(\cdot)$, el parámetro de regularización C y el número de nodos en la capa oculta N

1: Asignar arbitrariamente los pesos de entrada w_j y los biases (sesgos) de los nodos ocultos b_j.

2: Calcular la salida de la matriz de la capa oculta H por x_i, se refiere a la matriz de la expresión (11).

3: Determinar los pesos de salida β. En la ELM básica de usa la ecuación (13), para la R-ELM se utiliza la ecuación (18), para la W-ELM se utiliza la expresión (19) donde los elementos de la matriz ponderada W estan dadas por las ecuaciones (20)-(22). Particularmente, la DW-ELM demanda el establecimiento del parámetro de decaimiento d.

2.2.6 ELM bayesiana variacional

La máquina de aprendizaje extremo bayesiana variacional combina el método Bayesiano y la estrategia estocástica en la ELM, donde se adopta el Bayesiano Variacional para resolver el problema de sobreajuste [32].

2.2.6.1 ELM bayesiana

Se introduce el bayesiano completo antes del vector de peso de salida β en el modelo de la ELM para evitar el problema de sobreajuste y aprovechar la tecnología aproximada variacional para calcular la distribución del vector de peso de salida β. Se discute desde el punto de vista probabilístico, donde el objetivo de la variable objetivo T satisface la distribución gaussiana con media de $y(x, \beta)$ y precisión (varianza inversa) ϕ.

$$H\beta = T$$

$$p(t|\beta) = \mathcal{N}(t|y(x, \beta), \phi^{-1}) \qquad (23)$$

La probabilidad condicional de los datos $\{x_i, t_i\}_{i=1}^{L}$ es:

$$p(t|\beta) = \prod_{i=1}^{L} p(t_i|\beta) \qquad (24)$$

Se elije una distribución previa conjugada gaussiana sobre el vector de peso β:

$$p(\beta|\alpha) = \prod_{j=1}^{N} \mathcal{N}\left(\beta_j|0, a_j^{-1}\right) \qquad (25)$$

donde $\alpha = [\alpha_1, \dots, \alpha_N]^T$ corresponde al vector de hiperparámetros individual que se asociado independientemente con el vector de pesos de salida $\beta = [\beta_1, \dots, \beta_N]^T$ respectivamente. También se agrega una distribución previa conjugada gamma sobre el hiperparámetros α para obtener un modelo bayesiano completo.

$$p(\alpha) = \prod_{j=1}^{N} Gam\left(\alpha_j|\, a_j^0, c_j^0\right) \qquad (26)$$

donde Gamma tiene la forma:

$$Gam(\alpha|a, c) = \Gamma(a)^{-1} c^a \alpha^{a-1} e^{-c\alpha} \qquad (27)$$

Con la función Gamma, $\Gamma(a) = \int_0^\infty t^{a-1} e^{-t} dt$ en el modelo completo bayesiano, la precisión de ϕ^{-1} es una constante que denota el ruido de los datos, y se le puede asignar un valor pequeño como $\phi^{-1} = 0.2$. Los parámetros para todo j pertenecientes a $[1, \dots, n]$, a_j^0, c_j^0 con valores pequeños, haciendo que la distribución previa de Gamma no sea informativa. Con base a las ecuaciones (23)-(27), la distribución conjunta de todas las variables (t, β, α) tiene la siguiente forma:

$$p(t,\beta,\alpha) = p(t|\beta)p(\beta|\alpha)p(\alpha) \qquad (28)$$

En el modelo completo extremo bayesiano, se asignan primero todos los pesos de entrada $\{w_j, b_j\}_{j=1}^{N}$ aleatoriamente como se muestra en la ELM para obtener un aprendizaje extremadamente rápido, y se calculan los pesos de salida $\beta = [\beta_1, \dots, \beta_N]^T$ utilizando los datos de entrenamiento $\{x_i, t_i\}_{i=1}^{L}$. Con base al modelo bayesiano completo anterior, el objetivo es calcular la distribución posterior del parámetro $p(\beta, \alpha|t)$ utilizando:

$$p(\beta,\alpha|t) = \frac{p(t,\beta,\alpha)}{p(t)}$$
$$p(t) = \iint p(t,\beta,\alpha)\, d\beta\, d\alpha \qquad (29)$$

2.2.6.2 Calculo variacional

En el modelo completo de la B-ELM la marginación sobre los parámetros β, α en la parte posterior de (29) es analíticamente es intratable. Se adapta el método variacional para aproximar la distribución posterior para hacer una predicción probabilística. Este marco proporciona una forma flexible de aproximar la distribución posterior en el modelo bayesiano. Se introduce una distribución libre factorizada $q(\beta, \alpha) = q(\beta)q(\alpha)$ para aproximar la distribución posterior $p(\beta, \alpha|t)$. El método de variacional proporciona fórmulas de iteración de la distribución aproximada que se basan respectivamente en la distribución probabilística conjunta $p(t, \beta, \alpha)$, lo que quiere decir:

$$\ln q*(\alpha) \propto E_{q(\beta)}[\ln p(t,\beta,\alpha)],$$
$$\ln q*(\beta) \propto E_{q(\alpha)}[\ln p(t,\beta,\alpha)], \qquad (30)$$

Las actualizaciones de las distribuciones posteriores se pueden expresar formalizadas. Las fórmulas de iteración de $q(\alpha_i)$, $j = 1, \dots, N$ son distribuciones Gamma:

$$q*(\alpha_j) = Gam(\alpha_j|a_j, c_j)$$
$$a_j = a_j^0 + \frac{1}{2}$$
$$c_j = c_j^0 + \tfrac{1}{2}E\left[\beta_j^2\right] \qquad (31)$$

Algoritmo 2: *Proceso de aprendizaje para la VB-ELM.*

Entradas: Conjunto de datos $\boldsymbol{\Omega} = \{\boldsymbol{x_i}, \boldsymbol{t_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{L}}$, con **N** número de nodos ocultos, la función de activación $g(\cdot)$.

Salidas: Distribución posterior de $\boldsymbol{\beta}$: $\boldsymbol{q(\beta) = \mathcal{N}(\beta, n, S)}$.

1: Asignar arbitrariamente los pesos de entrada w_j y los sesgos de los nodos ocultos b_j. $\{w_1, \dots, w_N\} = rand\ (B, N), \{b_1, \dots, b_N\} = rand\ (1, N)$;

$$H'_{L\times N} = \begin{bmatrix} x_1 \\ \vdots \\ x_L \end{bmatrix}_{L\times B} * [w_1, \dots, w_N]_{B\times N+} \begin{bmatrix} b_1 & \cdots & b_N \\ \vdots & \ddots & \vdots \\ b_1 & \cdots & b_N \end{bmatrix}_{L\times N};$$
$$H'_{L\times N} = H(H')$$

2: Aprender los pesos de **salidas** $\boldsymbol{\beta}$ con la fórmula de iteración variacional (31) y (32).

Inicialización de la varianza inversa con $\phi^{-1} = 0.2$
Inicialización de los hiperparámetros aleatorios con
$\forall j \in [1, \dots, N], a_j^0, c_j^0 = 10^{-4} * rand(1)$.
For $i = 1$ hasta $\left|a^i\text{-}a^{i-1}\right| < 10^{-4}$ **do**
$S = (A + \phi H^T H)^{-1}$,
$n = \phi S H^T t$,
For $j = 1$ hasta N **do**
$a_j = a_j^0 + \frac{1}{2}$,
$c_j = c_j^0 + \frac{1}{2} n(j) n(j) + S(j, j)$,
end
$A = diag\left(\frac{a_1}{b_1}, \dots, \frac{a_N}{c_M}\right)$,
end
Salida $q(\beta) = \mathcal{N}(\beta | n, S)$.

donde $E_{q(\beta)}\left[\beta_j^2\right] = n(j)n(j) + S(j, j)$. Las fórmulas de iteración de $q(\beta)$ es una distribución Gaussiana *N*-dimensional, es decir:

$$q(\beta) = \mathcal{N}(\beta | n, S)|,$$
$$S = (A + \phi H^T H)^{-1},$$
$$n = \phi S H^T t, \quad (32)$$

donde:

$$A = diag\left(\frac{a_1}{b_1}, \dots, \frac{a_L}{c_N}\right)$$
$$H = \begin{bmatrix} g(w_1 \cdot x_1 + b_1) & \cdots & g(w_N \cdot x_1 + b_N) \\ \vdots & \ddots & \vdots \\ g(w_1 \cdot x_L + b_1) & \cdots & g(w_N \cdot x_L + b_N) \end{bmatrix}_{L\times N} \quad (33)$$

La interferencia variacional actualiza la distribución posterior aproximada $q(\beta)$, $q(\alpha)$ con las ecuaciones (31) y (32) alternativamente, hasta que se cumplan los criterios de convergencia.

Es importante mencionar que la interferencia variacional de VBELM tiene dos pasos de cálculo:

Elegir los vectores de pesos de entrada $\{w_1, \dots, w_N\}$ y el vector de biases $\{b_1, \dots, b_N\}$ aleatoriamente, luego calcular la distribución posterior de los pesos de salida β utilizando las fórmulas de iteración (31) y (32). La varianza del hiperparámetro ϕ^{-1} denota la incertidumbre de las muestras, que se inicializarse con una valor pequeño de 0.2. Los hiperparámetros $\forall j \in [1, \dots, N], a_j^0, c_j^0$ son los parámetros de los parámetros anteriores $\forall j \in [1, \dots, N], \alpha_j$. Se establecen hiperparámetros con valores pequeños aleatorios $\forall j \in [1, \dots, N], a_j^0, c_j^0 = 10^{-4} * rand(1)$, lo que hace que los previos $\alpha = \alpha_1, \dots, \alpha_N$ no presenten información por su valor tan pequeño y hace más estable al algoritmo de la VB-ELM con inicialización aleatorias.

2.2.6.3 Selección de nodo oculto

En el modelo completo bayesiano de la ELM, la distribución posterior del parámetro β y el hiperparámetro α son gaussianos y gamma, respectivamente. El vector $\alpha = \alpha_1, \ldots, \alpha_N$ se asocia con el vector de pesos de salida β, el cual determina los pesos de los nodos ocultos en el modelo BELM. Los parámetros a_i, b_i en (31), cuando el valor de a_i/b_i tiende a infinito, el i-ésimo elemento $S(i,i)$ en la matriz $S = (A + \phi H^T H)^{-1}$ tienden a cero, y la media correspondiente tiende a cero $n(i) \to 0$. Por ende, la distribución aproximada posterior $q = (\beta_i | t, \alpha)$ del parámetro β_i llega a tener un pico alto en cero, y el nodo oculto correspondiente β_i se puede podar del modelo.

2.2.6.4 Predicción probabilística

Estas se hacen con base en la distribución posterior aproximada $q(\beta)$. Dada una nueva entrada x^*, la distribución predictiva de t puede aproximarse con la distribución $p(t|\beta)$ en (23) y en la distribución posterior aproximada $q(\beta)$ de (32), lo que quiere decir:

$$p(t\,|x^*, t) = \int p(t\,|x^*, \beta) p(\beta|t) d\beta \approx \int p(t\,|x^*, \beta) p(\beta) d\beta \qquad (34)$$

Dado que ambos términos en la integración son gaussianos, la distribución predictiva también es una distribución gaussiana:

$$p(t\,|x^*, t) \approx \mathcal{N}\left(t \middle| \mu(x^*), \sigma^2(x^*)\right) \qquad (35)$$

en donde:

$$\mu(x^*) = n^T H(x^*),$$
$$\sigma^2(x^*) = \frac{1}{\phi} + H(x^*)^T S H(x^*), \quad (36)$$

3 Metodología

En sección se describirán las metodologías que emplearon en el proceso de identificación o clasificación de huellas dactilares en los trabajos que se revisaron. Formalmente, se revisaron en profundidad cinco trabajos, pero solo se consideran cuatro de ellos, debido a que uno de ellos corresponde a un ensayo.

3.1 Clasificación de huellas digitales a través de ELM estándar y desbalanceada [1]

Se describirá la metodología que empleó para la clasificación de huellas dactilares en el artículo [1], del mismo modo se presentara el tipo de evaluación que se llevó a cabo para evaluar el rendimiento de las ELM.

3.1.1 Base de datos de huellas dactilares

Se considero el software y base de datos SFINGE [33, 34], siguiendo la distribución natural de las clases (figura 1), la cual genera huellas dactilares de diferentes calidades y permite evaluar el rendimiento de los clasificadores. Se tuvieron en cuenta tres perfiles para emular varios escenarios, considerando los tres siguientes etiquetados:

Figura 5: Ejemplos de los tipos huellas dactilares generadas por el software SFINGE [1].

Como se puede apreciar en la figura 5, la calidad de las huellas dactilares en la base de datos SFINGE es diferente, esto se hace con el objetivo de apreciar el rendimiento del clasificador para cada caso que pueda presentarse. Se generaron en total 30000 huellas dactilares, siendo 10000 para cada perfil de calidad (HQNoPert, Default y VQAndPert).

3.1.2 Esquema de validación cruzada quíntuple

El método para evaluar los resultados consiste en una técnica que se enfoca en el aprendizaje automático, esta se conoce como aproximación de validación cruzada quíntuple [35]. Este esquema es el resultado de una medición imparcial y precisa del rendimiento del clasificador debido a que la capacitación y las pruebas no se desarrollan en partes fijas. La base de datos se divide en cinco partes, distribuidas uniformemente con las muestras, cada división se entrena utilizando el 80% de las huellas dactilares del resto de divisiones, y la prueba se realiza con la división actual. Para cada base de datos y clasificador, los resultados generales se informan a partir de un promedio de cinco ejecuciones. Para estimar los hiperparámetros óptimos de la ELM, se utiliza el 20% de cada conjunto de entrenamiento para fines de validación, por lo tanto, este, está destinado a encontrar los hiperparámetros de la ELM [36]. De manera de reforzar la explicación previa, se presenta la figura 6.

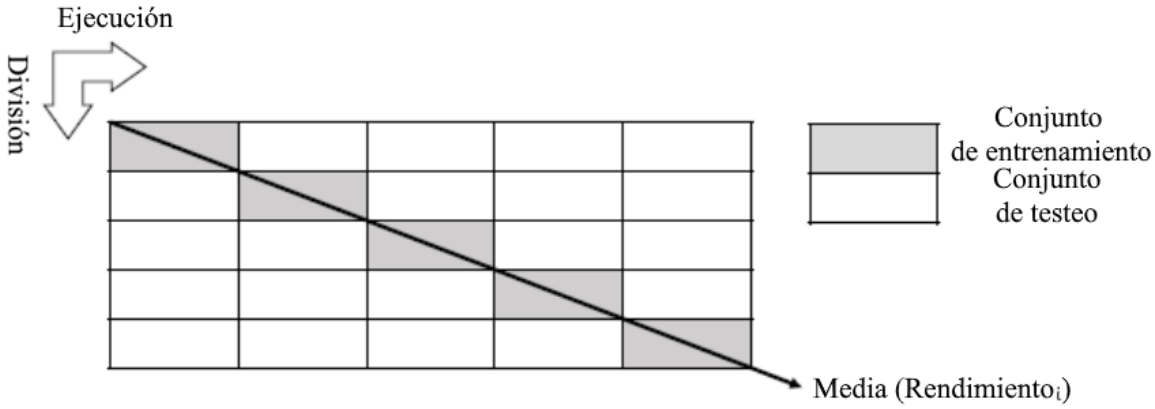


Figura 6: Representación del esquema de validación cruzada quíntuple, el conjunto de validación es descartado [1].

3.1.3 Métricas de rendimiento

Para evaluar la clasificación de huellas dactilares del nuevo enfoque (extractores de características junto con ELM) y los métodos comparativos basados en CNN [5, 37], la media geométrica (media G), el error cuadrático medio de la raíz (Precisión) y el error absoluto de las métricas de la tasa de penetración (PR) se utilizan como criterios de evaluación. Mientras que los valores de media G y precisión cercanos a 1 indican que el modelo correspondiente tiene mejor desempeño de clasificación, para un PR cercano a 0. Esta métricas se definen de la siguiente forma [38, 28, 39]:

$$Media\ G = \sqrt{\frac{TP}{TP+TN} \times \frac{TN}{TN+FP}}, \qquad (37)$$

$$Precisión\ (Acc) = \sqrt{\frac{1}{K}\sum_{i=1}^{K}(t_i - \overline{t_\iota})^2}, \qquad (38)$$

$$PR = \left| 0.2948 - \sum_{i=1}^{M} p_i[1 + Acc_i(p_i - 1)] \right|, \qquad (39)$$

donde *TP,TN, FP* y *FN* en la ecuación (15) representan respectivamente un verdadero positivo, verdadero negativo, falso positivo y un falso negativo en un problema de clasificación binaria para ejemplificar. Dicho en otras palabras, se puede interpretar como la raíz cuadrada de la precisión de la clasificación mayoritaria multiplicada por la precisión de la clase minoritaria. En el contexto de clasificación múltiple (*M*), la media G se convierte en la *M*-ésima raíz del producto de las precisiones dentro de cada clase, las cuales se denotan como Acc_i en la siguiente. En la ecuación (38), *K* denota el número de muestras de huellas, mientras que t_i y $\overline{t_i}$ denotan los valores reales y de

predicción del proceso de clasificación de huellas, respectivamente. Por último, se tiene que *M* significa el número de clases y p_i es la relación de la huella perteneciente a la *M*-ésima clase en la ecuación (39). Se adopta el error absoluto de la tasa de penetración (PR) para contrastar fácilmente los clasificadores. Las constante 0.2948 resulta de la tasa de penetración ideal siguiendo la distribución natural de las clases de huellas de SFINGE [33, 34].

La precisión calcula desviación total entre los valores reales y estimados, siendo criterio popular para evaluar el desempeño de los clasificadores. Aunque debido a que los datos son desequilibrados, es mejor utilizar media G, esto por la posible presencia de una clase significativa desequilibrada [26]. De hecho, la precisión se ve afectada por la distribución de clases y puede dar resultados engañosos, cosa que no sucede para la media G.

Finalmente, en la figura 7 se puede apreciar el esquema de resumen los pasos de la metodología para la clasificación de huellas digitales.

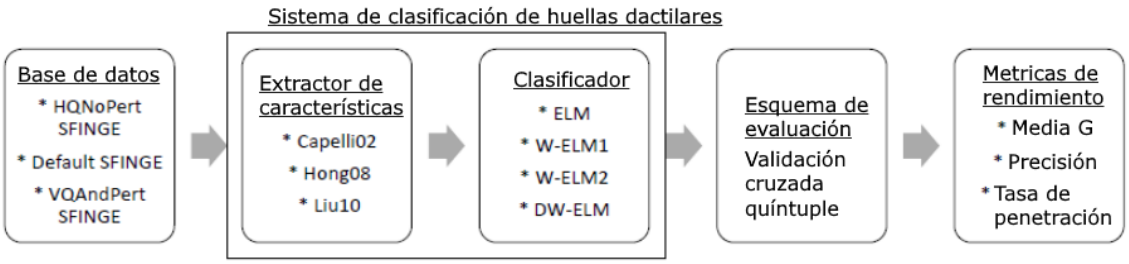


Figura 7: Esquema de la metodología donde se destaca el sistema de clasificación de huellas dactilares propuesto [1].

3.2 Clasificación de huellas dactilares naturales usando el histograma de gradiente como descriptor [13]

La metodología empleada por el autor en este trabajo se centra en el extractor de características principalmente, y la información que proporciona acerca del clasificador empleado, en este caso la ELM de kernel, es muy escasa y no se presenta teoría sobre esta. La contribución que se realiza es una modificación del extractor de características HOG.

3.2.1 Modificación del HOG

La huella se puede representar utilizando puntos minuciosos [40] o patrones de orientación de textura y cresta [41, 42, 43]. En la extracción de características se capturan los patrones de orientación de la cresta local utilizando el descriptor HOG. No se requiere la determinación de puntos singulares o minucias.

HOG es un descriptor de textura, basado en el campo de orientación y magnitudes de degradado de una imagen, para esto la imagen se divide en bloques. Se calcula un histograma para cada bloque utilizando su campo de orientación y magnitudes de gradiente, la concatenación de histogramas locales es el descriptor HOG, que representa la imagen. El proceso implica los siguientes pasos:

- La huella se normaliza mediante la transformación de la ley de potencia (ganmma) con ganmma.

- El gradiente de la huella normalizada se calcula utilizando $mask = [-1,0,1]$:

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} \qquad (40)$$

$$G_x = \frac{\partial f}{\partial x} \qquad (41)$$

$$G_y = \frac{\partial f}{\partial y} \qquad (42)$$

- El campo de la huella digital (θ) y magnitud del gradiente (mag) se calculan utilizando:

$$\theta = tan^{-1}(G_x, G_y) \quad (43)$$

$$mag = \sqrt{{G_x}^2 + {G_y}^2} \quad (44)$$

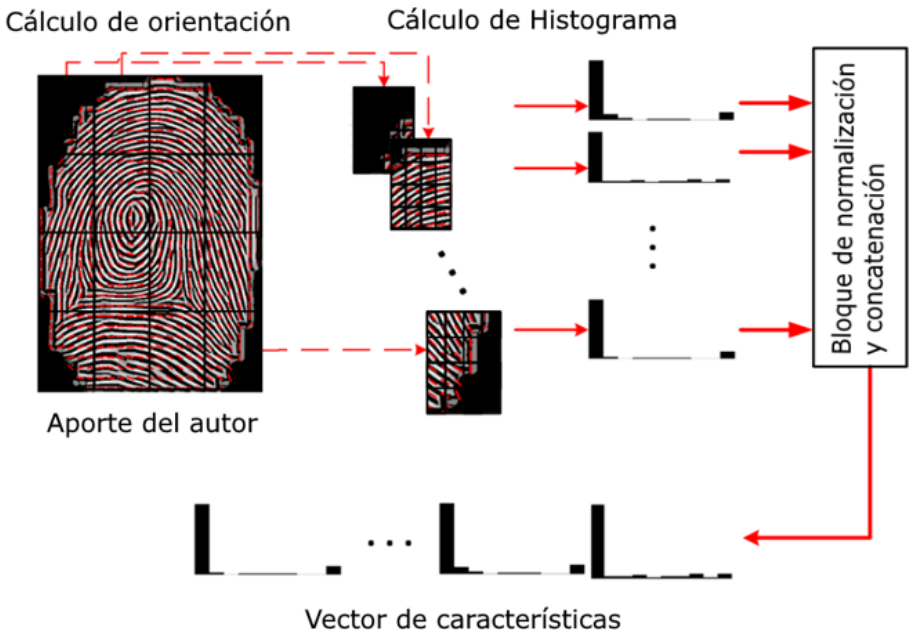


Figura 8: Calculo del descriptor del HOG modificado [13].

- La huella dactilar se divide en celdas. Se calcula un histograma local de orientaciones que contiene 9 contenedores de orientación para cada celda. Para cada píxel de una celda, el intervalo del histograma es seleccionado en función de su orientación y su voto se pondera con la magnitud del gradiente.
- La normalización del contraste se aplica utilizando bloques superpuestos de celdas de huellas dactilares para superar la variación en la iluminación y el contraste en el primer plano y el fondo. La imagen de la huella dactilar se divide primero en bloques de tamaño 16x16 pixeles, luego se crean parches de 4(2x2) celdas utilizando los bloques con una superposición del 50%. Para las cuatro celdas, los histogramas locales se calculan y concatenan, y finalmente el histograma concatenado se normaliza utilizando, usando L_2 normalización.

La orientación de la huella dactilar se basa en la ecuación (43), en donde el gradiente se calcula utilizando la mascara $[-1,0,1]$, y esta es sensible al ruido y la baja calidad de las huellas. Hong y otros [44], propusieron un método para calcular el campo de orientación de huellas dactilares, el cual es resistente al ruido, baja calidad y daños en la estructura de la huella dactilar. Para este método, se debe calcular la orientación de cada píxel y promediarla en función de las orientaciones de sus pixeles vecinos, y para superar el problema de resistencia al ruido se calculan las máscaras para el filtro de gradiente empleando las derivadas parciales de la función gaussiana. Mediante el método de orientación suavizada de Hong [44], se modificó el descriptor HOG como se puede apreciar en la figura 8, el procedimiento es el siguiente:

- Calcular los componentes horizontales y verticales empleando los siguientes filtros calculados usando la función 2D gaussiana:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (45)$$

$$G_x = \frac{-x}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (46)$$

$$G_y = \frac{-y}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (47)$$

Donde σ es la división estándar que especifica la amplitud de las funciones.

- Usando mascaras calculadas con las ecuaciones (46) y (47), calcular el gradiente de la huella dactilar.
- Utilizando la imagen de degradado, calcular la orientación de cada píxel tomando el promedio de su orientación y las orientaciones de los pixeles en un vecindario de 16×16 pixeles.

La orientación obtenida mediante este procedimiento es más resistente al ruido, la baja calidad y daños en las huellas dactilares, la comparación entre el HOG original y modificado se puede apreciar en la figura 3 de la sección 2.1.2.

3.2.2 Extracción de características mediante HOG modificado

Las huellas dactilares se representan utilizando el HOG modificado, esencialmente por su resistencia al ruido y calidad de huellas. La extracción implica los siguientes pasos:

- Dividir la huella dactilar en 16 (4×4) bloques.
- Calcular el HOG modificado para cada bloque h_i, $i = 1, \dots, 16$.
- Calcular la representación de la huella dactilar concatenando los descriptores h_i, $i = 1, \dots, 16$, es decir, $h = [h_1, h_2, \dots, h_{16}]$
- Normalizar h para cada valor en un rango de $[-1,1]$

Una vez h esta normalizada se utiliza como entrada para el clasificador, en este caso la ELM de Kernel (la cual no presenta teoría en el trabajo revisado [13]).

3.2.3 Modificación del HOG

Respecto a la ELM, los autores utilizaron arbitrariamente una función de activación radial sin considerar el problema de los datos desequilibrados, y no declaran el número de neuronas en la capa oculta, además la base de datos que utilizan, FVC-2004 [45], junta dos categorías de huellas (arco y arco tienda), aumentando la tasa de penetración y costo computacional. Se consideraron 4 subbases de datos de FVC-2004 con 880 huellas dactilares de 100 personas con 8 pulsaciones cada una, por lo tanto, el rendimiento se calcula para cada subbase de datos.

El rendimiento del método propuesto se midió considerando el siguiente procedimiento:

- Se dividen los datos aleatoriamente, 90% para entrenamiento y 10% para pruebas.
- Se utiliza el HOG modificado para extraer el vector de características para el entrenamiento y las pruebas de huellas dactilares.
- Se menciona el parámetro de Kernel y el coeficiente de regularización, los cuales se ajustan utilizando el 80% de los datos de entrenamiento (no se presentan mayor detalle).

Se calcula el promedio de precisión y división estándar, tras repetir el proceso anterior 50 veces, cada vez que los datos se dividen al azar. Después de ajustar los parámetros utilizando los daros de entrenamiento, se calcula precisión en función de los datos de prueba.

3.3 Identificación de huellas digitales basado en ELM [15]

Para este articulo académico se consideró la base de datos de huellas dactilares FVC2002 [46], la cual contiene 4 subbases, siendo DB1_a, DB2_a, DB3_a y DB4_a. Cada subbase de dato contiene 800 huellas dactilares de 8 bits en escala de grises, que pertenecen a 100 personas con 8 muestras cada una. La resolución de DB2_a es de 569 dpi, mientras que las otras subbases poseen una resolución de 500 dpi. A continuación, se describirá la metodología empleada para la identificación de huellas dactilares en el trabajo [15].

3.3.1 Descripción general del método propuesto

El método que se propone consta de tres etapas: preprocesamiento de imágenes de huellas dactilares, extracción de características y coincidencia de ELM (R-ELM), ver figura 9.

En el preprocesamiento se tiene la mejora de imagen, determinación de puntos de referencia y determinación del ROI.

En la extracción de características se tiene el análisis de momento invariante, combinación de análisis de momento invariante y análisis de componentes principales (PCA).

Para acelerar el proceso de general de extracción de características, se utiliza un área predefinida (ROI) alrededor del punto de referencia de la imagen de la huella, en vez de la huella completa. El centro de la imagen de ROI recortada es la posición del punto de referencia que se determinó en el paso anterior.

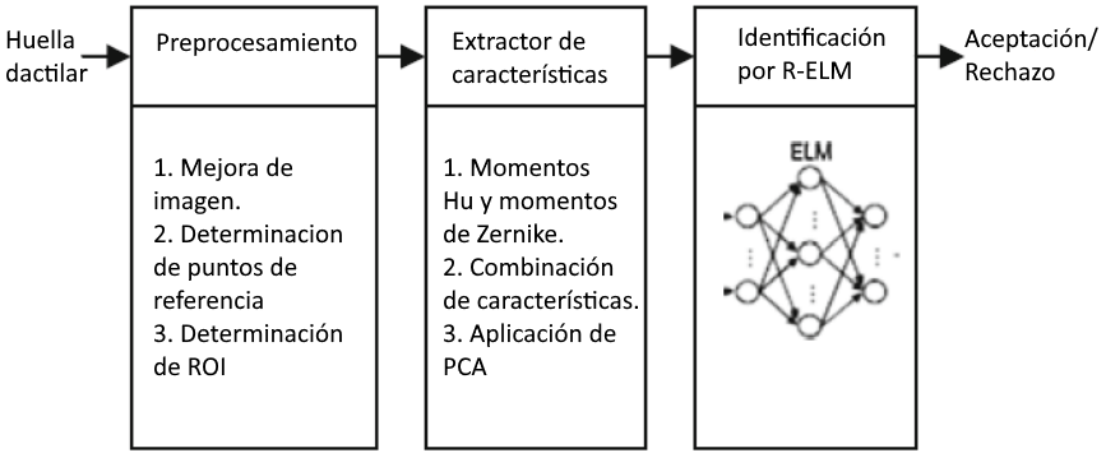


Figura 9: Descripción general del sistema de reconocimiento de huellas dactilares propuesto [15].

Se utilizan diferentes tamaños de ROI para optimizar la información de la huella. Se define un punto en común para todos los tipos de huelas dactilares, conocido como punto de la curvatura máxima en la cresta convexa, generalmente se encuentra en el área central de la huella [47]. El vector de características tiene 51 elementos y consta de tres conjuntos de momentos derivados de tres sub-ROI diferentes. Debido a que las tres sub-ROI se definen utilizando estructuras de crestas similares en una imagen y huella digital, las características obtenidas de estas tres sub-ROI pueden estar fuertemente correlacionadas. La dimensión del vector de características se reduce mediante PCA, que examina la matriz de covarianza de características y luego selecciona las características más distintas.

La última parte del sistema, el módulo de emparejamiento se implementó con las ELM para las etapas de entrenamiento y prueba.

En el entrenamiento de ELM, las imágenes de las huellas son procesadas primero por el módulo de preprocesamiento y el módulo de extracción de características luego sus características extraídas se almacenan como plantillas en la base de datos para uso posterior y para que la ELM aprenda el SLFN en el módulo correspondiente.

En la etapa de prueba, el módulo de preprocesamiento y el módulo de extracción de características procesan primero una imagen de la huella digital de un sujeto que se va a verificar, luego sus características extraídas se envían al módulo correspondiente con el ID de identidad del sujeto, que las compara con las plantillas correspondientes en la base de datos. Las dos etapas contienen los mismos módulos de preprocesamiento, extracción de características y coincidencia.

3.3.2 Preprocesamiento

3.3.2.1 Mejora de la imagen

El rendimiento depende de la calidad de imagen de la huella de entrada (claridad de estructura de cresta). Se mejora en dominio espacial como en el de la frecuencia, mediante un filtro de dos etapas [48]. Se usa la información de la primera imagen para estimar los parámetros con el segundo filtro.

3.3.2.2 Determinación del punto de referencia

Se determina con la imagen mejorada, el campo de orientación de la cresta se obtiene a partir de la imagen mejorada (precisión y fiabilidad). La determinación robusta y confiable de un punto de referencia se puede lograr detectando la curvatura máxima empleando métodos de filtrado complejos con múltiples resoluciones [12] aplicado a imágenes de campo de orientaciones de crestas. El método de resolución múltiple mejorar el error del campo de orientación con imágenes originales de mala calidad, ver Figura 10.

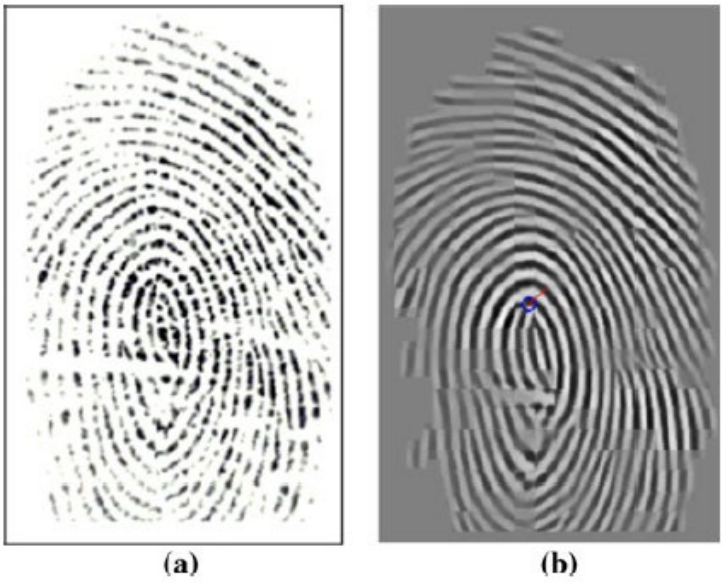


Figura 10: Ejemplo de una imagen de huella dactilar original y su referencia determinación de puntos: (a) una huella dactilar original (tamaño 296 × 560); (b) un punto de referencia que se muestra en la imagen mejorada de (a) [15].

3.3.2.3 Determinación del ROI

Se utilizan tres ROI de diferente tamaño $(64 \times 64, 32 \times 32\ y\ 16 \times 16)$ para extraer características para representar la huella.

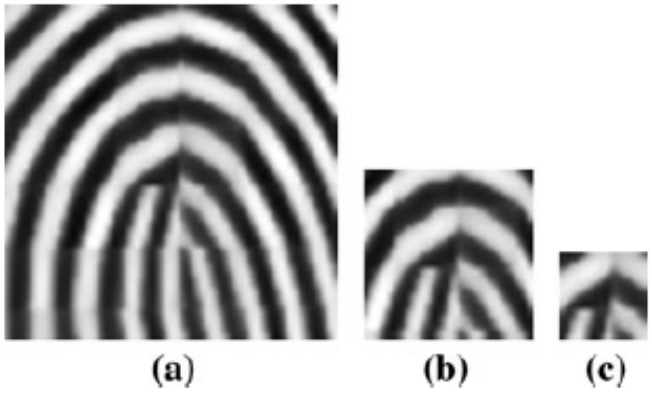


Figura 11: Región de interés de diferentes tamaño para la extracción de características. (a) corresponde a una ROI de tamaño 64×64, (b) corresponde a una ROI de tamaño 32×32 y (c) corresponde a una ROI de tamaño 16×16 [15].

Los tamaños de los ROI se determinan mediante experimentos; si los ROI son demasiado grandes, pueden extenderse fácilmente fuera del primer plano de la huella digital. En este experimento, 64×64 es el tamaño óptimo para el ROI más grande. El conjunto de momentos invariantes se calcula para cada ROI, para posteriormente extraer múltiples conjuntos de características de momento geométrico y momento Zernike. La región de interés de

diferentes tamaño para la extracción de características. (a) corresponde a una ROI de tamaño 64×64, (b) corresponde a una ROI de tamaño 32×32 y (c) corresponde a una ROI de tamaño 16×16.

3.3.3 Extracción de características

3.3.3.1 Momentos HU y momentos Zernike

Los momentos geométricos y momentos Zernike se calculan para cada sub-ROI. Se utiliza la imagen binaria en vez de la imagen en escala de grises. Cada sub-ROI en escala de grises de binariza aplicando umbrales locales en las regiones de acuerdo con la varianza local de las subimágenes.

3.3.3.2 Ensamblaje de características de momento invariante

Se extraen tres conjuntos de momentos invariantes que consisten en momentos geométricos y momentos Zernike como para características para representar una huella digital.

Se tiene $\phi_{k,l}, k = 1,2,3, l = 1,2,3, \dots,17$, como un elemento de los tres conjuntos de características, donde $\phi_{k,l}$ para , $l = 8,9,10, \dots,17$ consta de momentos Zernike, y , k es el índice de los sub-ROI.

Los vectores de características totales $V = [\phi_{1,1}, \phi_{1,2}, \dots, \phi_{1,17}, \phi_{2,1}, \phi_{2,2}, \dots, \phi_{3,17}]$ son combinaciones de los tres conjuntos de momentos geométricos y momentos de Zernike. La longitud de los vectores de características totales es de $3 \times 17 = 51$.

3.3.3.3 Selección de características mediante análisis de componentes principales

PCA es una técnica eficaz que permite reducir la dimensión de un conjunto de datos conservando la mayor cantidad posible de variación en el conjunto de datos [49].

Supongamos que X es un conjunto de datos $\{x_1, x_2, x_3, \dots, x_n\} \in R^{u \times N}$ con N muestas de entrenamiento, donde x_i es un vector de columna que corresponde a una imagen y u es la longitud del vector de características.

3.3.3.4 Coincidencia de ELM

El proceso de clasificación puede verse como un problema de dos clases. Existen dos fases para la clasificación de huellas dactilares con ELM, fase de entrenamiento y fase de prueba. En la fase de entrenamiento, en cada entrada se envían los vectores de características de las huellas la ELM para obtener los parámetros del modelo ELM. En la fase de prueba se utiliza la ELM con los parámetros entrenados para verificar la coincidencia entre los vectores de características de la imagen de la huella digital de prueba y los de la imagen de la plantilla, y la salida de la ELM es la más cercana clase de salida. Básicamente este proceso consiste en aplicar el algoritmo de la R-ELM descrito en la sección 2.2.2.

3.4 Reconocimiento biométrico multimodal basado en características visuales de fusión local y máquina de aprendizaje extremo bayesiano variacional [18]

La metodología presentada en este articulo incluye cuatro pasos: preprocesamiento de imágenes, extracción de características locales, modelado de características visuales de fusión local y reconocimiento por VB-ELM. El trabajo revisado incluye un tipo de biometría multimodal, en pocas palabras consiste en la combinación de dos o más elementos para identificar un individuo, por ende, la metodología considera como elementos biométricos la huella dactilar y el rostro en la extracción y modelado de entradas para el clasificador.

Se considera la base de datos biométrica multimodal, denominada FERET_FVC2002 , la cual contiene bases de datos de caras publicas FERET [50] y huellas dactilares de la base de datos FVC2002 [46]. La base de datos FERET_FVC2002 contiene 240 pares de imágenes, siendo 240 de rostros y 240 de huellas dactilares correspondiente una de la otra. Adicionalmente, con motivos de comparación se utiliza una actualización de la base de datos FERET_FVC2002, considerando la base de datos de huellas dactilares FVC2004, es decir, de FERET_FVC2002, pasa a ser FERET_FVC2004.

3.4.1 Preprocesamiento de imágenes

Se crea un preprocesamiento para las imágenes de huella dactilares y cara, luego se fusionan estas imágenes para construir las imágenes biométricas multimodales. En la cara se emplea un preprocesamiento de iluminación y normalización geométrica [19]. En la huella dactilar es utiliza una mejora de imagen, segmentación y el preprocesamiento de normalización, para luego extraer la ROI [51].

3.4.2 Extracción de características locales

Se extraen las características locales de ambos elementos biométricos, mediante los extractores de características, en los que se incluye el filtro de Gabor, momentos de Zernike y la función de patrón binario local (LBP). Las características poseen invariancia local y una fuerte descripción de la región, superando factores como efectos de luz. La fusión de características locales puede caracterizar mejor todos los detalles locales de las imágenes.

Se emplean algoritmos basados en bloques dinámicos que extraen las características locales. Para una imagen A se determina una ventana de bloque con el tamaño de $p \times q$, luego dicha ventana se mueve de izquierda a derecha, y de abajo hacia arriba. De esta forma se consiguen los L bloques de la imagen. Mediante los extractores de características filtro de Gabor (sección 2.1.4), momentos de Zernike (sección 2.1.3) y función de patrón binario local (2.1.4) se extraen las características locales de cada bloque de imagen.

3.4.3 Modelado de características visuales de fusión local

Se diseño un marco de fusión multimodal para las imágenes faciales y de huellas dactilares. Con las características visuales locales extraídas se construye una matriz de imágenes de características basada en bloques para obtener todas las características locales. Luego se extraen características de fusión compacta basada en la matriz de imágenes, con métodos 2DPCA o NMF, los cuales son una mejora del análisis de componentes principales (PCA), empleado también en el trabajo anterior [15].

Se denotan los vectores para características locales del i-ésimo bloque de imagen facial y huella dactilar, de la siguiente forma: $F_B^i = [F_G^i, F_H^i, F_L^i]^T$ y $Ft_B^i = [Ft_G^i, Ft_H^i, Ft_L^i]^T$ respectivamente, con $i = [1, \dots, K]$, donde K es el numero de bloque de la imagen de la cara y huella dactilar. La extracción de las características visuales de fusión local de la imagen biométrica multimodal se muestra a continuación:

- Construir una matriz X de características fusión local.

$$X = [F_B^1, F_B^2, \dots, F_B^K, Ft_B^1, Ft_B^2, \dots, Ft_B^K]_{Q \times 2K} \qquad (51)$$

  Esta matriz es construida con toda las características locales de los K bloques de imágenes de huellas dactilares y rostro.

- Descomponer la matriz de características. Empleando 2DPCA o NMF se puede conservar la información espacial. Si se consideran T muestras de imágenes X_i, $i = [1, \dots, T]$, se puede definir la matriz de covarianza de imagen G.

$$G = \sum_{i=1}^{T} [X_i - E(X)]^T [X_i - E(X)] \qquad (52)$$

  Se calculan los autovalores de la matriz G, y de esta forma obtener D vectores propios más grandes correspondientes a los valores propios, que se denotan como $[u_1, u_2, \dots, u_d]$, donde el parámetro D esta determinado por la regla de la contribución de los autovalores más grandes D a todos los autovalores es del 95%.
- Construir las características visuales de fusión local. Para una matriz de imagen de entrada X, con base a los autovalores $[u_1, u_2, \dots, u_d]$, se puede obtener la función de fusión local y_k.

$$y_k = Xu_k, \qquad k = 1,2, \dots, D. \qquad (53)$$

3.4.4 Reconocimiento por VB-ELM

Para las características visuales de fusión local extraídas de la imagen biométrica multimodal, se emplea la VB-ELM como clasificador para realizar la clasificación. Esta ELM fue descrita en la sección (2.2.6). A continuación, se presentará un algoritmo que sintetiza la metodología empleada en este trabajo.

Algoritmo 3: *Metodología para el reconocimiento biométrico multimodal basado en características visuales de fusión local y VB-ELM*

1: Preprocesamiento de imágenes faciales y huellas dactilares:
Para las imágenes del rostros se emplea iluminación, normalización de segmentación; Para las imágenes de huellas dactilares se mejora la imagen, segmentación, normalización y extracción de ROI.

2: Extracción de características locales:
Características locales de imágenes faciales y huellas dactilares, $F_B^i = [F_G^i, F_H^i, F_L^i]^T$ y $Ft_B^i = [Ft_G^i, Ft_H^i, Ft_L^i]^T$ respectivamente, con $i = [1, \dots, K]$. Al ser biometría multimodal esta presenta una fusión de características: $X = [F_B^1, F_B^2, \dots, F_B^K, Ft_B^1, Ft_B^2, \dots, Ft_B^K]_{Q\times 2K}$

3: Características visuales de fusión local:
Descomposición de la matriz con 2DPCA o NMF para obtener características visuales de fusión local, es decir:

$$y_k = Xu_k, \qquad k = 1,2, \dots, D.$$

4: Reconocimiento por VB-ELM:
Se ejecuta el algoritmo de la VB-ELM (sección 2.2.6), generar la distribución posterior del parámetro, es decir:

$$q(\beta) = \mathcal{N}(\beta|n, S)$$

4 Resultados

En esta sección se presentarán los resultados expuestos por los autores en sus respectivos trabajos, se realizará un análisis comparativo.

La mayoría de los autores definieron su métrica para evaluar en esta sección, por lo que, si es necesario, esta se definirá en caso de que no haya sido mencionada explícitamente en la metodología. En la mayoría de los casos, la métrica de evaluación de rendimiento no será compatible, ya que no todos los autores adaptan una forma universal, del mismo modo también hay que tener en cuenta que hay dos tipos de trabajos que se orienta a la identificación de huellas dactilares, mientras que los otros dos se orientan a la clasificación de huellas dactilares. Por ende, para realizar la comparación se tendrán en cuenta los factores comunes que se puedan observar durante los resultados.

4.1 Clasificación de huellas digitales a través de ELM estándar y desbalanceada [1]

Se consideraron tres tipos de ELM durante la experimentación, siendo estas la R-ELM, la W-ELM (teniendo en cuenta sus dos divisiones, W-ELM1 y E-ELM2 respectivamente) y la DW-ELM.

En este trabajo se adoptó la función de mapeo o activación sigmoide: $g(x) = 1/[1 - \exp(x)]$, debido a que garantiza la capacidad de aproximación universal de SLFN propensos a cualquier algoritmo ELM [21].

El ordenador que se utilizó para la obtención de los resultados fue una computadora simple con las siguientes características: un procesador Intel Core i5 a 2,6 GHz de velocidad de reloj y 4 GB de RAM. El entorno empleado fue MATLAB R2018a. Los resultados de comparación proporcionados por Peralta y otros [5] se obtuvieron utilizando una GPU Nvidia GeForce GTX TITAN (2688 núcleos, 6144 MB GDDR5 RAM).

4.1.1 Estimación de hiperparámetros óptimos de la ELM y evaluación de media G

Con el objetivo de maximizar la métrica de rendimiento para datos desequilibrados, es decir, la media G, se estiman los hiperparámetros de regularización C y el número de nodos N en la capa oculta en el conjunto de validación.

El conjunto se de datos fue dividido en tres subconjuntos: entrenamiento, validación y prueba. Habitualmente se suelen utilizar el subconjunto de entrenamiento y prueba, pero por cuestiones de protocolo, se adopta el subconjunto de validación.

El hiperparámetro C tuvo una variación entre 2^{-12} y 2^{12}, el número de N va aumentando gradualmente, esto con el objetivo de establecer observaciones generales y tuvieron en cuenta los extractores de características Capelli02, Hong08 y Liu10, en conjunto de los tres tipos de calidad de huellas dactilares (HQNoPert, Default y VQAndPert) generados por SFINGE [33, 34].

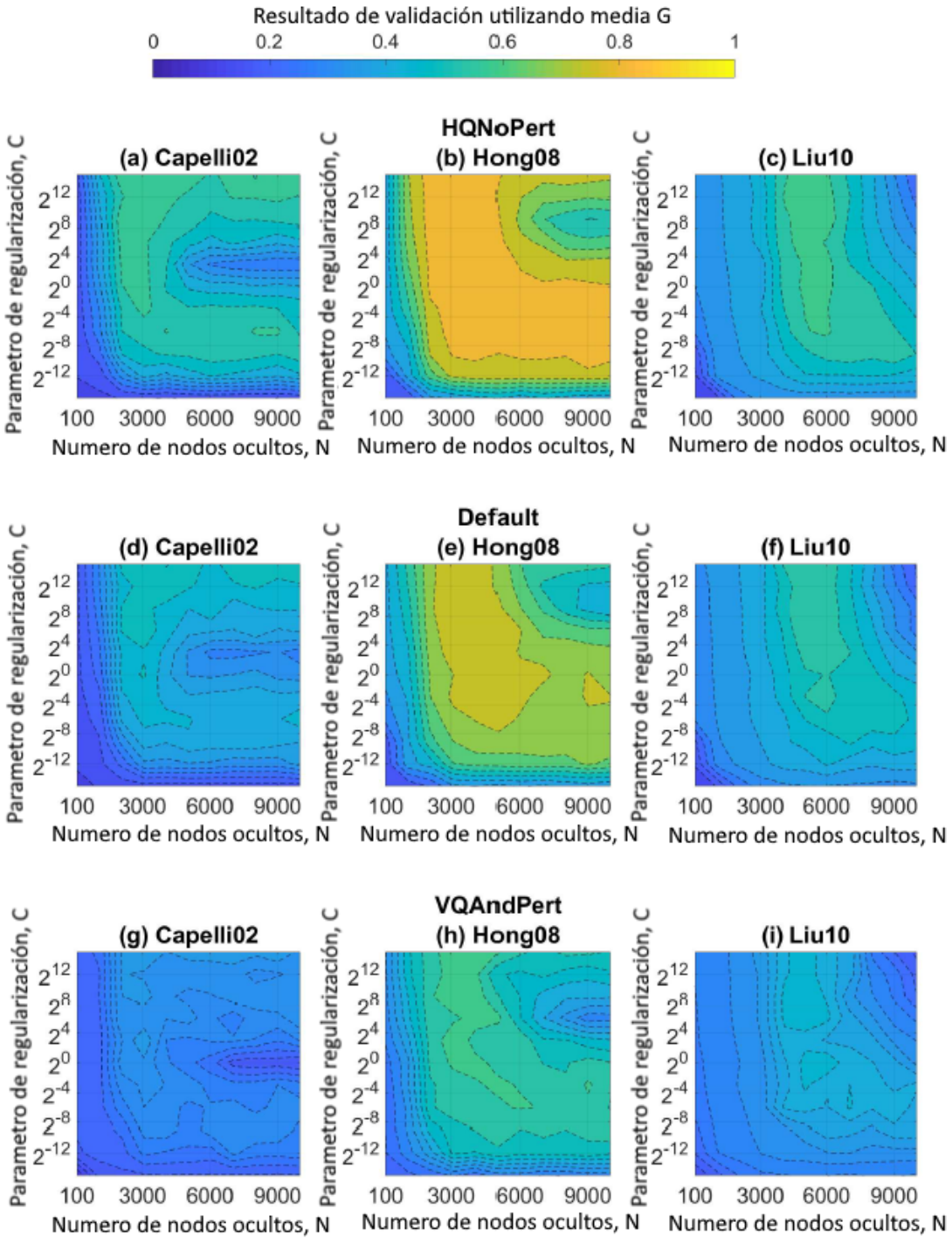


Figura 12: Resultado de validación obtenido por la R-ELM en función del hiperparámetro de regularización y número de nodos ocultos. Se consideraron los tres tipos de calidad de huellas y los tres extractores de características [1].

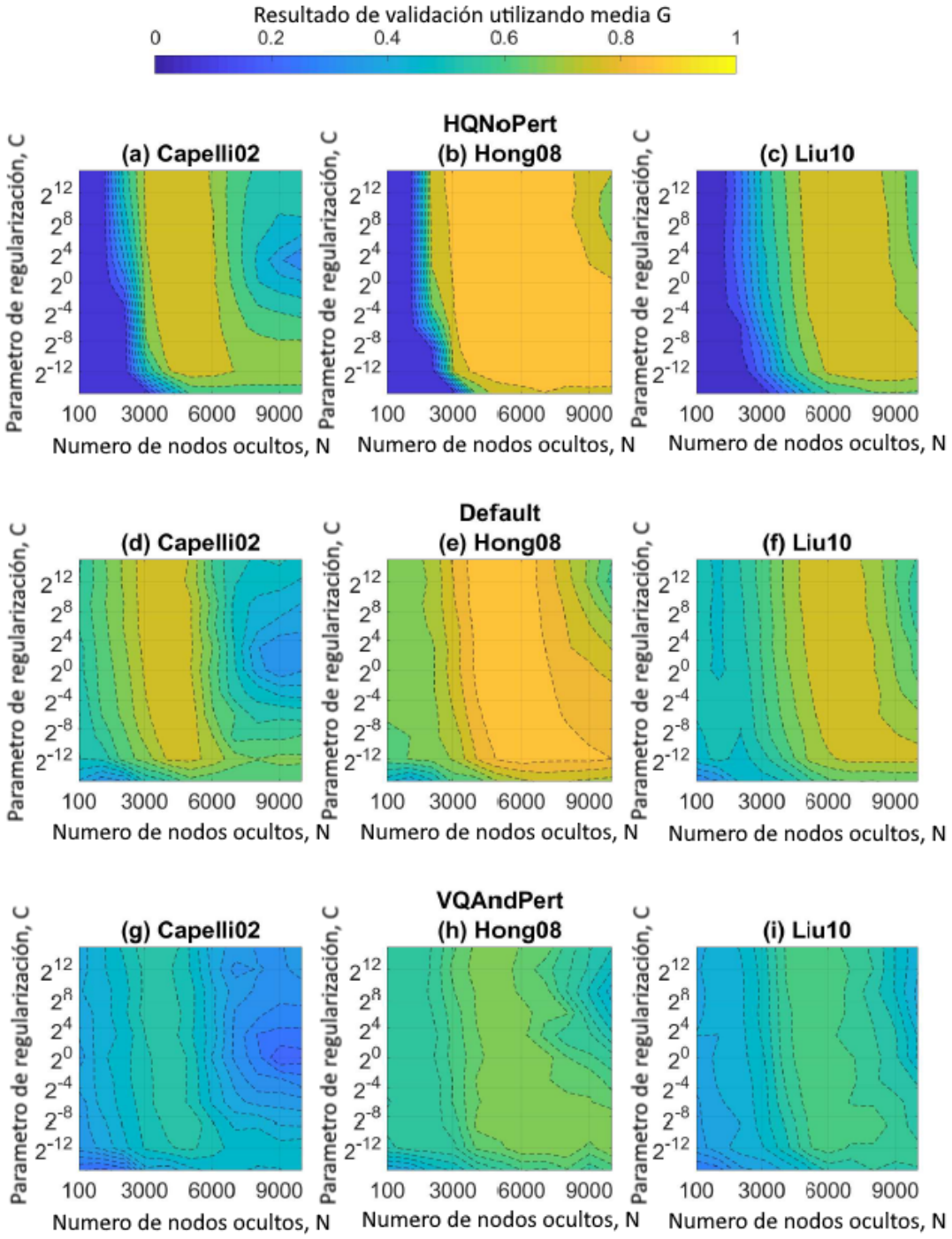


Figura 13: Resultado de validación obtenido por la W-ELM1 en función del hiperparámetro de regularización y número de nodos ocultos. Se consideraron los tres tipos de calidad de huellas y los tres extractores de características [1].

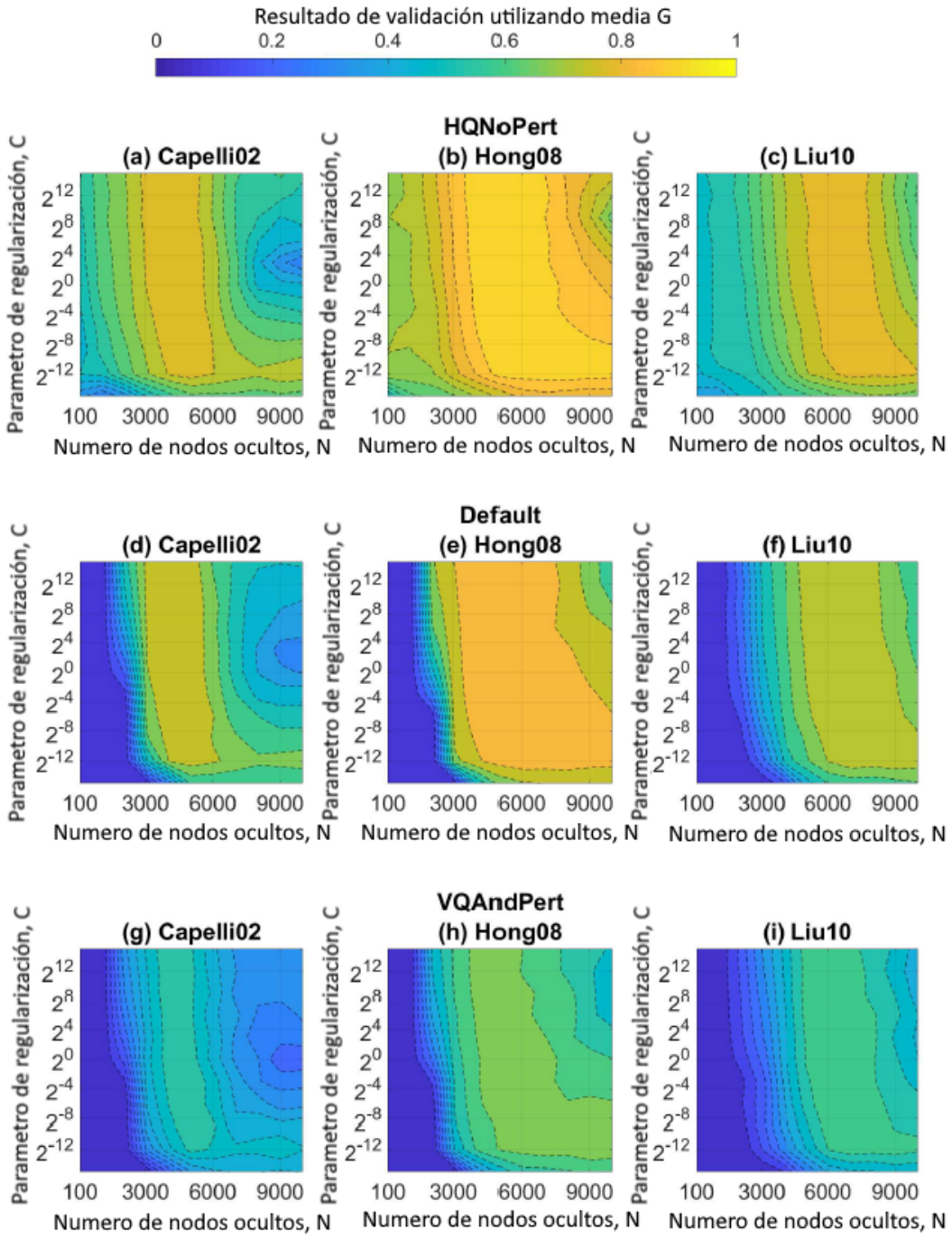


Figura 14: Resultado de validación obtenido por la W-ELM2 en función del hiperparámetro de regularización y número de nodos ocultos. Se consideraron los tres tipos de calidad de huellas y los tres extractores de características [1].

Como se puede apreciar en las figuras 12-14, la región para un mejor rendimiento de la ELM es bastante irregular, por estos motivos es necesario determinar una relación entre C y N para obtener una media G maximizada. Esto es posible de realizar por la característica de generación aleatoria de parámetros de neuronas de las ELM, en otros algoritmos se requieren procesos de iterativos y/o métodos informáticos de alto rendimiento.

De acuerdo con las gráficas de las figuras 12-14, se tabularon los valores óptimos de nodos ocultos N y el hiperparámetro de regularización C, los cuales corresponde a un intercepción de las áreas de mejor desempeño, siempre considerando un valor de N lo más bajo posible, para reducir costo computacional.

(a) R-ELM	**Capelli02**			**Hong08**			**Liu10**		
	N	C	**Media G**	N	C	**Media G**	N	C	**Media G**
HQNoPert			0.64			0.86			0.65
Default	3000	2^{10}	0.54	3000	2^{10}	0.80	5000	2^{8}	0.62
VQAndPert			0.31			0.58			0.40
(b) W-ELM1	**Capelli02**			**Hong08**			**Liu10**		
	N	C	**Media G**	N	C	**Media G**	N	C	**Media G**
HQNoPert			0.64			0.92			0.67
Default	4000	2^{6}	0.54	5000	2^{4}	0.88	5000	2^{15}	0.63
VQAndPert			0.37			0.65			0.49
(c) W-ELM2	**Capelli02**			**Hong08**			**Liu10**		
	N	C	**Media G**	N	C	**Media G**	N	C	**Media G**
HQNoPert			0.66			0.93			0.69
Default	4000	2^{6}	0.57	5000	2^{4}	0.89	5000	2^{15}	0.64
VQAndPert			0.40			0.67			0.51

Figura 15: Tabla de resultados de media G obtenidos por R-ELM, W-ELM1 y W-ELM2 con los hiperparámetros C y N optimizados. Se consideraron los extractores de características Capelli02, Hong08 y Liu10, y los tres tipos de calidad de huellas dactilares [1].

Como se puede apreciar en la figura 15, para todos los tipos de ELM, el mejor rendimiento se logra con la calidad de huella HQNoPert, la cual presenta la mejor calidad de todas, por otra parte, el mejor extractor de características que se puede notar es Hong08. De acuerdo con la media G en la tabla de la figura 15, la mejor combinación es emplear la W-ELM2, considerando la calidad de huellas dactilares HQNoPert y el extractor de características Hong08. Mientras que el peor resultado es para la R-ELM. Esto se explica por el tipo de datos, ya que no siguen una distribución uniforme, es decir, son datos desequilibrados, por ende, la R-ELM queda propensa a dicho problema, mientras que la W-ELM fue propuesta como una solución para los datos desequilibrados.

Por otra parte, para evaluar media G de la DW-ELM es necesario obtener el hiperparámetro d, además de C y N. Para esto se adoptaron los hiperparámetros óptimos de C y N correspondientes a la W-ELM1, debido a que DW-ELM es una extensión de este tipo de ELM.

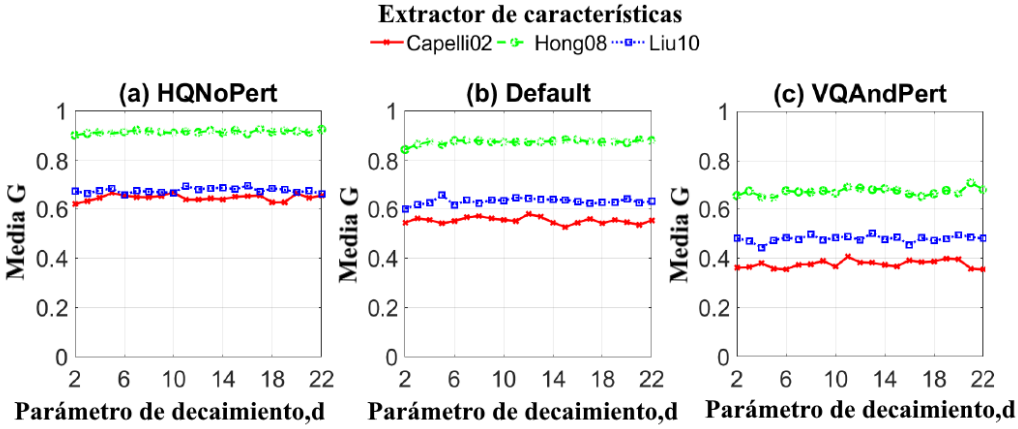


Figura 16: Gráficos de media G obtenidos frente al parámetro de decaimiento d de la DW-ELM para cada extractor de características y calidad de huellas dactilares. C y N se adoptaron de la W-ELM1 [1].

DW-ELM	Capelli02		Hong08		Liu10	
	d	Media G	d	Media G	d	Media G
HQNoPert	9	0.67	16	0.93	15	0.69
Default	11	0.58	15	0.89	4	0.65
VQAndPert	10	0.40	20	0.71	12	0.50

Figura 17: Resultado de la media G para con el hiperparámetro d optimizado de acuerdo con los gráficos de la figura 16 [1].

Como se puede apreciar en la tabla de la figura 17, los resultados de la media G de la DW-ELM oscilan entre la W-ELM1 y la W-ELM2, por lo tanto, no es recomendable utilizarla, porque involucra un costo computacional más elevado en comparación de las W-ELM. Por ende, la W-ELM2 con HQNoPert y Hong08 se sigue manteniendo como la mejor opción en términos computacionales y media G.

4.1.2 Evaluación y comparación mediante precisión y tasa de penetración

Este tipo de métricas tradicionales tienen como propósito ser de utilidad para realizar comparaciones con otros algoritmos de clasificación de huellas dactilares, como las CNN, que son la competencia directa que se tiene en la actualidad. La precisión (Acc) es muy común que se utilice como métricas en los problemas de clasificación y regresión, mientras que la tasa de penetración (PR) se adoptó recientemente en el contexto de clasificación de huellas dactilares [38, 5].

(a) Capelli 02	R-ELM		W-ELM1		W-ELM2		DW-ELM	
	Acc	PR	Acc	PR	Acc	PR	Acc	PR
HQNoPert	0.80	0.1788	0.79	0.1650	0.81	0.1500	0.79	0.1645
Default	0.79	0.2112	0.74	0.1969	0.76	0.1785	0.64	0.1942
VQAndPert	0.61	0.2913	0.60	0.2522	0.63	0.2349	0.61	0.2521
(b) Hong08	**R-ELM**		**W-ELM1**		**W-ELM2**		**DW-ELM**	
	Acc	**PR**	**Acc**	**PR**	**Acc**	**PR**	**Acc**	**PR**
HQNoPert	0.95	0.0485	0.94	0.0340	0.95	0.0332	0.95	0.0330
Default	0.94	0.0662	0.93	0.0412	0.94	0.0406	0.94	0.0412
VQAndPert	0.86	0.0954	0.88	0.0519	0.88	0.0512	0.88	0.0521
(c) Liu10	**R-ELM**		**W-ELM1**		**W-ELM2**		**DW-ELM**	
	Acc	**PR**	**Acc**	**PR**	**Acc**	**PR**	**Acc**	**PR**
HQNoPert	0.78	0.2060	0.79	0.1727	0.80	0.1651	0.79	0.1711
Default	0.79	0.2220	0.77	0.1866	0.77	0.1751	0.78	0.1787
VQAndPert	0.66	0.2696	0.67	0.2327	0.68	0.2166	0.68	0.2315

Figura 18: Precisión y tasa de penetración de error absoluto para cada calidad imagen de huella dactilar y cada tipo de ELM considerada en el trabajo con sus respectivos hiperparámetros optimizados [1].

En la figura 18 se puede apreciar que nuevamente la combinación de la W-ELM2 con el extractor de características Hong08 y la calidad HQNoPert producen los rendimientos superiores, especialmente para la tasa de penetración, mientras que la R-ELM se mantiene como la peor en cuanto a rendimiento. Por lo tanto, se puede afirmar que la W-ELM2 es capaz de mejorar la tasa de reconocimiento de la clase minoritaria de huella dactilar, de esta forma maximizar la media G y la PR, y garantizar una clasificación adecuada para la clase mayoritaria, manteniendo una precisión superior.

Para tener una visión más clara de la capacidad, la combinación más optima de la W-ELM2 se comparó con un trabajo de un método novedoso basado CNN [5] y una modificación de CaffeNet CNN [52] para el problema de clasificación de huellas dactilares. Cada CNN procesaron imágenes de huellas dactilares sin un extractor de características. Y el rendimiento se evaluó en términos de precisión y PR considerando solo las bases de datos, a diferencia del trabajo revisado que considero la validación cruzada quíntuple.

	Hong08 y W-ELM2		CaffeNet CNN modificado		Nueva CNN	
	Acc	PR	Acc	PR	Acc	PR
HQNoPert	0.95	0.0332	0.99	0.0051	0.99	0.0031
Default	0.93	0.0406	0.97	0.0211	0.98	0.0153
VQAndPert	0.88	0.0512	0.96	0.0329	0.96	0.0279

Figura 19: Comparación de los resultados obtenidos por la mejor combinación de las ELM estudiadas en el trabajo, el CaffeNet CNN y el método CNN propuesto por Peralta y otros [1].

Como se puede apreciar en la figura 19, los métodos por CNN son ligeramente superiores a la ELM propuesta en el trabajo, donde se aprecia una mejor competitividad en términos de precisión de clasificación.

Cabe mencionar que el grado de complejidad del método propuesto en el trabajo revisado comparado con las CNN contrasta demasiado. Ya que implementar una CNN requiere de tecnología computacional de última generación, en comparación del algoritmo basado en ELM que puede ejecutarse mediante una computadora personal, y aun teniendo los recursos para ejecutar los clasificadores basados en CNN, el tiempo aprendizaje es mucho mayor que el de los clasificadores basados en ELM, tal como se puede apreciar en la figura 20.

	Hong08 y W-ELM2	CaffeNet CNN modificado	Nueva CNN
HQNoPert	880	2306	960
Default	885	2329	957
VQAndPert	882	2328	960

Figura 20: Comparación en términos del tiempo de aprendizaje para la mejor combinación de las ELM estudiadas en el trabajo, el CaffeNet CNN y el método CNN propuesto por Peralta y otros [1].

4.2 Clasificación de huellas dactilares naturales usando el histograma de gradiente como descriptor [13]

En este trabajo como ya se ha mencionado en reiteradas ocasiones, centra su contenido en la modificación del extractor de características HOG. No se especifican detalles técnicos o teóricos de la ELM utilizada. El ordenador con el que se obtuvieron los resultados no fue especificado en el documento.

Para evaluar los resultados se utilizaron huellas dactilares de la base de datos FVC2004, las cuales son huellas dactilares ruidosas, considerando cuatro subbases de datos (DB1, DB2, DB3 y DB4) con 800 huellas dactilares de 100 personas con 8 muestras cada una [45]. La distribución de datos se pasó por alto, y además la base de datos mezcla las categorías de arco y arco tienda, lo que provoca un aumento en el costo computacional y aumenta la tasa de penetración.

En primer lugar, se extrajeron las características con el HOG considerando huellas dactilares mejoradas y sin mejora. Una vez calculadas las características para tipo de imagen de huella dactilar, se aplica directamente la ELM sin especificaciones por parte del autor. Los resultados se pueden apreciar en la figura 21.

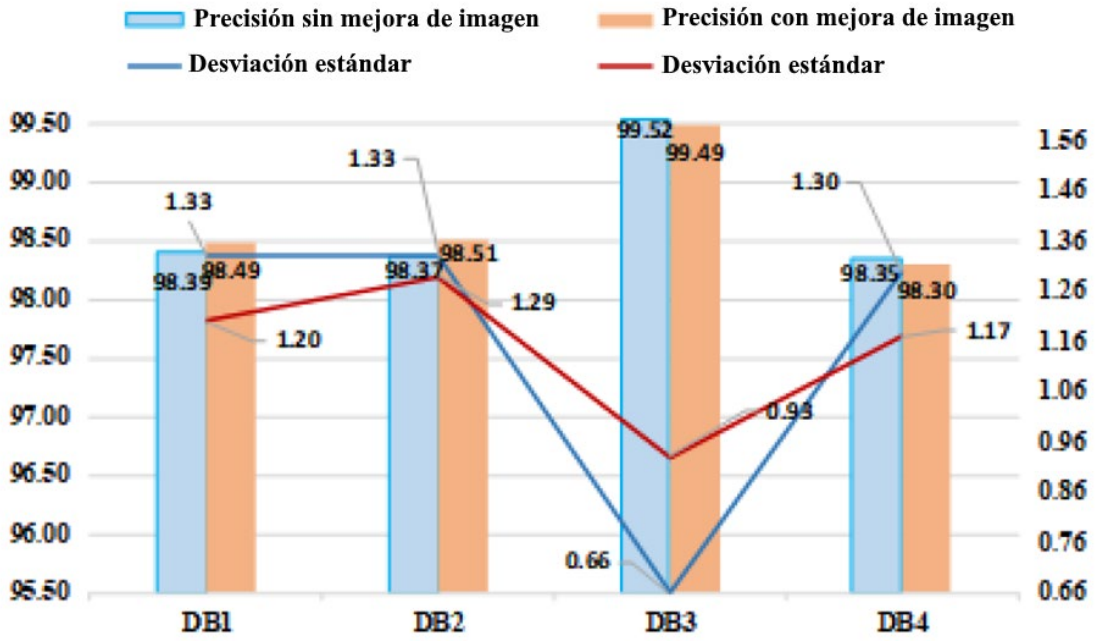


Figura 21: Precisión de clasificación y desviación estándar para cada subbase de datos de la base de datos de huellas dactilares FVC2004, considerando imágenes con y sin mejora [13].

Como se puede apreciar en la figura 21, no existe una diferencia significativa entre la imagen de huella dactilar con mejora y sin mejora. La máxima precisión se alcanza en la subbase de datos DB3, sin mejora con un valor de 99.52, mientras que con mejora alcanza un valor de 99.49. Esto quiere decir que el HOG modificado por el autor es resistente a la extracción de características de imágenes de huellas dactilares con ruido y baja calidad. El promedio de la base de datos es de 98.66 sin mejora, y 98.70 con mejora, siendo levemente superior la precisión con mejora de imagen de huella dactilar.

Tabla 1: Comparación de métodos de clasificación de huellas dactilares usando FVC2004 [13].

Artículo	Características	Precisión
Guo y otros [53]	Punto Central, Punto Delta, CDF, BAF	92.70%
Jung y Lee [54]	Bloque de núcleo, Información direccional	97.40%
Dorasamy y otros [55]	Patrones direccionales y puntos singulares.	92.20%
Método propuesto con mejoras	HOG Modificado	98.70%
Método propuesto sin mejoras	HOG Modificado	96.66%

Como se puede apreciar en la tabla 1, el método propuesto con o sin mejora es superior a los métodos más avanzados de comparación, por lo que el método propuesto en el trabajo es capaz de capturar mejor las características de una imagen de huellas dactilares ruidosas.

No se incluyen otro tipo de métricas de rendimiento, como tiempo de entrenamiento o PR, por lo que, al comparar este trabajo con otros tipos de ELM solo se considerara la precisión.

4.3 Identificación de huellas digitales basado en ELM [15]

El ordenador empleado para la obtención de los resultados fue una Pentium 2 GHz con el software MATLAB. Se considero la base de datos FVC2002 [46], con cuatros subbases de datos especificadas en la metodología (sección 3.3). Debido a que las bases de datos pueden incluir casos en que los puntos de referencia de las huellas dactilares este en un lugar no apropiado, puede conducir a un error en la adquisición de las ROI, por lo tanto, se utiliza un

vector para las impresiones actuales, el cual es el promedio de dos vectores de características que se obtiene de la impresión previa y siguiente. Esto es más que nada para descartar las impresiones rechazas de huellas dactilares.

Como cada subbase contiene 800 huellas dactilares siendo de 100 personas con 8 muestras cada una, el proceso de entrenamiento de la ELM toma 6 de las muestras de las 8 muestras, mientras que el proceso de prueba de la ELM toma directamente las 8 muestras. Estos datos se normalizaron escalando los valores máximos de cada columna en $(0,1]$ para evitar la no convergencia. El vector de características con $17 \times 3 = 51$ elementos, los cuales consisten en tres conjuntos de momentos derivados de tres conjuntos de diferentes ROI, el cual fue refinado por PCA para la selección de características.

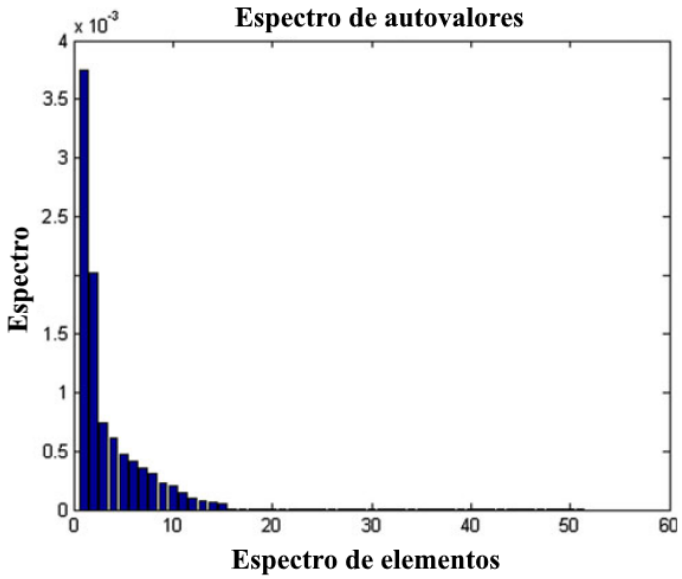


Figura 22: Espectro de autovalores con diferentes elementos obtenidos utilizando PCA para la selección de características [15].

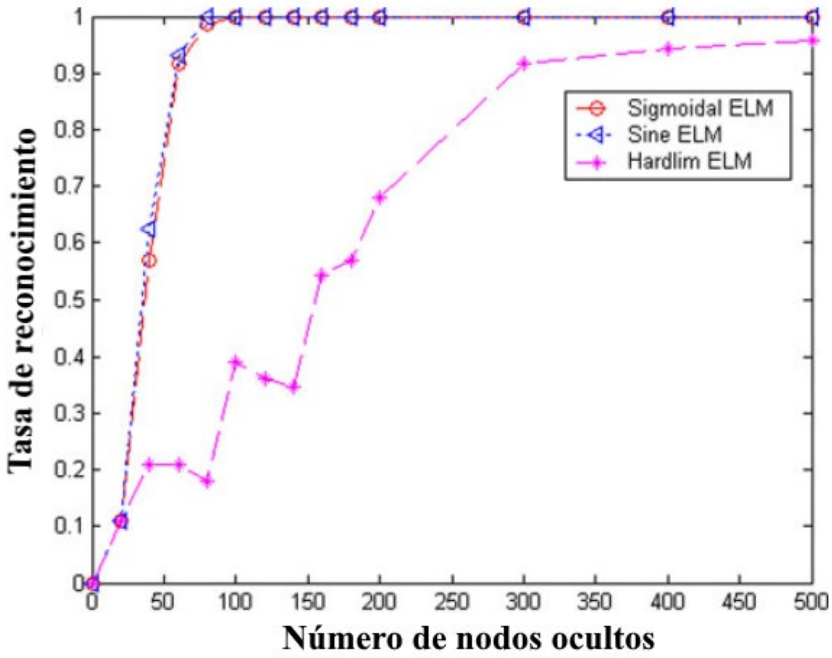


Figura 23: Tasa de reconocimiento con respecto al número de nodos ocultos y diferentes tipos de funciones de mapeo para las ELM utilizando características de 15 dimensiones para la identificación de huellas dactilares [15].

En la figura 22 se puede apreciar que todo el espectro de autovalores con diferentes elementos obtenidos mediante PCA para la selección de características, es evidente que este se concentra por debajo de 20 autovalores, por lo tanto, se escogieron 15 autovectores (k=15) para mantener una variación de del 98%. Por lo tanto, las nuevas 15 características de 15 dimensiones no están correlacionadas. Con las características seleccionadas para los datos de prueba y entrenamiento, se proceden a enviar a la ELM o R-ELM para ver si coinciden o no.

En la figura 23 se puede observar que la tasa de reconocimiento aumenta a medida que el número de nodos ocultos aumenta, la función de mapeo sigmoide y seno tiene un resultado similar, mientras que la Hardlim está muy por debajo de estas dos últimas. De acuerdo con estos resultados, la función de mapeo que se adopta es la sigmoidal debido a que garantiza la capacidad de aproximación universal de SLFN propensos a cualquier algoritmo ELM. La alta tasa de reconocimiento se explica por número alto de nodos ocultos.

4.3.1 Evaluación de rendimiento

Se selecciono el protocolo de desempeño que se utiliza en la base de datos FVC2002 [46]. Esta corresponde a la tasa de aceptación (FAR) y la tasa de rechazo (FRR), y para calcularlas se realizó una coincidencia genuina y una coincidencia impostora. Para la coincidencia genuina, cada impresión de prueba de cada persona se comparó con la plantilla de la misma persona.

$$FRR = \frac{Número\ de\ reclamaciones\ geniunas\ rechazadas}{Numero\ total\ de\ accesos\ geniunos} \times 100\% \qquad (54)$$

$$FAR = \frac{Número\ de\ reclamaciones\ impostoras\ aceptadas}{Numero\ total\ de\ accesos\ impostores} \times 100\% \qquad (55)$$

El método para evaluar la tasa de reconocimiento del método de ELM de identificación de huellas dactilares, se empleó la características operativa del receptor (ROC), el cual es un gráfico de la tasa de aceptación genuina ($GAR = 1 - FRR$) frente al FAR. Para medir el rendimiento se utilizó la **tasa de error equivalente (ERR)**, la cual es un punto medio donde FRR y FAR son iguales. Para calcular el FAR y FRR de la identificación genuina y la identificación impostora se consideraron las 4 bases de datos secundarias de FVC2002. Para una coincidencia genuina, cada patrón de huellas dactilares de cada persona se comparó con otros patrones de huellas dactilares de la misma persona. Para la coincidencia impostora, cada patrón de huellas dactilares de prueba se comparó con uno de los patrones de huellas dactilares pertenecientes a otras personas. Dado que hubo 600 patrones de entrenamiento y 800 patrones de prueba, el número total de coincidencias genuinas e impostoras fue de $800 \times 6 = 4800$ y $800 \times \frac{99}{2} = 39600$ para cada subbase de datos.

4.3.2 Comparación de rendimiento

Para comparar el rendimiento de los enfoques de identificación propuestos basados en R-ELM y ELM se realizó un análisis estadístico de los resultados experimentales. Es decir, se replicaron los experimentos con los mismos datos de entrenamiento y prueba en ambos tipo de ELM.

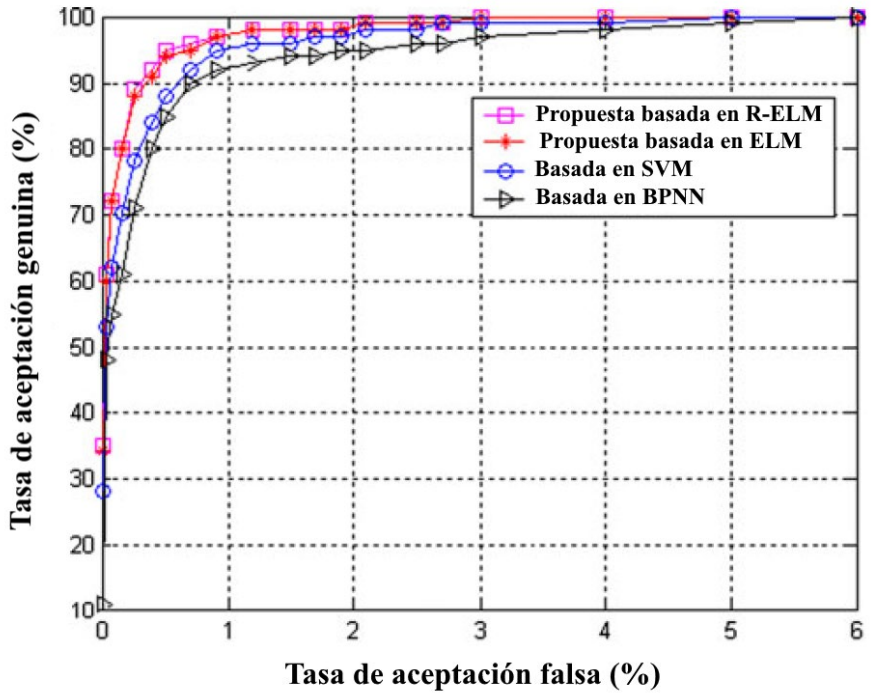


Figura 24: Curvas ROC que indican la tasa de reconocimiento del método propuesto con ELM y R-ELM, comparado con un método de SVM [56] y un método basado en BPNN [57]. Se considera la base de datos FVC2002 DB1_a [15].

Como se puede apreciar en la figura 24, la curva roja y azul correspondiente a los métodos propuestos en el trabajo, son más altas que los métodos basados en SVM [56] y BPNN [57].. Esto quiere decir que los métodos propuestos basados en ELM tiene mejor precisión en la base de datos FVC2002 DB1_a. La diferencia no logra a divisarse porque es mínima, pero entre la R-ELM y la ELM, la tasa de aceptación genuina más alta corresponde a la R-ELM propuesta, manteniendo la misma tasa de aceptación falsa y con un rendimiento superior a los demás métodos.

	DB1_a	DB2_a	DB3_a	DB4_a	Promedio
BPNN	2.17	3.13	4.25	3.63	3.29
SVM	1.32	2.18	2.38	2.21	2.02
Propuesta ELM	1.21	2.13	2.24	2.12	1.93
Propuesta R-ELM	1.11	2.08	2.16	2.09	1.86

Figura 25: Comparación del EER (%) de los métodos propuestos basados en ELM y R-ELM, y los métodos basados en BPNN y SVM. Se consideran las cuatro subbase de datos de FVC2002 [15].

En la figura 25 se puede apreciar que las subbases de datos presentan diferentes ERR, esto es según la calidad de las muestras y los métodos aplicados. El valor promedio de ERR para la ELM es de 1.93, mientras que el valor promedio ERR de la R-ELM es de 1.86, esto demuestra que el error es menor en comparación a los otros métodos de comparación (BPNN y SVM).

	Promedio de tiempo de entrenamiento (s)	Promedio de tiempo de prueba (s)	Promedio total de tiempo (s)
BPNN	40.94	0.82	41.76
SVM	0.45	0.15	0.60
Propuesta ELM	0.25	0.13	0.38
Propuesta R-ELM	0.27	0.13	0.40

Figura 26: Comparación de los promedio de tiempo de entrenamiento, prueba y total de los métodos propuestos basados en ELM y R-ELM, y los métodos basados en BPNN y SVM. Se consideran las cuatro subbase de datos de FVC2002 [15].

En la figura 26 se demuestra que el promedio de entrenamiento, prueba y total para los clasificadores basados en ELM es menor al tiempo de los otros métodos de comparación (BPNN y SVM), estos resultados de simulación demuestran que el método que se propone en el trabajo es superior a los métodos que se consideraron para la comparación.

4.4 Reconocimiento biométrico multimodal basado en características visuales de fusión local y máquina de aprendizaje extremo bayesiano variacional [18]

No se especifica el ordenador empleado para la obtención de resultados en este trabajo. La base de datos considerada para los experimentos resulta ser una fusión de dos bases de datos, esta se denomina FERET_FVC2002, la cual surge de la unión de la base de datos de rostros FERET [50] y la base de datos de huellas dactilares FVC2002 [46]. La base de datos FERET_FV2002 contiene 240 pares de imágenes, siendo 240 de rostros y 240 de huellas dactilares correspondiente una de la otra. Se seleccionaron al azar 40 personas con 6 imágenes para cada persona para cada base de datos.

Para cada imagen se realiza el preprocesamiento (sección 3.4.1), la extracción de características (sección 3.4.2) y la fusión local de características visuales (sección 3.4.3). De esta forma imagen biométrica multimodal se adapta para ser una entrada para la VB-ELM.

4.4.1 Comparaciones de rendimiento

Pese que el trabajo se enfoca en el reconocimiento biométrico multimodal, se realizaron comparaciones con datos biométricos individuales que permiten comparar los rendimientos de ambos sistemas de reconocimiento. Se consideraron algoritmos SVM [58], ELM y VB-ELM con los diferentes extractores de características (filtro de Gabor, momentos de Zernike y LBP) y los dos modelos de características visuales de fusión local (2DPCA y NMF) descritos en la sección 3.4.4.

Se selecciono N imágenes ($N=2,3,4,5$) para cada persona como muestras de entrenamiento, y en la izquierda $6-N$ imágenes ($4,3,2,1$) para cada persona como muestras de prueba.

Algoritmos	Número de muestras de entrenamiento para cada persona (N)				
	(N)	N = 2	N = 3	N = 4	N = 5
SVM	Filtro de Gabor	0.8675	0.8883	0.9125	0.9355
	Momentos de Zernike	0.8313	0.8417	0.8825	0.9045
	LBP	0.8575	0.8655	0.9068	0.9175
ELM	Filtro de Gabor	0.8854	0.9006	0.9164	0.9317
	Momentos de Zernike	0.8405	0.8606	0.9044	0.9140
	LBP	0.8619	0.8823	0.9086	0.9267
VBELM	Filtro de Gabor	**0.9037**	**0.915**	**0.9225**	**0.9475**
	Momentos de Zernike	**0.8689**	**0.8736**	**0.9105**	**0.9168**
	LBP	**0.8941**	**0.9057**	**0.9175**	**0.9345**

Figura 27: Comparaciones de rendimiento de SVM, ELM y VB-ELM con características biométricas únicas considerando la base de datos de rostros FERET [18].

En la figura 27 se puede apreciar que algoritmo VB-ELM tiene la mejor tasa de reconocimiento más alta en los tres tipos de extractores de características en comparación de los algoritmos basados en SVM y ELM. En cuanto a peor rendimiento, se puede apreciar que el algoritmo de la ELM básica y SVM varían dependiendo las muestras de entrenamiento. El filtro de Gabor demuestras ser superior en comparación de los otros dos extractores de características.

Algoritmos	Número de muestras de entrenamiento para cada persona (N)				
	Tiempo(s)N=2	Tiempo(s) N=3	Tiempo(s) N=4	Tiempo(s) N=5	Tiempo(s) Promedio
SVM	41.58	49.07	56.74	65.44	53.21
ELM	0.13	0.13	0.14	0.16	0.14
VBELM	7.12	7.78	7.97	8.23	7.77

Figura 28: Comparación de tiempo-consumo entre tres clasificadores en segundos [18].

Los tiempo de entrenamiento se pueden ver en la figura 28, se puede notar que el algoritmo SVM consume una gran cantidad de tiempo en comparación de los algoritmos basados en ELM. El algoritmo que consume menor tiempo es la ELM básica, en comparación con su mejora VB-ELM es más sencillo a nivel computacional, por ende, se explica que toma menos tiempo el entrenamiento.

Algoritmos	Número de muestras de entrenamiento para cada persona (N)				
	(N)	N = 2	N = 3	N = 4	N = 5
SVM	Filtro de Gabor	0.875	0.9185	0.9285	0.9400
	Momentos de Zernike	0.8615	0.9050	0.9175	0.9356
	LBP	0.8875	0.8967	0.9256	0.9385
	LFVF-2DPCA	0.8825	0.9233	0.9525	0.9655
	LFVF-NMF	0.8975	0.9367	0.9575	0.9650
ELM	Filtro de Gabor	0.9088	0.9195	0.9228	0.9475
	Momentos de Zernike	0.8982	0.9098	0.9128	0.9354
	LBP	0.8922	0.9088	0.9295	0.9445
	LFVF-2DPCA	0.9157	0.9216	0.9427	0.9595
	LFVF-NMF	0.9141	0.9313	0.9565	0.9682
VBELM	Filtro de Gabor	0.8929	0.9213	0.9443	0.9495
	Momentos de Zernike	0.8977	0.9167	0.9355	0.9465
	LBP	0.8875	0.9117	0.9275	0.9448
	LFVF-2DPCA	0.9187	0.9317	0.9568	0.9625
	LFVF-NMF	0.9275	0.9417	0.9775	0.9875

Figura 29: Comparaciones de rendimiento SVM, ELM y VB-ELM con diferentes características, monomodo y multimodo en la base de datos FERET_FVC2002 [18].

Como se puede apreciar en la figura 29, a medida que aumenta el número de muestras N, la tasa de reconocimiento aumenta para todos los algoritmos y extractores de características, incluyendo a los fusionadores de características (2DPCA y NMF). Sin embargo, la VB-ELM demuestra superioridad, y la tasa de reconocimiento máxima de 0.9875 que alcanza, es cuando utiliza la biometría multimodal con el fusionador de características NMF con N=5 muestras.

Se considera una base de datos actualizada llamada FERET_FVC2004, la cual toma la base de datos de huellas dactilares FVC2004 [45] que es más grande que su antecesora FVC2002 [46], de esta forma se puede crear una base de datos multimodal más grande al combinarla con FERET [50]. Esta base de datos utiliza 480 pares de imágenes, siendo 480 imágenes de rostros y 480 imágenes de huellas dactilares, perteneciente a 80 personas con 6 muestras cada una.

Con fines de comparación, se replicó la metodología empleada en la base de datos FERET_FVC2002 en la base de datos FERET_FVC2004, considerando el preprocesamiento de imagen, extracción de características, fusión de características y aplicación de los algoritmos basados en ELM (ELM básica y VB-ELM).

Algoritmos	Número de muestras de entrenamiento para cada persona (N)				
	(N)	N = 2	N = 3	N = 4	N = 5
ELM	Filtro de Gabor	0.8411	**0.9089**	**0.9394**	**0.9747**
	Momentos de Zernike	0.8321	0.8976	0.9120	0.9352
	LBP	0.8334	0.9056	0.9334	0.9498
	LFVF-2DPCA	0.8627	0.9209	0.9438	0.9850
VBELM	Filtro de Gabor	**0.8531**	0.9067	0.9250	0.9625
	Momentos de Zernike	**0.8577**	**0.9085**	**0.9178**	**0.9565**
	LBP	**0.8625**	**0.9100**	**0.9437**	**0.9750**
	LFVF-2DPCA	**0.8875**	**0.9217**	**0.9750**	**0.9875**

Figura 30: Comparación de rendimiento de ELM y VB-ELM con diferentes características biométricas monomodo y multimodo en la base de datos FERET_FBV2004 [18].

En la figura 30, se puede apreciar que, aunque la base de datos multimodal aumente de tamaño, la VB-ELM se mantiene como mejor clasificador en tasa de reconocimiento considerando las características fusionadas con 2DPCA, mientras que ELM básica tiene un incremento en la tasa de reconocimiento a medida que las muestras aumentan, especialmente en las características biométricas monomodal extraídas por el filtro de Gabor. Sin embargo, se demostrado con las figuras 27-30 que la VB-ELM representa una mejora para el reconocimiento biométrico, ya sea monomodal y multimodal.

5 Conclusiones

5.1 Análisis

Los documentos revisados emplearon en sus respectivas propuestas diferentes tipos de ELM para afrontar el problema de clasificación y/o identificación de huellas dactilares. La identificación de huellas dactilares puede ser más precisa y rápida al disminuir la tasa de penetración de las bases de datos de huellas dactilares, que es hacía donde apuntan los trabajos que buscan clasificar de huellas dactilares. Por lo que existe una relación directa entre ambos tipos ambos tipos de problemas.

En los resultados de cada trabajo revisado, de forma general se puede apreciar que las ELM utilizadas obtuvieron un mejor rendimiento, ya sea en clasificación o identificación de huellas dactilares, en comparación de los algoritmos más tradicionales como SVM y BPNN. Por otra parte, los algoritmos basados en CNN siguen siendo ligeramente superiores a la combinación de la W-ELM2 con el extractor de características Hong08 (mejor combinación en el trabajo) en términos de precisión y tasa de penetración, aunque hay que considerar que los algoritmos basados en CNN no son accesibles, y tienen costo computacional muy elevado al igual que el tiempo de entrenamiento.

A pesar de esto, la combinación de W-ELM2 junto al extractor Hong08 demuestra ser una excelente alternativa para ejecutar clasificación de huellas dactilares, ya que su algoritmo es accesible, perfectamente se puede ejecutar con un computador personal, y sus resultados son muy buenos, en términos de media G, precisión y reducción de la tasa de penetración de la base de datos.

La precisión y rendimiento no se pueden evaluar de forma justa, pues, en los trabajos [15, 18, 13] se utilizaron base de datos FVC2002 y FVC2O04, las cuales son bases de datos ideales, no siguen una distribución natural de las huellas dactilares, facilitando mucho el trabajo del clasificador, ya que no se consideran los problemas que surgen cuando la distribución es desbalanceada, como es naturalmente en las huellas dactilares, esta distribución natural solo se consideró en el trabajo [1].

La máxima precisión en clasificación de huellas dactilares se logra con el trabajo [13], pero este carece de mucha información técnica y teórica acerca del clasificador empleado, además de que utiliza una base de datos ideal que combina dos clases de huellas dactilares (arco y arco tienda).

La mejora tasa de reconocimiento (identificación) se obtuvo en el trabajo [18], siendo ligeramente superior a la tasa de reconocimiento del trabajo [15], esto puede explicarse por el tipo de ELM utilizado en el trabajo [18], el cual consiste en la VB-ELM, además de que utiliza características visuales multimodales fusionadas (rostro y huella dactilar), mientras que en el trabajo [15] se utiliza la R-ELM y las características son monomodales, seleccionadas por PCA simple, mientras que el anterior fusiona y selecciona las características visuales con PCA mejorado. Sin embargo, la diferencia es muy leve, siendo la tasa de reconocimiento de 0.9875 para el trabajo [18], y de 0.9814 para el trabajo [15], esto cobra bastante sentido, porque ambas tipos de ELM fueron propuestas para un problema de sobreajuste de las ELM básica.

Con base al análisis de resultados, se puede concluir que los algoritmos basados en ELM son una alternativa económica y accesible para reducir la tasa de penetración de la base de datos en comparación de algoritmos tradicionales. Si bien existen algoritmos más potentes como CNN, no son una alternativa al alcance de cualquier individuo, además de los problemas que se han reiterado en varias ocasiones sobre este tipo de clasificador. Las ELM demostraron ser veloces en los tiempos de entrenamiento, superando a toda la competencia que se le comparo en sus respectivos trabajos, incluyendo las CNN. Los resultados en rendimiento en la mayoría de los casos fueron bastante buenos, superando a la competencia sin considerar los algoritmos basados en CNN.

Los resultados más cercanos a la realidad fueron del trabajo [1] donde se consideró un distribución real de huellas dactilares y un gran número de muestras, en comparación de los otros trabajos.

5.2 Trabajos Futuros

Se ha demostrado que las ELM son competentes a la hora de enfrentar problemas de clasificación o identificación, no obstante, se pudo observar que muchos de los experimentos se realizaron con entornos ideales, vale decir, bases

de datos uniformes y una cantidad prudente información, por lo tanto, aún queda mucho por explorar en cuanto potencial y rendimiento que pueden ofrecer los algoritmos basados en ELM.

Al hablar del futuro, se tiene previsto que las ELM continuarán un sendero donde serán una opción viable y accesible para los métodos biométricos, ya que su simplicidad y velocidad representan una gran ventaja en comparación de la competencia considerada en los distintos trabajos académicos revisados.

Un punto para considerar en el futuro es utilizar las ELM con técnicas que trabajen directamente con las imágenes de las huella dactilares, sin la necesidad de algún tipo de extractor de características. De esta forma se podría evaluar el comportamiento de las ELM mediante una modalidad más profunda, y analizar los resultados permitiría realizar una comparación más justa con los resultados que se obtienen con algoritmos basados en CNN, pues, estos trabajan de forma directa con la imagen de la huella dactilar y no emplean extractores de características como fue para el caso de los trabajos revisados.

El desafío más importante que le queda a la comunidad científica y autores de este campo es implementar las ELM con bases de datos más grandes (orden de cientos de miles) para tener una visión más cercana a la realidad de los resultados, pues, las bases de datos utilizados en los experimentos de los trabajos revisados son muy pequeñas como para concluir algo de forma certera.

Agradecimientos

Los autores agradecen al Laboratory of Technological Research in Pattern Recognition (LITRP) de la Universidad Católica del Maule (https://www.litrp.cl/) y al proyecto ANID FONDECYT Regular 2020 No. 1200810, Very Large Fingerprint Classification based on a Fast and Distributed Extreme Learning Machine, Agencia Nacional de Investigación y Desarrollo, Ministerio de Ciencia, Tecnología, Conocimiento e Innovación, Gobierno de Chile.

Bibliografía

[1] D. Zabala-Blanco, M. Mora, R. Barrientos, R. Hernández García and J. Naranjo Torres, “Fingerprint Classification through Standard and Weighted Extreme Learning Machines,” *Applied Sciences,* vol. 10, 6 2020.

[2] J. Feng and A. K. Jain, “Fingerprint Reconstruction: From Minutiae to Phase,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 33, pp. 209-223, 2011.

[3] M. Galar, J. Derrac, D. Peralta, I. Triguero, D. Paternain, C. Lopez-Molina, S. García, J. M. Benítez, M. Pagola, E. Barrenechea, H. Bustince and F. Herrera, “A survey of fingerprint classification Part I: Taxonomies on feature extraction methods and learning models,” *Knowledge-Based Systems,* vol. 81, pp. 76-97, 2015.

[4] A. M. Bazen and S. H. Gerez, “Systematic methods for the computation of the directional fields and singular points of fingerprints,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 24, pp. 905-919, 2002.

[5] D. Peralta, I. Triguero, S. García, Y. Saeys, J. M. Benitez and F. Herrera, “On the use of convolutional neural networks for robust classification of multiple fingerprint captures,” *International Journal of Intelligent Systems,* vol. 33, pp. 213-230, 2018.

[6] M. Galar, J. Derrac, D. Peralta, I. Triguero, D. Paternain, C. Lopez-Molina, S. García, J. Benítez, M. Pagola, E. Barrenechea, H. Sola and F. Herrera, “A Survey of Fingerprint Classification Part II: Experimental Analysis and Ensemble Proposal,” *Knowledge-Based Systems,* vol. 81, 2 2015.

[7] R. Cappelli, D. Maio and D. Maltoni, “A Multi-Classifier Approach to Fingerprint Classification,” *Pattern Anal. Appl.,* vol. 5, pp. 136-144, 6 2002.

[8] M. Kawagoe and A. Tojo, “Fingerprint pattern classification,” *Pattern Recognition,* vol. 17, pp. 295-303, 1984.

[9] J.-H. Hong, J.-K. Min, U.-K. Cho and S.-B. Cho, “Fingerprint classification using one-vs-all support vector machines dynamically ordered with naı¨ve Bayes classifiers,” *Pattern Recognition,* vol. 41, pp. 662-671, 2 2008.

[10] A. K. Jain, S. Prabhakar and L. Hong, “A multichannel approach to fingerprint classification,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 21, pp. 348-359, 1999.

[11] M. Liu, "Fingerprint classification based on Adaboost learning from singularity features," *Pattern Recognition,* vol. 43, pp. 1062-1070, 3 2010.

[12] K. Nilsson and J. Bigün, "Localization of corresponding points in fingerprints by complex filtering," *Pattern Recognition Letters,* vol. 24, pp. 2135-2144, 9 2003.

[13] F. Saeed, M. Hussain and H. A. Aboalsamh, "Classification of Live Scanned Fingerprints using Histogram of Gradient Descriptor," in *2018 21st Saudi Computer Society National Computer Conference (NCC)*, 2018.

[14] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005.

[15] J. Yang, S. Xie, S. Yoon, D. Park, Z. Fang and S. Yang, "Fingerprint matching based on extreme learning machine," *Neural Computing and Applications,* vol. 22, 3 2013.

[16] M.-K. Hu, "Visual pattern recognition by moment invariants," *IRE Transactions on Information Theory,* vol. 8, pp. 179-187, 1962.

[17] R. C. Gonzalez, Digital Image Processing 2Nd Ed., Prentice-Hall Of India Pvt. Limited, 2002.

[18] Y. Chen, J. Yang, C. Wang and N. Liu, "Multimodal biometrics recognition based on local fusion visual features and variational Bayesian extreme learning machine," *Expert Systems with Applications,* vol. 64, 7 2016.

[19] S. Wang and Y. Wang, "Fingerprint Enhancement in the Singular Point Area," *Signal Processing Letters, IEEE,* vol. 11, pp. 16-19, 2 2004.

[20] G.-B. Huang, Q.-Y. Zhu and C.-K. Siew, "Extreme learning machine: Theory and applications," *Neurocomputing,* vol. 70, pp. 489-501, 2006.

[21] D. Zabala-Blanco, M. Mora, C. Azurdia-Meza, A. Dehghan Firoozabadi, P. Palacios and I. Soto, "Relaxation of the Radio-Frequency Linewidth for Coherent-Optical Orthogonal Frequency-Division Multiplexing Schemes by Employing the Improved Extreme Learning Machine," *Symmetry,* 4 2020.

[22] S. Ding, H. Zhao, Y. Zhang, X. Xu and R. Nie, "Extreme learning machine: algorithm, theory and applications," *Artificial Intelligence Review,* vol. 44, p. 103–115, 2015.

[23] P. L. Bartlett, "The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network," *IEEE Transactions on Information Theory,* vol. 44, pp. 525-536, 1998.

[24] W. Deng, Q. Zheng and L. Chen, "Regularized Extreme Learning Machine," in *2009 IEEE Symposium on Computational Intelligence and Data Mining*, 2009.

[25] Z. Man, K. Lee, D. Wang, Z. Cao and C. Miao, "A new robust training algorithm for a class of single-hidden layer feedforward neural networks," *Neurocomputing,* vol. 74, pp. 2491-2501, 9 2011.

[26] K. Zhang and M. Luo, "Outlier-robust extreme learning machine for regression problems," *Neurocomputing,* vol. 151, pp. 1519-1527, 3 2015.

[27] Q. Shen, X. Ban, R. Liu and Y. Wang, "Decay-weighted extreme learning machine for balance and optimization learning," *Machine Vision and Applications,* vol. 28, pp. 1-11, 10 2017.

[28] W. Zong, G.-B. Huang and Y. Chen, "Weighted extreme learning machine for imbalance learning," *Neurocomputing,* vol. 101, pp. 229-242, 2013.

[29] C. Lu, H. Ke, G. Zhang, Y. Mei and H. Xu, "An improved weighted extreme learning machine for imbalanced data classification," *Memetic Computing,* vol. 11, 3 2019.

[30] Y. Akbulut, A. Sengur, Y. Guo and F. Smarandache, "A Novel Neutrosophic Weighted Extreme Learning Machine for Imbalanced Data Set," *Symmetry,* vol. 9, 8 2017.

[31] M. Maimaitiyiming, V. Sagan, P. Sidike and M. T. Kwasniewski, "Dual Activation Function-Based Extreme Learning Machine (ELM) for Estimating Grapevine Berry Yield and Quality," *Remote Sensing,* vol. 11, 2019.

[32] Y. Chen, J. Yang, C. Wang and D. S. Park, "Variational Bayesian extreme learning machine," *Neural Computing and Applications,* vol. 27, pp. 185-196, 1 2016.

[33] "Synthetic Fingerprint Generation," in *Handbook of Fingerprint Recognition*, London, Springer London, 2009, p. 271–302.

[34] D. Maltoni, D. Maio, A. K. Jain and S. Prabhakar, Handbook of fingerprint recognition, Springer Science & Business Media, 2009.

[35] J. G. Moreno-Torres, J. A. Saez and F. Herrera, "Study on the Impact of Partition-Induced Dataset Shift on <formula formulatype="inline"><tex Notation="TeX">k</tex></formula>-Fold Cross-Validation," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 23, pp. 1304-1312, 2012.

[36] G. Huang, S. Song, J. Gupta and C. Wu, "Semi-Supervised and Unsupervised Extreme Learning Machines," *IEEE Transactions on Cybernetics,* vol. 44, pp. 1-1, 12 2014.

[37] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009.

[38] T. Zia, M. Ghafoor, A. Tariq and I. Taj, "Robust Fingerprint Classification with Bayesian Convolutional Networks," *IET Image Processing,* vol. 13, 2 2019.

[39] N. Huang, C. Yuan, G. Cai and E. Xing, "Hybrid Short Term Wind Speed Forecasting Using Variational Mode Decomposition and a Weighted Regularized Extreme Learning Machine," *Energies,* vol. 9, p. 989, 11 2016.

[40] D. Peralta, M. Galar, I. Triguero, D. Paternain, S. García, E. Barrenechea, J. M. Benítez, H. Bustince and F. Herrera, "A survey on fingerprint minutiae-based local matching for verification and identification: Taxonomy and experimental evaluation," *Information Sciences,* vol. 315, pp. 67-87, 2015.

[41] S. Karungaru, K. Fukuda, M. Fukumi and N. Akamatsu, "Classification of fingerprint images into individual classes using Neural Networks," 2008.

[42] A. Cotrina-Atencio, J. L. A. Samatelo and E. O. T. Salles, "Evaluation of GMM approach to fingerprint classification," in *ISSNIP Biosignals and Biorobotics Conference 2011*, 2011.

[43] S. Shapoori and N. Allinson, "GA-Neural Approach for Latent Finger Print Matching," in *2011 Second International Conference on Intelligent Systems, Modelling and Simulation*, 2011.

[44] L. Hong, Y. Wan and A. Jain, "Fingerprint image enhancement: algorithm and performance evaluation," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 20, pp. 777-789, 1998.

[45] D. Maio, D. Maltoni, R. Cappelli, J. L. Wayman and A. K. Jain, "FVC2004: Third Fingerprint Verification Competition," in *Biometric Authentication*, Berlin, 2004.

[46] D. Maio, D. Maltoni, R. Cappelli, J. Wayman and A. Jain, "FVC2002: Second Fingerprint Verification Competition," 2002.

[47] J. C. Yang and D. S. Park, "A fingerprint verification algorithm using tessellated invariant moment features," *Neurocomputing,* vol. 71, pp. 1939-1946, 2008.

[48] J. Yang, N. Xiong and A. Vasilakos, "Two-Stage Enhancement Scheme for Low-Quality Fingerprint Images by Learning From the Images," *Human-Machine Systems, IEEE Transactions on,* vol. 43, pp. 235-248, 3 2013.

[49] L. Fausett, *Fundamentals of neural networks: architectures, algorithms, and applications,* Prentice-Hall, Inc., 1994.

[50] P. J. Phillips, H. Moon, S. A. Rizvi and P. J. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 22, pp. 1090-1104, 2000.

[51] A. A. Mohammed, R. Minhas, Q. M. J. Wu and M. Sid-Ahmed, "Human face recognition based on multidimensional PCA and extreme learning machine," *Pattern Recognition,* vol. 44, pp. 2588-2597, 10 2011.

[52] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems*, 2012.

[53] G. Jing, Y.-F. Liu, J.-Y. Chang and J.-D. Lee, "Fingerprint classification based on decision tree from singular points and orientation field," *Expert Systems with Applications: An International Journal,* vol. 41, p. 752–764, 2 2014.

[54] H.-W. Jung and J.-H. Lee, "Noisy and incomplete fingerprint classification using local ridge distribution models," *Pattern Recognition,* vol. 48, 2 2015.

[55] K. Dorasamy, L. Webb, J. Tapamo and N. P. Khanyile, "Fingerprint classification using a simplified rule-set based on directional patterns and singularity features," in *2015 International Conference on Biometrics (ICB)*, 2015.

[56] J. Yang, N. Xiong, A. V. Vasilakos, Z. Fang, D. Park, X. Xu, S. Yoon, S. Xie and Y. Yang, "A Fingerprint Recognition Scheme Based on Assembling Invariant Moments for Cloud Computing Communications," *IEEE Systems Journal,* vol. 5, pp. 574-583, 2011.

[57] J. Yang and D. Park, "Fingerprint Verification Based on Invariant Moment Features and Nonlinear BPNN," *International Journal of Control, Automation and Systems,* vol. 6, 12 2008.

[58] C.-C. Chang and C.-J. Lin, *A library for support vector machines (version 2.3).*