

Proyecto de reconocimiento de rostros

Juan Carlos Gutiérrez

Juan Carlos Gutiérrez Cáceres es magíster scientiae por la Universidade de São Paulo (Brasil) e ingeniero de Sistemas por la Universidad Nacional de San Agustín (Arequipa). Es socio fundador de la Sociedad Peruana de Computación, así como de la Sociedad Peruana de Reconocimiento de Patrones. Actualmente es investigador de la cátedra CONCYTEC-UNSA. Se desempeña como profesor en la Universidad Católica San Pablo (Arequipa) y en la Universidad Nacional San Agustín; en esta última universidad enseña también en diversas maestrías.

Proyecto de reconocimiento de rostros

Resumen

Este trabajo presenta un prototipo eficaz para el reconocimiento de rostros humanos que usa la transformada *wavelet* para la reducción de la dimensión de la imagen, lo cual es importante en aplicaciones reales (eficaces), además de ser invariante ante transformaciones geométricas como rotación, traslación y escala, ya que utiliza la normalización mediante la aplicación de momentos regulares. Así mismo, tiene una alta tolerancia a las fallas al utilizar una red neuronal basada en la teoría del caos en el proceso de reconocimiento. Las pruebas demostraron que el método propuesto alcanzó un alto porcentaje de efectividad.

Palabras clave: reconocimiento de rostros, transformada *wavelet*, momentos, matriz pseudoinversa, memoria asociativa caótica.

Las investigaciones sobre tecnología de reconocimiento de rostros han cobrado mucha importancia en los últimos veinte años: neurocientíficos, físicos e ingenieros vienen proponiendo diferentes métodos y tecnologías de reconocimiento de rostros basados en los gestos y características faciales (Cao, Yin, Sun, y Tang, 2010; Intrator, Reisfeld, y Yeshurun, 1996; Kondo y Yan, 1999; Pinto, DiCarlo, y Cox, 2009). Este interés surge debido a las potenciales aplicaciones de esta tecnología: sistemas de control de seguridad, identificación de criminales, robótica e incluso el aprovechamiento de redes sociales como *Facebook*, donde se pueden encontrar gran cantidad de imágenes de rostros (Becker y Ortiz, 2008).

Se pueden dividir en dos grupos las diferentes técnicas y métodos propuestos: los estáticos (imágenes 2D) y los dinámicos (video) (Chellappa, Wilson, y Sirohey, 1995; W. Zhao, Chellappa, Phillips, y Rosenfeld, 2003). Las técnicas estáticas son las aplicaciones más conocidas; se basan en fotografías de rostros, donde aspectos como la iluminación, la orientación, la rotación, el tamaño o cambios en las expresiones faciales (Chang, Hu, Feris, y Turk, 2006) son puntos que se deben tomar en cuenta y se deben controlar (Lai, Yuen, y Feng, 2001). Las técnicas dinámicas, por su parte, toman en cuenta aspectos adicionales, ya que en una escena de video aparecen también elementos que no corresponden a un rostro, amén de que las imágenes generadas por una videocámara suelen ser de baja calidad (Campos, De Campos, y Cesar, 2000; Hu, Harguess, y Aggarwal, 2009).

Otra forma de clasificar las técnicas de reconocimiento de rostros es dividir las que se basan en componentes constituyentes y las que se basan en el rostro (Liu, 2006; Samal y Lyengar, 1992). Las primeras se apoyan en la interrelación entre las diferentes características faciales como nariz, boca, ojos, forma del rostro, etcétera (Jeng, Liao, Han, Chern, y Liu, 1998; Lam y Yan, 1996). La técnica de *template* deformable (Jin, Lou, Yang, y Sun, 2007) es una de las más representativas de este tipo; Yuille y sus colaboradores, por ejemplo, la utilizaron para detectar ojos en la imagen de un rostro (Yuille, Hallinan, y Cohen, 1996). Esta técnica ayuda a resolver algunos problemas de invariancia de transformaciones geométricas, y proporciona alta flexibilidad en la manipulación de características faciales, tales como ojos, nariz, boca, entre otras. Sin embargo, la garantía de este método depende de la correcta detección de las características faciales, por lo que la resolución de la imagen debería ser relativamente alta (Lai, et al., 2001).

Las técnicas basadas en el rostro utilizan la imagen bidimensional con variaciones de intensidad. Aquí el reconocimiento se realiza mediante la búsqueda de regularidades estadísticas subyacentes. El método de análisis de componentes principales (PCA) y sus variaciones son un ejemplo típico de esta técnica. Entre sus ventajas se cuenta que con-

sideran el rostro de manera global y que la resolución de la imagen puede ser relativamente baja (Bianchi, 2006; Sirovich y Kirby, 1987).

En este artículo se proponen dos fases para el reconocimiento de rostros: (1) extracción de características, y (2) clasificación, las cuales corresponden a un sistema de reconocimiento de patrones estándar. La extracción de características se realizará mediante la utilización de la transformada *wavelet* (Bianchi, 2006; Gomes, Brancalion, y D., 2007; Oliveira, 2007; Pordeus, 2006), lo cual reducirá la dimensión de la imagen para que el proceso de clasificación sea computacionalmente más ligero. También se realizará una normalización de las transformaciones geométricas. En la segunda fase se utilizará una red neuronal basada en teoría del caos para la clasificación de los patrones (L. Zhao, Gutierrez Caceres, Damiance, y Szui, 2003, 2006).

La transformada wavelet para la extracción de características y reducción de la dimensión de una imagen

Según Lai y sus colaboradores, la información representada por medio de frecuencias espaciales es importante en el reconocimiento de rostros (Lai, et al., 2001). Las imágenes en banda de baja y alta frecuencia tienen un rol diferente en la descripción de una imagen: la baja frecuencia contribuye a la descripción global de la imagen, mientras que la alta corresponde a sus detalles finos. Lai y sus colaboradores también demuestran que los efectos de diferentes expresiones faciales se pueden atenuar eliminando los componentes de alta frecuencia, y que los componentes de baja frecuencia son suficientes para su reconocimiento.

La transformada wavelet.

Las *wavelets* son en la actualidad materia de estudio para múltiples aplicaciones (Daubechies, 1990; Grossmann y Morlet, 1984; Mallat, 1989). Para entender el origen de la transformada *wavelet* es necesario entender la transformada de Fourier, la cual tiene por objetivo transformar una señal del dominio de espacio para el dominio de frecuencia. La transformada de Fourier está dada por la siguiente ecuación:

$$F[u] = \int f[t] e^{-i2\pi ut} dt \quad [1]$$

Esta ecuación corresponde a la transformada de Fourier de una señal continua $j(t)$, la cual presenta algunas deficiencias, debido a que esta transformación no está localizada en el dominio espacial. Debido a ello no puede representar adecuadamente los cambios que suceden en la señal. Esto sucede porque la transformada se basa en la integración de toda la función para el cálculo de cada frecuencia.

Para solucionar ese problema, Gabor adaptó la transformada de Fourier para analizar sólo una pequeña porción de la señal en un tiempo, proceso llamado *transformada por ventanas de Fourier (WFT)*. En ese caso una ventana de observación se desplaza en el dominio de tiempo, y se calcula la transformada de Fourier a partir de la porción *visible* de la señal para cada posición de la ventana (Gabor, 1946). Considerando $j(t)$ como la función que cumple el papel de ventana, se puede definir la transformada por ventanas de Fourier de una señal continua $f(t)$ como:

$$F(u, b) = \int j(t-b)f[t]e^{i2\pi ut} dt \quad [2]$$

El problema ahora es que una vez que se define un tamaño para la ventana en el dominio de tiempo, ésta permanece constante para todas las frecuencias. Puede suceder que alguna señal necesite un enfoque más flexible, es decir, que el tamaño de la ventana pueda cambiar con el objetivo de detectar el contenido local de la frecuencia (Daubechies, 1992; Gomez, Velho, y Goldenstein, 1997).

La transformada *wavelet* puede verse como una técnica por ventanas con regiones de dimensión variable, donde las *wavelets*, a diferencia del caso de la transformada de Fourier, tienen como base una función de duración limitada, esto es, de soporte compacto, que es una propiedad cuyo dominio es distinto de cero en una extensión finita, e igual a cero en todo el resto. Es interesante la utilización de las *wavelets* en el caso específico del análisis de imágenes, pues los cambios de regiones o bordes pueden ser detectados más fácilmente.

La definición de una transformada *wavelet* considerando una señal continua está dada por:

$$F(a, b) = \int f(t)\psi_{a,b}(t)dt \quad [3]$$

En esta ecuación los parámetros a y b varían continuamente en \mathbb{R} , y las funciones $\psi_{a,b}$ se denominan *wavelets*, y se definen de la siguiente forma:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad [4]$$

La transformada *wavelet* para señales discretas se define del siguiente modo:

$$F_{m,n}(a,b) = a_0^{-m/2} \int f(t) \psi(a_0^{-m} t - nb_0) dt \quad [5]$$

Puede verse claramente que el comportamiento de esta función se basa en dilataciones y traslaciones a partir de una *wavelet* madre ψ . En ambos casos la *wavelet* madre debe satisfacer la siguiente propiedad:

$$\int \psi(t) dt = 0 \quad [6]$$

Observando la ecuación 3 se puede ver que la transformada *wavelet* depende de dos parámetros, a y b , que corresponden a las informaciones de escala y tiempo respectivamente (Daubechies, 1992; Gomez, et al., 1997; Misiti, Oppenheim, Poggi, y Misiti, 2001).

Para la transformada *wavelet* discreta se distinguen dos enfoques: sistemas redundantes discretos (*frames*) y bases ortonormales de *wavelet* (Daubechies, 1992). Este último considera la estrategia de análisis de multirresolución (Mallat, 1989), que consiste en la descomposición de una señal (imagen) en distintas escalas.

Una descomposición multirresolución permite obtener una interpretación invariante de escala de la imagen. La escala de una imagen cambia en razón de la distancia entre la escena y el centro óptico de la cámara. Cuando la escala de la imagen es modificada, la interpretación de la imagen no debería cambiar. Una representación multirresolución puede ser parcialmente invariante de escala si la secuencia de parámetros de la resolución $(V_j)_{j \in \mathbb{Z}}$ varía exponencialmente (Mallat, 1989).

Si V_j es un espacio vectorial, es necesario definir sus funciones base¹. Las funciones base para los espacios V_j se denominan *funciones escalares*, y usualmente se denotan con el símbolo ϕ . Una base simple para V_j se da por el conjunto de *funciones caja escaladas y trasladadas*: $\phi_i^j(x) := \phi(2^j x - i) \quad i=0, \dots, 2^j-1$ donde: $\phi(x) := \begin{cases} 1 & \text{si } 0 \leq x < 1 \\ 0 & \text{en cualquier otro caso} \end{cases}$

Ahora es necesario escoger un *producto interno* definido sobre los espacios de vectores V_j . Éste puede ser:

$$\langle f | g \rangle := \int_0^1 f(x) g(x) dx \quad [7]$$

1. *Grosso modo* se podría decir que una *base* para un espacio vectorial consiste en un conjunto mínimo de vectores linealmente independientes, de los cuales todos los demás en el espacio vector se pueden generar a través de combinaciones lineales.

Se dice que dos vectores u y v son *ortogonales* bajo un producto interno escogido si $\langle u | v \rangle = 0$. Ahora es posible, por tanto, definir un nuevo espacio vectorial W_j como el *complemento ortogonal* de V_j en V_{j+1} . En otras palabras, W_j es el espacio de todas las funciones en V_{j+1} ortogonales a todas las funciones en V_j en el producto interno elegido.

Una colección de funciones $\psi_i^j(x)$ que generan W_j se denomina *wavelet*. Estas funciones base tienen dos propiedades importantes:

- * Las funciones base $\psi_i^j(x)$ de W_j junto a las funciones base ϕ_i^j de V_j forman una base para V_{j+1} .
- * Cada función base $\psi_i^j(x)$ de W_j es ortogonal a cada función base ϕ_i^j de V_j en el producto interno escogido.

El análisis de multirresolución consiste en una secuencia de espacios de aproximación sucesivos V_j . Específicamente los subespacios cerrados V_j satisfacen:

$$\dots V_{-2} \subset V_{-1} \subset V_0 \subset V_1 \subset V_2 \dots \quad [8]$$

con

$$\overline{\bigcup_{j \in \mathbb{Z}} V_j} = L^2(\mathbb{R}) \quad [9]$$

$$\bigcap_{j \in \mathbb{Z}} V_j = \{0\} \quad [10]$$

Todos los espacios son una versión escalada del espacio central V_0 ,

$$f(x) \in V_j \Leftrightarrow f(2^j x) \in V_0 \quad [11]$$

y

$$f(x) \in V_0 \Rightarrow f(x-n) \in V_0, \text{ para todo } n \in \mathbb{Z} \quad [12]$$

indica la invariancia de V_0 bajo traslaciones enteras. Ahora se necesita saber que $\phi \in V_0$

$$\phi_{0,n}(x) = \phi(x-n), n \in \mathbb{Z} \quad [13]$$

constituye una base ortonormal para V_0 .

Extracción de características y reducción de la dimensión de la imagen usando wavelet.

El algoritmo para la transformada *wavelet* bidimensional se basa en el producto tensorial de sus funciones base unidimensionales, las cuales mantienen la propiedad de separabilidad. De esa forma, la transformada puede implementarse en dos dimensiones con la aplicación de los algoritmos unidimensionales correspondientes en filas y en columnas de la imagen (Stollnitz, Deroose, y Salesin, 1996).

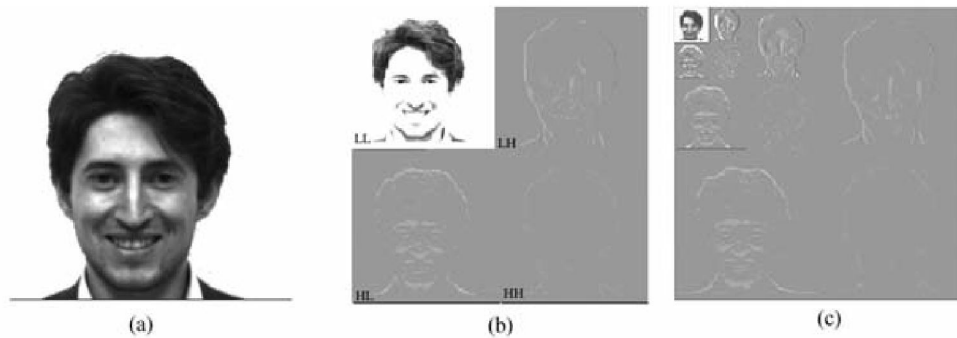


Figura 1

Descomposición *wavelet* 2D de la imagen de un rostro.

En la figura 1 se puede ver la aplicación de las *wavelets* a la imagen de un rostro: la figura 1a corresponde a la imagen original del rostro; la figura 1b muestra la descomposición *wavelet* de la imagen en cuatro sub-bandas: *LL* corresponde a los componentes de baja frecuencia en dirección vertical y horizontal, lo que corresponde, respectivamente, a las filas y columnas de la imagen original; *LH* corresponde a los componentes de baja frecuencia (*L*) en dirección horizontal y alta frecuencia en dirección vertical (*H*); de forma similar sucede en *HL* y *HH*, que corresponden a las frecuencias altas.

Este proceso de descomposición se puede aplicar en forma recursiva sobre la sub-banda de baja frecuencia (la imagen reducida) para así lograr una versión comprimida de la imagen original, lo cual será aprovechado en este trabajo. En la figura 1c se muestra el resultado después de aplicar tres niveles de descomposición.

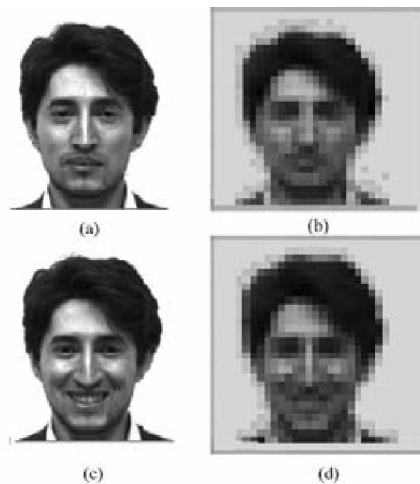


Figura 2
Descomposición wavelet de la imagen de un rostro.

Las principales diferencias en el reconocimiento de un rostro son las transformaciones geométricas que puede sufrir la imagen, así como los diferentes gestos, que varían principalmente en los ojos, la boca y los músculos de la cara. Como se puede ver en la figura 2, los gestos del rostro pueden ser atenuados. La figura 2a corresponde a un rostro sin gestos y la figura 2b corresponde a sus componentes en baja frecuencia después de tres niveles de descomposición *wavelet*. La figura 2c, por su parte, corresponde a un rostro con gesto de sonrisa, mientras que la figura 2d corresponde a los componentes de baja resolución de ésta. Puede apreciarse que las figuras 2b y 2d son similares por causa de la aplicación de tres niveles de descomposición *wavelet*.

Normalización con respecto a transformaciones geométricas

Uno de los principales problemas de un sistema automático de reconocimiento de imágenes es poder identificar un patrón a pesar de sufrir cierto tipo de transformaciones geométricas. Muchos trabajos han intentado resolver este problema. Por ejemplo, Ravichandran y Yegnanarayana utilizan la normalización para imágenes de objetos simples (Ravichandran y Yegnanarayana, 1991); otros investigadores presentan formas de resolver este problema en reconocimiento de rostros (Haddadnia, Faez, y Moallem, 2001; Lai, et al., 2001).

Cuando un objeto se encuentra bajo algún tipo de transformación geométrica, su posición varía en el espacio; sin embargo, ciertas relaciones entre algunos puntos del objeto se mantienen. Por ejemplo, el centro de masa (o centro de gravedad) de la imagen es

una buena referencia para poder obtener invariancia ante alguna posible variación del desplazamiento del objeto en la imagen. Existen otras funciones que nos ayudan a obtener invariancia de las otras transformaciones; éstas son llamadas *momentos*.

El centro de masa es un momento que está definido por la siguiente ecuación:

$$x_c = \sum \frac{x_i}{N} \quad y_c = \sum \frac{y_i}{N} \quad [14]$$

donde x_i, y_i corresponde a valores no nulos de la imagen; N es el número de componentes de x_i y y_i .

Otros puntos nos ayudan a la normalización de la rotación del objeto, y son llamados *momentos centrales*, los cuales están dados por:

$$P_x = \frac{\sum F_i x_i}{\sum F_i} \quad y \quad P_y = \frac{\sum F_i y_i}{\sum F_i} \quad [15]$$

donde

$$F_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} \quad [16]$$

donde P_x y P_y es llamado *primer momento central*. Existe otro momento que nos ayuda a normalizar la imagen, llamado *segundo momento central* (Q_x, Q_y), y se puede calcular usando la ecuación 15, con la diferencia de que la función F_i está dada para este caso por la siguiente ecuación:

$$F_i = \frac{1}{\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}} \quad [17]$$

Dados estos momentos, la normalización de la imagen puede estar dada por los siguientes pasos:

1. Calcular el centro de gravedad con la ecuación 14.
2. Calcular pesos w_i que correspondan a la distancia de cada píxel al centro de masa, los cuales están dados por:

$$W_i = \sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} \quad [18]$$

3. Calcular los momentos centrales por:

$$\begin{aligned}
 X_{w1} &= \frac{\sum (W_i x_i)}{\sum W_i} \\
 Y_{w1} &= \frac{\sum (W_i y_i)}{\sum W_i} \\
 X_{w2} &= \frac{\sum (\frac{x_i^2}{W_i})}{\sum \frac{1}{W_i}} \\
 Y_{w2} &= \frac{\sum (\frac{y_i^2}{W_i})}{\sum \frac{1}{W_i}}
 \end{aligned}
 \tag{19}$$

4. Calcular la distancia entre los momentos centrales dada por:

$$d = \sqrt{(x_{w1} - x_{w2})^2 + (y_{w1} - y_{w2})^2}
 \tag{20}$$

5. Para todos los puntos P_i ($i = 1, 2, \dots, N$) se redefinen nuevos puntos por:

$$\begin{aligned}
 X_i &= \frac{(X_i - X_{w1})(X_{w1} - X_{w2})}{d} + \\
 &\quad \frac{(Y_i - Y_{w1})(Y_{w1} - Y_{w2})}{d} \\
 Y_i &= \frac{(X_{w1} - X_i)(Y_{w1} - Y_{w2})}{d} + \\
 &\quad \frac{(Y_i - Y_{w1})(X_{w1} - X_{w2})}{d}
 \end{aligned}
 \tag{21}$$

6. Para obtener invariancia a escala, cada nuevo punto será dividido por la distancia d .

La binarización de la imagen del rostro ayuda a la normalización de las transformaciones geométricas. El empleo de un simple *threshold* puede aplicarse para tener como resultado una imagen binaria que representará al rostro, como se puede apreciar en la figura 3.



Figura 3
Binarización de una imagen.

Aplicando los pasos descritos anteriormente vemos que es posible encontrar una invariancia en las transformaciones geométricas. En la figura 4 se puede apreciar esto: la figura 4a muestra la imagen de un rostro girado a 45°; la figura 4b corresponde al mismo rostro con la diferencia de que éste es de menor tamaño y tiene otra rotación, de 90°; la figura 4c corresponde a una normalización en desplazamiento y en tamaño de la figura 4a; y lo mismo ocurre con la figura 4d, que corresponde a la normalización de la figura 4b; finalmente, en la figura 4e se muestra el resultado de la normalización total de la figura 4a, y lo mismo ocurre en la figura 4f, que es el resultado final de la figura 4b.

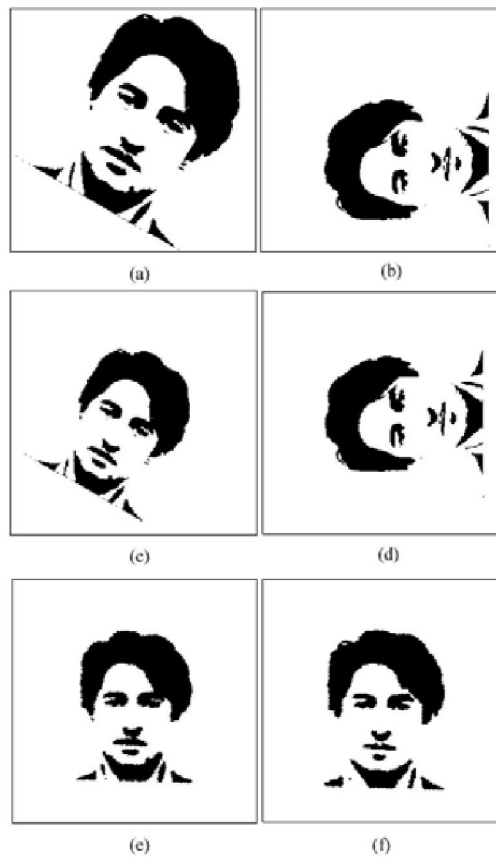


Figura 4
Proceso de normalización de la imagen de un rostro.

Modelo de la red neuronal caótica para el reconocimiento de patrones

Recientemente se vienen desarrollando nuevas redes neuronales caóticas utilizando directamente las propiedades dinámicas del caos. Zhao y sus colaboradores propusieron una nueva red neuronal basada en teoría del caos (L. Zhao, et al., 2003, 2006). Al permitir incrementar la capacidad de almacenamiento, este modelo demostró tener una

mejor eficacia en comparación con modelos tradicionales como el propuesto por Hopfield: al usar la red de Hopfield el almacenamiento de rostros solamente puede ser reducido a 3 ó 4 rostros; sin embargo, en el caso de la red caótica se almacenaron hasta 20 rostros y se recuperaron todos correctamente. Por ello se seleccionó la red neuronal caótica para la etapa de reconocimiento.

Este modelo funciona en dos etapas: (1) etapa de aprendizaje; y (2) etapa de reconocimiento. En la primera el mismo algoritmo de matriz pseudoinversa se utiliza para almacenar los patrones en puntos fijos. En la etapa de reconocimiento se utilizan las dinámicas caótica y periódica del mapa logístico: la órbita periódica representa el patrón reconocido, y la órbita caótica representa un estado de búsqueda. Como ya se mencionó, el mapa logístico usado en este modelo se puede sustituir por cualquier mapa que presente dinámica caótica y periódica, de periodo deseado.

La etapa de aprendizaje.

El proceso de aprendizaje es similar al presentado en el modelo anterior utilizando la matriz pseudoinversa Q^+ .

Para almacenar patrones se utilizan los puntos fijos de la ecuación:

$$WP = P \quad [22]$$

donde W es la matriz de conexión, pudiendo ser determinada por la matriz pseudoinversa (Hertz, Krogh, y Palmer, 1991).

$$W = P P^+ \quad [23]$$

Para poder aprovechar la estructura simétrica de la bifurcación de duplicación de periodo, en este modelo los valores originales de los patrones $p_i^{(j)}$ se transforman por la siguiente regla antes del proceso de almacenamiento:

$$Tp_i = 2 \left(i - \frac{V}{2} \right) + 1 \quad [24]$$

donde i es un posible valor de los patrones, $i = 1, 2, \dots, V$. La transformación distribuye los valores de forma simétrica sobre cero.

La etapa de reconocimiento.

La dinámica del modelo está dada por las siguientes ecuaciones:

$$x_i(t+1) = z_i(\tau) x_i(t) (1 - x_i(t)) \quad [25]$$

$$Z_i(\tau) = \begin{cases} A_p & \text{si } |s(x_i(t)) - \sum_{j=1}^N w_{ij} s(x_j(t))| \\ A_c & \text{en cualquier otro caso} \end{cases} \quad [26]$$

$$s(x) = \begin{cases} T_{p1} & \text{si } x > (X_1 + X_2)/2 \\ T_{p2} & \text{si } (X_1 + X_2)/2 > x > (X_2 + X_3)/2 \\ \dots & \dots \\ T_{pv-1} & \text{si } (X_{v-2} + X_{v-1})/2 > x > (X_{v-1} + X_v)/2 \\ T_{pv} & \text{si } x > (X_{v-1} + X_v)/2 \end{cases} \quad [27]$$

La ecuación 25 corresponde al mapa logístico, siendo $x_i(t)$ la variable de estado del i -ésimo elemento en la red, para $i = 1, 2, \dots, N$. En el mapa logístico original, $z_i(\tau)$ corresponde al parámetro de bifurcación; pero aquí la variable $z_i(\tau)$ define la dinámica de cada elemento. $z_i(\tau)$ puede tomar uno de los siguientes valores: A_p y A_c . Si $z_i(\tau) = A_p$, el comportamiento del i -ésimo elemento corresponde a la dinámica periódica. Si $z_i(\tau) = A_c$ la dinámica del i -ésimo elemento es caótica. Así, la dinámica de cada elemento salta entre los estados periódico y caótico, dependiendo del valor retornado por la función de la ecuación 27.

La variable x se evalúa en cada iteración t , mientras que z se evalúa en cada iteración Vt , donde V corresponde al número de posibles valores de los patrones almacenados, y también a la órbita periódica seleccionada. Podemos percibir que cada punto periódico se utiliza para representar un posible valor del patrón. Por ejemplo, si fueran almacenados patrones binarios, $V = 2$, entonces, A_p sería escogido donde ocurra un ciclo de periodo 2. En este caso, $z_i(\tau)$ sería evaluada cada dos iteraciones. Un punto periódico representa el valor 1 y otro representa el valor 0. Finalmente la ecuación 27 define el valor correspondiente entre cada valor transformado Tp_i para cada variable de estado $x_i(t)$ del mapa logístico.

A continuación se describirá el funcionamiento del modelo cuando $V = 4$. Primeramente, un conjunto de patrones se almacenan en la memoria usando el algoritmo de aprendizaje descrito anteriormente en la ecuación 23. En la etapa de reconocimiento se selecciona el valor de A_p , que corresponde a una órbita con periodo 4, así como el valor de A_c , que corresponde a una órbita caótica. Los $x_i(0)$ son colocados inicialmente en la órbita A_c . En ese momento todos los elementos son inicialmente caóticos. Debido a la propiedad de ergodicidad del caos, cada elemento en algún instante se puede aproximar a un estado en el cual $|s(x_i(t)) - \sum_{j=1}^N w_{ij} s(x_j(t))|$ es pequeño. Eso significa que el i -ésimo elemento se está aproximando del punto fijo definido por el proceso de almacenamiento. Cuando esto sucede, la primera condición de la ecuación 26 se satisface. Entonces el i -ésimo elemento es llevado para la órbita periódica. Si ese elemento converge en el punto periódico correcto, permanecerá allí; de lo contrario, la segunda condición de la ecuación 26 se satisfará, y el mismo elemento retornará al estado caótico. De este modo, los elementos están saltando entre los estados periódico y caótico hasta encontrar sus estados correctos; en otras palabras, la diferencia $|s(x_i(t)) - \sum_{j=1}^N w_{ij} s(x_j(t))|$ es minimizada. El modelo termina su proceso cuando todos los elementos llegan a sus puntos periódicos correctos y el patrón almacenado es recuperado.

En resumen, el modelo presenta dos estados: el periódico, que representa el patrón recuperado, y el estado caótico, que permite al elemento buscar su estado correcto.

Red caótica para el almacenamiento de rostros.

Para utilizar la red de caótica como memoria asociativa para almacenar las imágenes, cada píxel deberá corresponder a una neurona. Por tal motivo es importante tener en cuenta la dimensión de la imagen, puesto que eso determinará el tamaño de la red y afectará su desempeño. La red caótica —al igual que la de Hopfield— trabaja con valores binarios o bipolares; por tal motivo, la imagen de un rostro debe ser binarizada para poder ser utilizada en la red, como se puede apreciar en la figura 3.

Un patrón de entrada para la red corresponde a un vector unidimensional de tamaño $n \times m$ de componentes, donde n y m corresponden a la dimensión de la imagen usada. El vector puede tener valores de -1 ó 1, donde 1 corresponde al color negro y -1 al blanco de la figura binarizada.

Método

El modelo seguido para la elaboración del prototipo se da en las dos etapas clásicas vistas anteriormente, pero organizadas en cinco *pasos*, como se puede ver en la figura 5.

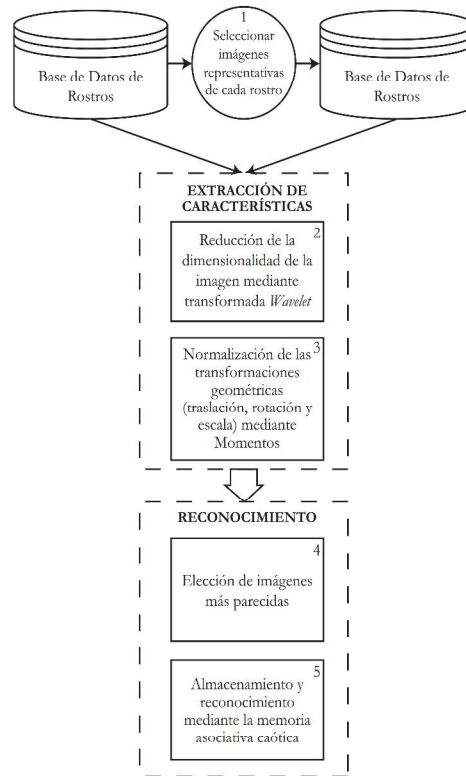


Figura 5

Diagrama de bloques del para el modelo propuesto.

1. Se propone la elección de rostros representativos de cada persona, puesto que en la base de datos se encuentran imágenes con diferentes gestos de un mismo sujeto. Se denominó *prototipo* a la imagen representativa. Para nuestro caso elegimos un rostro sin gestos (normal), el cual pasará a formar parte de una nueva base de datos donde sólo se encontrarán los prototipos.
2. En esta fase se realiza la aplicación de la transformada *wavelet*. El resultado será la reducción de la dimensión de la imagen y la atenuación de ciertos gestos del rostro, como se mostró en la figura 2.
3. En esta etapa se lleva a cabo la normalización de la imagen. Debe considerarse que cuando se toma una foto a una misma persona, no siempre se obtienen imágenes

iguales debido a variaciones principalmente en traslación, rotación y tamaño. Por lo tanto, como salida de esta fase obtendremos una imagen sin transformaciones geométricas.

4. Las memorias asociativas, en general, poseen ciertas limitaciones en la capacidad de almacenamiento de patrones. Para ayudar a incrementar la capacidad de almacenamiento del sistema se realizará una correlación entre los patrones almacenados y aquél que se desea reconocer, y se elegirán sólo los más parecidos, los cuales pueden ser tres prototipos de nuestra base de datos. Sería ideal utilizar estructuras de datos adecuadas para esta parte —estructuras métricas, por ejemplo—, pues con ello se optimizaría la eficacia del sistema en aplicaciones con gran cantidad de rostros almacenados; esto, sin embargo, deberá ser materia de un futuro trabajo.
5. Finalmente, se construye la memoria asociativa caótica y se realiza el entrenamiento con los prototipos elegidos en la fase anterior. El reconocimiento se realiza con la presentación de la imagen que se desea reconocer, dejando a la red elegir entre los rostros almacenados.

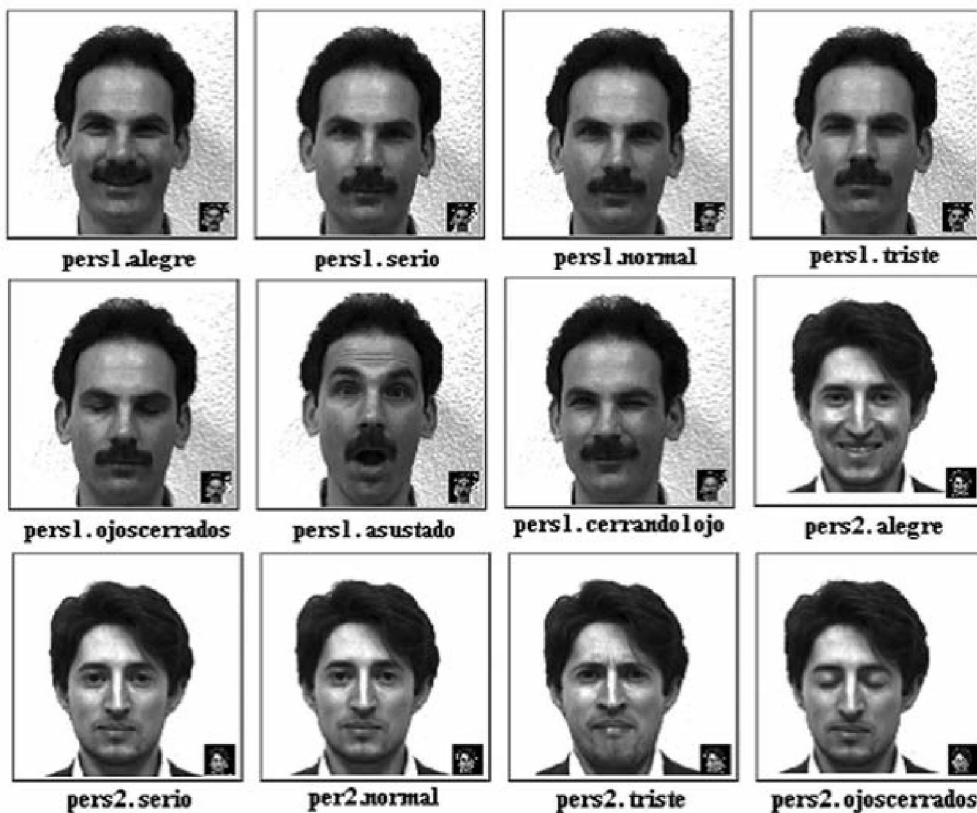


Figura 6
Algunas imágenes de rostros de la base de datos Yale.

Resultados

Para la aplicación del prototipo implementado se utilizó la base de datos estándar de la Universidad de Yale. La base de datos contiene las imágenes de los rostros de 15 personas, las cuales muestran expresiones faciales diferentes para cada una, tales como alegría, susto, tristeza, seriedad, etcétera, como se aprecia en la figura 6. Estas imágenes pueden tener transformaciones mínimas en rotación, traslación y tamaño. Las imágenes tienen una resolución de 256 x 256 píxeles, y están en formato bmp de 256 colores

Se implementaron tres tipos de *wavelets*: la Haar, la Daubechies 4 y la biortogonal. Los resultados de la descomposición en dos niveles se pueden observar en la figura 7. La selección de la *wavelet* depende de diferentes aspectos, como la calidad de la imagen o la complejidad computacional (Grgic, Grgic, y Zovko-Cihlar, 2001). Como se observa en la figura 7, la diferencia entre las diferentes *wavelets* no es tan significativa, puesto que después de la descomposición se ha de realizar la binarización, lo cual minimiza aún más la diferencia. Para la compresión de imágenes se suele usar la biortogonal, dado que mantiene la mayor calidad de la imagen (Grgic, et al., 2001).

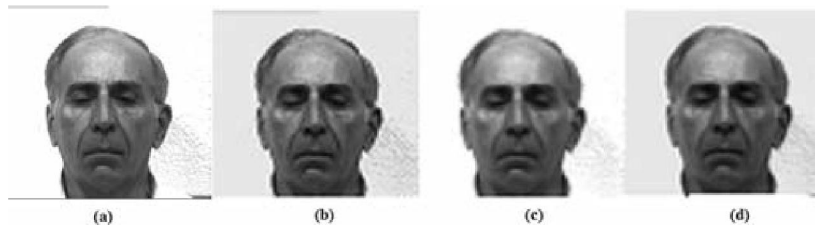


Figura 7

(a) Imagen original; (b) biortogonal; (c) Haar; (d) Daubechies.

Daubechies (1992) utiliza los siguientes filtros:

$$\frac{1 + \sqrt{3}}{4\sqrt{2}}, \frac{3 + \sqrt{3}}{4\sqrt{2}}, \frac{3 - \sqrt{3}}{4\sqrt{2}}, \frac{1 - \sqrt{3}}{4\sqrt{2}}$$

$$\frac{1 - \sqrt{3}}{4\sqrt{2}}, \frac{-3 + \sqrt{3}}{4\sqrt{2}}, \frac{3 + \sqrt{3}}{4\sqrt{2}}, \frac{-1 - \sqrt{3}}{4\sqrt{2}}$$

Los primeros cuatro generan los valores del espacio de baja frecuencia, mientras que los últimos cuatro, los valores del espacio de alta frecuencia.

Para obtener la imagen de dimensión reducida, los filtros anteriores se aplican a cada una de las filas de la imagen y posteriormente a las columnas. Después de aplicar tres niveles del proceso de multirresolución se obtienen imágenes reducidas de 32 x 32 píxeles, lo cual se aprovecha para que el procesamiento computacional de la red neuronal sea liviano. La figura 8 muestra las imágenes resultantes.

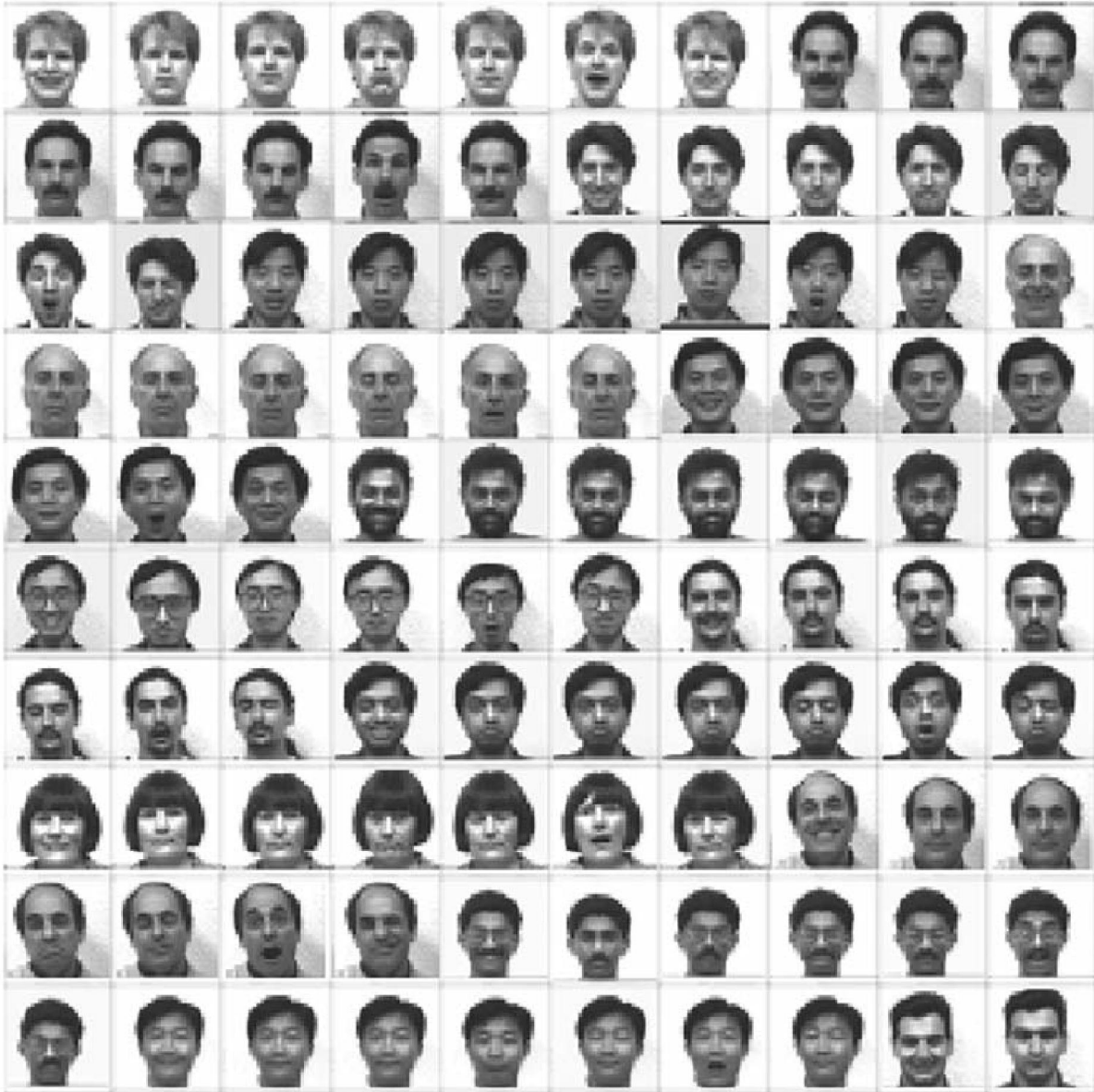


Figura 8
Imágenes de rostros a las cuales se les aplicó la transformada *wavelet*.

Luego de obtener los resultados de las *wavelets* se procedió a la binarización de las imágenes, para lo cual se aplicó un *threshold* con valor de 65. Este valor se obtuvo mediante un proceso empírico, ya que en gran medida está en función de las imágenes con las que se trabaja.

El procedimiento de binarización se realiza para todas las imágenes que se desea reconocer. La normalización de las transformaciones geométricas se realizó después del proceso de binarización, puesto que es necesario para el funcionamiento adecuado de la red neuronal. La figura 9 presenta los resultados de estas dos etapas aplicadas a las imágenes de la figura 8. Las imágenes de la base de datos de la Universidad de Yale no tienen mucha variabilidad en cuanto a las transformaciones geométricas: las que existen son mínimas.

Después de la descomposición *wavelet* debe realizarse una binarización de la imagen para luego aplicar el proceso de normalización; estos resultados se pueden ver en la figura 9.

































Figura 9

Resultado de la binarización y normalización de algunas imágenes de la base de datos.

En la tabla 1 se muestran los resultados del modelo, obtenidos después del procesamiento de 105 imágenes de la base de datos Yale, correspondientes a 15 personas con gestos faciales diferentes.

Tabla 1
Resultados obtenidos a partir del modelo propuesto

Nro.	Persona	Nro. de gestos	% Clasificación			Prototipo
			correcta	errada	%	
1		7	7	0	100	
2		7	7	0	100	
3		7	7	0	100	
4		7	6	1	85	
5		7	7	0	100	
6		7	7	0	100	
7		7	7	0	100	
8		7	6	1	85	
9		7	7	0	100	
10		7	7	0	100	
11		7	7	0	100	
12		7	7	0	100	
13		7	7	0	100	
14		7	6	1	85	
15		7	7	0	100	

Discusión

En la actualidad se vienen utilizando diversas técnicas para la clasificación de patrones. La ventaja que proporciona la utilización de la red caótica es que ofrece la posibilidad de reconstruir la imagen, lo cual es importante en cualquier sistema de reconocimiento. La figura 11a muestra el caso de reconstrucción de una imagen.

Otra ventaja del modelo propuesto es la cantidad de patrones que permite que la red aprenda. Por ejemplo, cuando se usa una red de Hopfield, la red puede aprender a lo sumo tres patrones, lo cual limita sus posibilidades, pues si el individuo que se ha de reconocer no está dentro de los tres más parecidos al prototipo, se cometerá un error. La red caótica, en cambio, consigue almacenar un mayor número de rostros, en promedio 15.

Otra área que demuestra la mayor eficacia del modelo propuesto frente a otros es que necesita pocas iteraciones para lograr la convergencia. En la figura 11b se puede ver, por ejemplo, que el número de iteraciones es menor en comparación con Hopfield.

No se hizo una comparación del método propuesto con otros, ya que Zhang demostró que el uso de *wavelet* y memorias asociativas tiene ventajas frente a modelos tradicionales como el PCA (Zhang, Zhang, y Ge, 2004); Zhang presenta cuadros estadísticos del porcentaje de reconocimiento que tiene la combinación de estas técnicas frente a otras ya consolidadas (Zhang, et al., 2004).



Figura 10

Ejemplo de imágenes que el modelo propuesto consiguió aprender a pesar de sus distorsiones.

Otro aspecto importante en el desarrollo de este método es su alta tolerancia a errores. En la figura 10 se puede ver que en el caso de uno de los rostros binarizados, tan sólo la cuarta parte del rostro aparece, y a pesar de esta limitación la red consigue reconocerlo; el mismo resultado se obtuvo con las binarizaciones del mismo rostro que presentaban dificultades similares.

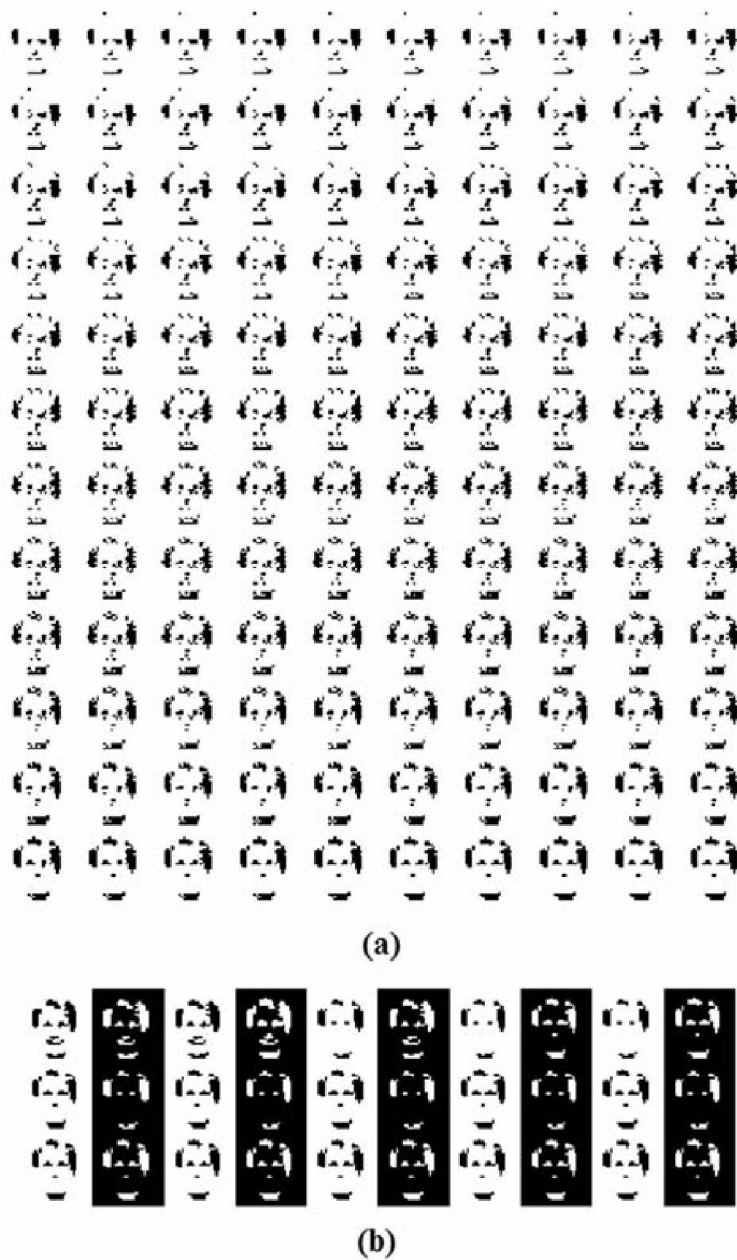


Figura 11
 Reconstrucción de la imagen parcial de un rostro:
 (a) Hopfield; (b) red caótica.

Aly comparó algunas técnicas conocidas de reconocimiento de rostros (Aly, 2006). Debido a que utilizó la misma base de rostros empleada en este trabajo, se pudo hacer una comparación entre las diferentes técnicas analizadas —específicamente *eingefaces*, *fisherfaces* y SIFT (Scale Invariant Feature Transform)— y el modelo propuesto. En el cuadro 1 se resumen los resultados de la comparación, en la que puede apreciarse que el modelo propuesto es más robusto debido a su mayor tasa de aciertos.

Cuadro 1
Comparación del modelo propuesto con otras técnicas

Técnica	% Aciertos	% Error
<i>Eigenface</i>	72,1	17,9
<i>FisherFace</i>	86,9	13,1
SIFT	91,7	8,3
Modelo propuesto	97,1	2,9

La transformada *wavelet* es utilizada en este trabajo para la reducción de la dimensión de una imagen, aspecto muy importante para poder almacenar imágenes de grandes dimensiones de forma que no ocupen mucho espacio en bases de datos, lo cual mejora la eficacia de cualquier sistema de bases de datos multimedia. En todo sistema real de reconocimiento automático de imágenes se tienen problemas con las transformaciones geométricas que puedan sufrir (traslación, rotación y escala). La aplicación de momentos se utiliza para normalizar las imágenes y hacer al modelo propuesto más robusto, ya que la imagen de un rostro no siempre está en la misma posición en una foto.

Las redes neuronales son muy utilizadas principalmente como clasificadores, Una propiedad importante de las redes caóticas es su capacidad de poder reconstruir ciertas partes de una imagen y su alto grado de tolerancia a ruido, lo que hace que el modelo propuesto acepte imágenes con errores.

Tanto la extracción de características como la clasificación son temas bastante tratados. Las técnicas usadas en este trabajo pueden ser aplicadas a otro tipo de imágenes o a otras áreas, lo que es importante para el avance en el desarrollo de estos métodos. La simple utilización de la imagen de baja frecuencia implica dejar de lado las sub-bandas de alta frecuencia donde se almacenan los detalles de la imagen que pueden ser aprovechados para otras técnicas de reconocimiento. Otros tipos de redes neuronales también se pueden aplicar para la solución de este problema. Todo esto forma parte de posibles trabajos futuros de esta investigación.

Referencias

- Aly, M. (2006). Face recognition using SIFT features. *CNS/Bi/EE Report, 186*.
- Becker, B., y Ortiz, E. (2008). Evaluation of face recognition techniques for application to Facebook. *8th IEEE International Conference on Automatic Face and Gesture Recognition, 2008. FG '08* (pp. 1-6).
- Bianchi, M. F. (2006). *Extração de características de imagens de faces humanas através de wavelets, PCA e IMPCA*. Universidade de São Paulo.
- Campos, R., De Campos, T., y Cesar, R. (2000). Detection and tracking of facial features in video sequences. *Lecture Notes in Artificial Intelligence* (1793), 197-206.
- Cao, Z., Yin, Q., Sun, J., y Tang, X. (2010). Face recognition with learning-based descriptor.
- Chang, Y., Hu, C., Feris, R., y Turk, M. (2006). Manifold based analysis of facial expression. *Image and Vision Computing, 24*(6), 605-614.
- Chellappa, R., Wilson, C., y Sirohey, S. (1995). Human and machine recognition of faces: a survey. *Proceedings of the IEEE, 83*(5), 605-614.
- Daubechies, I. (1990). The wavelet transform, time requency localization and signal analysis. *IEEE Transactions on Information Theory, 36*(5), 961-1005.
- Daubechies, I. (1992). *Ten lectures on Wavelets*: CBMS-NSF.
- Gabor, D. (1946). Theory of communication. Part 1: The analysis of information. *Electrical Engineers-Part III: Radio and Communication Engineering, Journal of the Institution, 93*(26), 429-441.
- Gomes, Brancalion, y D., F. (2007). Reconhecimento automático de alvos em imagens SAR utilizando-se *wavelets* ea classificação pela mínima distância e pelo MAP.
- Gomez, J., Velho, L., y Goldenstein, S. (1997). *Wavelets: teoria, software e aplicações*: Instituto de Matemática Pura e Aplicada.
- Grgic, S., Grgic, M., y Zovko-Cihlar, B. (2001). Performance analysis of image compression using wavelets. *IEEE Transactions on Industrial electronics, 48*(3), 682-695.
- Grossmann, A., y Morlet, J. (1984). Decomposition of hardy function into square integrable wavelets of constant shape. *SIAM Journal of Mathematical Analysis, 15*(4), 723-736.
- Haddadnia, J., Faez, K., y Moallem, P. (2001). Neural network based face recognition with moment invariants. *IEEE International Conference on Acoustics, Speech, and Signal Processing, 1*, 1018-1021.

- Hertz, J., Krogh, A., y Palmer, R. G. (1991). *Introduction to the theory of neurocomputing*. Addison-Wesley.
- Hu, C., Harguess, J., y Aggarwal, J. (2009). Patch-based face recognition. Tomado del video *International Conference on Image Processing (ICIP)* (pp. 1-4).
- Intrator, N., Reisfeld, D., y Yeshurun, Y. (1996). Face recognition using a hybrid supervised/unsupervised neural network. *Pattern Recognition Letters*, 17, 67-76.
- Jeng, S., Liao, H., Han, C., Chern, M., y Liu, Y. (1998). Facial feature detection using geometrical face model: an efficient approach. *Pattern Recognition*, 31(3), 273-282.
- Jin, Z., Lou, Z., Yang, J., y Sun, Q. (2007). Face detection using template matching and skin-color information. *Neurocomputing*, 70(4-6), 794-800.
- Kondo, T., y Yan, H. (1999). Automatic human face detection and recognition under non-uniform illumination. *Pattern Recognition*, 32, 1707-1718.
- Lai, J., Yuen, P. C., y Feng, G. (2001). Face recognition using holistic Fourier invariant features. *Pattern Recognition*, 34(1), 95-109.
- Lam, K., y Yan, H. (1996). Locating and extracting the eye in human face images. *Pattern Recognition*, 29(5), 771-779.
- Liu, C. (2006). Capitalize on dimensionality increasing techniques for improving face recognition grand challenge performance. *IEEE transactions on pattern analysis and machine intelligence*, 725-737.
- Mallat, E. G. (1989). A theory for multiresolution signal decomposition: the wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(7), 674-693.
- Misiti, M., Oppenheim, G., Poggi, J.-M., y Misiti, Y. (2001). *Wavelet Toolbox Documentation*.
- Oliveira, H. (2007). *Deteção automática de alvos em imagens*. PhD. Tesis de maestría, Instituto Tecnológico de Aeronáutica, São José dos Campos.
- Pinto, N., DiCarlo, J. J., y Cox, D. D. (2009). How far can you get with a modern face recognition test set using only simple features? *Proceedings of Computer Vision and Pattern Recognition 2009*: IEEE.
- Pordeus. (2006). *Reconhecimento automático de alvos em imagens SAR. 2006*. Tesis de maestría. Instituto Tecnológico de Aeronáutica, São José dos Campos.
- Ravichandran, A., y Yegnanarayana, B. (1991). A two-stage neural network for translation, rotation and size-invariant visual pattern recognition. *IEEE International Conference on Acoustics, Speech, and Signal Processing*, 4, 2393-2396.

- Samal, A., y Lyengar, P. (1992). Automatic recognition and analysis of human faces and facial expressions: a survey. *Pattern Recognition*, 25(1), 65-77.
- Sirovich, L., y Kirby, M. (1987). Low-dimensional procedure for the characterization of human faces. *J. Opt. Soc. Am. A.*, 4(3), 519-524.
- Stollnitz, E. J., Deroose, T. D., y Salesin, D. H. (1996). *Wavelets for computer graphics, theory and applications*: Morgan Kaufmann Publisher.
- Yuille, A., Hallinan, P., y Cohen, D. (1996). Feature extraction from faces using deformable templates. *International J. Comput. Vision*, 8(2), 99-111.
- Zhang, B. L., Zhang, H., y Ge, S. S. (2004). Face recognition by applying wavelet sub-band representation and kernel associative memory. *IEEE Transactions on Neural Networks*, 15(1).
- Zhao, L., Gutierrez Caceres, J. C., Damiance, J., y Szui, H. (2003). Chaotic Associative Recalls for Fixed Point Attractor Patterns. *International Joint Conference Neural Networks*.
- Zhao, L., Gutierrez Caceres, J. C., Damiance, J., y Szui, H. (2006). Chaotic dynamics for multi-value content addressable memory. *Neurocomputing*, 69(13-15), 1628-1636.
- Zhao, W., Chellappa, R., Phillips, P. J., y Rosenfeld, A. (2003). Face recognition: A literature survey. *ACM Computing Surveys (CSUR)*, 35(4), 399-458.