



Computación y Sistemas

ISSN: 1405-5546

computacion-y-sistemas@cic.ipn.mx

Instituto Politécnico Nacional

México

Ramírez Alonso, Graciela María de Jesús; Chacón Murguía, Mario Ignacio
Clasificación de Defectos en Madera utilizando Redes Neuronales Artificiales
Computación y Sistemas, vol. 9, núm. 1, julio-septiembre, 2005, pp. 17-27
Instituto Politécnico Nacional
Distrito Federal, México

Disponible en: <http://www.redalyc.org/articulo.oa?id=61590102>

- Cómo citar el artículo
- Número completo
- Más información del artículo
- Página de la revista en redalyc.org

redalyc.org

Sistema de Información Científica
Red de Revistas Científicas de América Latina, el Caribe, España y Portugal
Proyecto académico sin fines de lucro, desarrollado bajo la iniciativa de acceso abierto

Clasificación de Defectos en Madera utilizando Redes Neurales Artificiales *Wood Defects Classification Using Artificial Neural Network*

Graciela María de Jesús Ramírez Alonso y Mario Ignacio Chacón Murguía

Instituto Tecnológico de Chihuahua
Laboratorio de DSP y Visión
Av. Tecnológico 2909, Chihuahua, Chih.,
México C.P. 31310
Tel.4-13-74-74 Ext 112 y 114
{gramirez,mchacon}@itchihuahua.edu.mx

Artículo recibido en junio 06, 2004; aceptado en abril 01, 2005

Resumen

Este artículo describe un clasificador neural que diferencia entre 7 tipos de defectos en maderas llamados botones. La inspección visual de estos defectos por humanos tiene un alto grado de complejidad debido a la varianza intraclase. Las características utilizadas se extrajeron de las imágenes de maderas mediante filtros Gabor de 2D. Estos filtros son pasa banda selectivos a la orientación y frecuencia, muy utilizados para imágenes en donde la textura es un factor importante. Para optimizar las características se realizó una reducción de dimensión del resultado de los filtros Gabor mediante el método de Análisis de Componentes Principales. La red neural que se implementó fue una red Perceptrón multicapa de 3 capas entrenada con el algoritmo de *Resalient Backpropagation*. La tasa de reconocimiento de la red fue de un 83.91%, siendo este resultado aceptable teniendo en cuenta que un inspector humano alcanza un reconocimiento entre el 75 y 85%.

Palabras Clave: Redes neurales, filtros Gabor, procesamiento de imágenes.

Abstract

This paper describes a neural classifier to classify 7 different wood defects called knots. Human visual inspection of these defects involves a high degree of complexity due to inter-class variance. 2D Gabor filters were used for feature extraction. These filters are selective band pass filters to orientation and frequency. These filters are used where texture is an important feature. The method of principal component analysis was used to reduce the number of features generated by the Gabor filters. The neural network implemented was a multilyer perceptron with 3 layers trained with the Resalient backpropagation algorithm. The performance of the classifier was 83.91% of correct classification. This result is acceptable considering that the performance of a human inspector is 75% to 85%.

Keywords: Neural networks, Gabor filters, image processing

1 Introducción

Las Redes Neurales Artificiales, RNA, están compuestas de un gran número de elementos de procesamiento altamente interconectados, Neuronas, trabajando al mismo tiempo para la solución de problemas específicos. Actualmente las RNA están siendo utilizadas en un gran número de áreas debido a que tienen la capacidad de aprender y recordar de ejemplos. Algunas de estas áreas son: mercado bursátil, meteorología, robótica, minería, y reconocimiento de patrones, entre otras [1], [2], [3].

Los tipos de redes neurales se caracterizan por su topología, número de capas, número de neuronas, la manera en que se conectan las neuronas en cada capa, por su tipo de entrenamiento o regla de aprendizaje y manera en que se adaptan los pesos en la red. Existen dos clases de entrenamiento, el supervisado y el no supervisado. En el supervisado se presentan a la red tanto los patrones de entrada como la salida que se desea para esos patrones.

En el no supervisado solo se presentan a la red los patrones de entrada y la red los agrupa de acuerdo a una medida de similitud entre ellos.

Uno de los tipos de redes más utilizados para el área de reconocimiento de patrones es la Red Perceptrón Multicapa, MLP, con el entrenamiento supervisado de Retropropagación, BP [2]. Este tipo de red es un aproximador universal de funciones y se utiliza en este trabajo para clasificar diferentes tipos de defectos en maderas.

La inspección visual de defectos en madera es una tarea muy difícil ya que las diferentes variaciones que existen en los defectos incorpora un alto grado de subjetividad en la clasificación del defecto, haciendo pensar a una persona que un defecto pertenece a una clase mientras que otra persona la clasifica dentro de otra clase. Un inspector raramente alcanza un desempeño mejor del 85% en reconocer, localizar e identificar defectos en madera [4]. Según estudios realizados muestran

que el clasificar un defecto en 4 categorías solo el 60% de los defectos fueron asignados con el mismo grado por dos diferentes inspectores [4]. La importancia de la correcta clasificación de los defectos incide en la determinación de la calidad de la madera y a su vez la calidad incide en el precio.

En este trabajo se clasifican 7 tipos de defectos denominados “botones”. Se tienen un total de 438 imágenes las cuales han sido clasificadas por expertos. Tanto las imágenes como su clasificación fueron adquiridas de un experimento ya realizado por la Universidad de Oulu Finlandia [5]. Anteriormente se han hecho otros trabajos que clasifican estas mismas clases de defectos pero utilizando diferentes técnicas para extracción de características o diferentes tipos de redes [6], [7], [8], [9]. En este trabajo se trata de simplificar las técnicas para la extracción de características, así como el uso de una red neural fácil de entrenar pero al mismo tiempo disminuyendo el número total de errores en la clasificación.

Este artículo está organizado de la siguiente manera. En la sección 2 se explica cuales son los defectos en maderas que se analizan en el experimento, en la sección 3 se describe el proceso de extracción de características en las imágenes para formar el vector de entrada de la red. En la sección 4 se mencionan algunas consideraciones que se deben de seguir en el diseño de una red neural para obtener mejores resultados en la clasificación. En la sección 5 se expone el desarrollo del experimento. En la sección 6 se presentan los resultados y finalmente la sección 7 muestra las conclusiones finales del trabajo.

2 Tipos de ‘Botones’ en Maderas

Los botones son la categoría de defectos más común que se encuentra durante la inspección de maderas. En el presente trabajo se clasifican 7 clases de botones, los cuales se identificarán por sus siglas en ingles. **DR** se refiere al defecto llamado *dry*, **EN** al defecto *encased*, **SO** al defecto *sound*, **LE** al defecto *leaf*, **ED** al defecto *edge*, **HO** al defecto *horn* y **DE** al defecto *decayed*.

Los botones **DR** son aquellos que están rodeados por un círculo oscuro muy estrecho, los botones **EN** están rodeados por una región oscura más grande que los botones DR. El botón **LE** tiene forma de hoja, éste puede confundirse con el **HO** ya que cuentan con la misma forma, la diferencia es que el HO está localizado en la orilla de la madera. Otra categoría son los botones **ED** los cuales son parecidos a los DR pero estos se encuentran en la orilla de la madera. Los defectos denominados **DE** son aquellos que tienen el grado más severo de descomposición. En cambio los defectos **SO** tienen poco grado de severidad pero existen variaciones en su forma, algunos tiene pequeñas rasgaduras dentro del botón, otros pueden tener un poco de porosidad y los menos leves apenas se alcanza a notar la forma del botón [6], [7], [8], [9]. La Fig. 1 muestra un ejemplo de cada uno de los defectos antes mencionados.

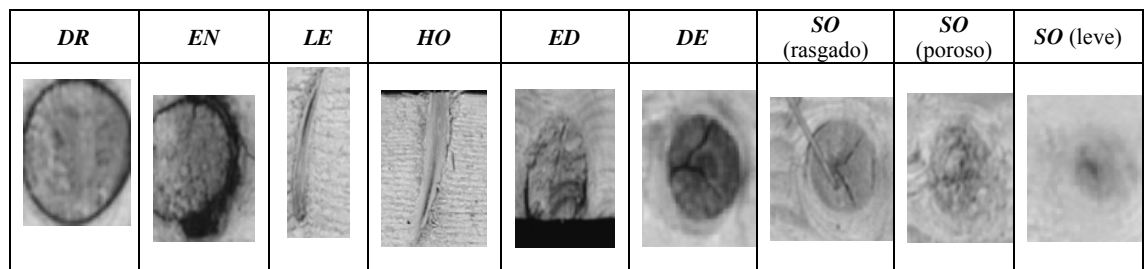


Fig. 1. Tipos de botones en maderas

La importancia de la correcta clasificación de los defectos incide en la determinación de la calidad de la madera y a su vez la calidad incide en el precio. Un inspector reconoce los tipos de botones basado en sus características de forma, tamaño, estructura y color. En maquinas de inspección visual el mayor problema de la clasificación de tipos de botones se tiene en la selección de características. Comúnmente esta clasificación se basa en características geométricas de las imágenes, como tamaño del defecto, promedio de nivel de gris, tamaño del mínimo cuadrado que encierra el defecto, diámetro y la posición del defecto en la madera [8]. Sin embargo, la información que se basa en estas características no es suficiente para separar todas las clases de botones ya que algunos defectos son muy similares entre si y es muy difícil separarlos si se cuenta solamente con esta información. Además, las características geométricas requieren de la segmentación de los defectos lo cual resulta difícil para algunos tipos de clases [8]. Una buena opción para analizar las imágenes y extraer características invariantes a rotación y traslación es utilizar filtros Gabor. Los filtros Gabor se pueden describir como un modelo computacional que pretende simular la forma en la que son analizadas las imágenes por la población de células de la corteza

visual [10]. Lo que se realiza en los filtros Gabor de 2-D es lo siguiente: se diseñan varios filtros pasa banda (banco de filtros) los cuales responden a diferentes frecuencias y orientaciones definidas por las características de la onda Gaussiana (ancho y orientación de la onda), la imagen del defecto será filtrada con cada uno de estos filtros resultando en varias imágenes filtradas las cuales guardaran información relacionada a la frecuencia y orientación a la que se encuentra el filtro.

3 Extracción de Características

3.1 Filtros Gabor

Los filtros Gabor son utilizados para el análisis de imágenes debido a su relevancia biológica y propiedades computacionales [11]. Además, se pueden seleccionar orientaciones y frecuencias específicas que se pueden localizar fácilmente en los dominios del espacio y la frecuencia.

Los filtros Gabor de 2D son filtros pasa banda selectivos a la orientación y la frecuencia haciéndolos confiables para la extracción de características en imágenes. Las formas de los kernels Gabor son sinusoidales espaciales localizada por una ventana Gaussiana. Los filtros constituyen una familia o banco de filtros donde cada uno de los filtros esta dilatado, trasladado y rotado con respecto a otro. Los filtros pueden ser definidos como sigue:

$$\psi(f, \theta, x, y) = \exp(i(f_x x + f_y y) - \frac{f^2(x^2 + y^2)}{2\sigma^2}) \quad (1)$$

donde

$$f_x = f \cos \theta, \quad f_y = f \sin \theta \quad y \quad i = \sqrt{-1}$$

x y y son las coordenadas de los píxeles de la imagen en el rango $(-x/2, x/2)$ y $(-y/2, y/2)$. La frecuencia central del pasa banda es f , la orientación espacial es θ , y el parámetro σ determina el ancho de banda del filtro. En este trabajo el valor de σ es 1.5π .

Teniendo cuidado de que las frecuencias de los filtros no fueran mayor de π se determinaron los siguientes valores para f_k :

$$f_k = \frac{\pi}{\sqrt{2}^k} \quad k = 2, 3, 4 \text{ y } 5 \quad (2)$$

Las orientaciones que se utilizaron fueron las siguientes:

$$\theta_t = \frac{\pi t}{8}, \quad t = 0, \dots, 7 \quad (3)$$

donde k determina la banda de frecuencia (resolución de los detalles extraídos por el filtro) y t enumera las orientaciones.

La representación de la transformada Gabor de una imagen, es la convolución de la imagen con el banco de filtros Gabor de la ecuación 1. Aplicando el teorema de convolución, la salida del filtro Gabor de la imagen por medio de la transformada de Fourier se define de la siguiente manera

$$\mathfrak{F}\{\theta_{f,\theta}(x, y)\} = \mathfrak{F}\{I(x, y)\} \mathfrak{F}\{\psi_{f,\theta}(x, y)\} \quad (4)$$

y

$$\theta_{f,\theta}(x, y) = \mathfrak{F}^{-1} \{ \mathfrak{F} \{ I(x, y) \} \mathfrak{F} \{ \psi_{f,\theta}(x, y) \} \} \quad (5)$$

donde $\theta_{f,\theta}(x, y)$ es la imagen filtrada, $I(x, y)$ es la imagen original, $\psi_{f,\theta}(x, y)$ es la onduleta Gabor a utilizar donde f y θ indican la frecuencia y la orientación a la que se esta trabajando, \mathfrak{F} y \mathfrak{F}^{-1} indica la transformada de Fourier y la transformada inversa de Fourier respectivamente.

En el desarrollo de este trabajo solo se utiliza la magnitud de los filtros (no la fase) ya que se ha determinado que es robusta y tolerante a distorsión para el reconocimiento de patrones [8].

La transformación de Gabor fue implementada en el dominio de la frecuencia. Todas las imágenes de botones fueron normalizados a un tamaño de 64x64, para acelerar el procesamiento de los defectos.

En este trabajo se manejan 8 orientaciones y 4 frecuencias formando un banco de 32 (8x4) filtros de Gabor. Cuando se convoluciona una imagen con este banco de filtros se producen 32 imágenes filtradas cada una con diferentes características a diferentes frecuencias y orientaciones. Cada imagen filtrada es de tamaño de 64x64 píxeles. Para agrupar en una sola matriz toda la información de las 32 imágenes filtradas se realizo lo siguiente:

Desde $\theta=1$ hasta 8 (indica número de orientaciones que se manejan)

Desde $f=1$ hasta 4 (número de frecuencias)

Se toma la salida del filtro $O_{f\theta}$ a frecuencia f orientación θ , la cual tiene un tamaño de 64x64

Se concatenan las columnas de la imagen $O_{f\theta}$ formando una matriz de 4096x1

Se submuestra la matriz de 4096x1 por un factor de 4 formando ahora una matriz de 1024x1

Se guarda ese resultado

$f=f+1$
 $\theta=\theta+1$.

Realizando los pasos anteriores por cada imagen original se obtiene una matriz de 1024x32 la cual guarda diferentes características a 4 diferentes frecuencias y 8 orientaciones. En el siguiente paso se calcula la media y desviación estándar de cada una de las 32 columnas formando una matriz de 2x32. Esta matriz de dos dimensiones se reajusta a un tamaño de 1x64 en la cual los primeras 32 elementos indican la desviación estándar y los siguientes 32 la media. Además se añadieron dos características más, la desviación estándar y media de la imagen original, formando un vector de características de tamaño 1x66. La Fig. 2 muestra un esquema de lo que se explico anteriormente. Todas las características que se utilizaron se extrajeron de imágenes en escala de grises. La razón por la cual se prefiere utilizar características en niveles de gris es para reducir los costos de implementación. Una cámara que obtenga imágenes a color es más costosa que una que obtiene imágenes en niveles de gris y ya que este trabajo forma parte de un proyecto de inspección de madera se pensó en reducir lo más posible los costos de implementación.

Hasta esta etapa, el vector de características de las imágenes es de una dimensión muy grande por lo que se decidió realizar una reducción en la dimensión del vector de características mediante el Análisis de Componentes Principales (PCA).

3.2 Análisis de Componentes Principales

El objetivo del Análisis de Componentes Principales, ACP, es transformar el espacio de representación X en nuevo espacio Y de menor dimensión, en el que los datos no estén correlacionados.

La matriz de covarianza en ese espacio será diagonal. Con el método se trata de encontrar el nuevo conjunto de ejes ortogonales en el que la varianza de los datos sea máxima y el objetivo final es reducir la dimensionalidad del problema.

Los pasos que se siguieron en ACP son los siguientes:

1. Se normalizan los vectores de entrada X haciendo que su media sea cero y su desviación estándar sea igual a 1.
2. Se calcula la matriz de covarianza de los vectores ya normalizados Σ_X . En este paso se utilizan todos los vectores de entrada.
3. Se calculan los valores característicos de Σ_X los cuales se definirán como $\lambda_1, \lambda_2, \dots, \lambda_d$, donde $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d$.
4. Se calculan los vectores característicos $\Phi_1, \Phi_2, \dots, \Phi_d$ asociados a $\lambda_1, \lambda_2, \dots, \lambda_d$
5. Se forma la matriz de transformación $W = [\Phi_1, \Phi_2, \dots, \Phi_d]$
6. Se calculan los nuevos vectores Y con la siguiente ecuación

$$Y = W^T X \quad (6)$$

Donde T indica la transpuesta de la matriz W , y Y es la matriz que contiene los nuevos vectores disminuidos en dimensión.

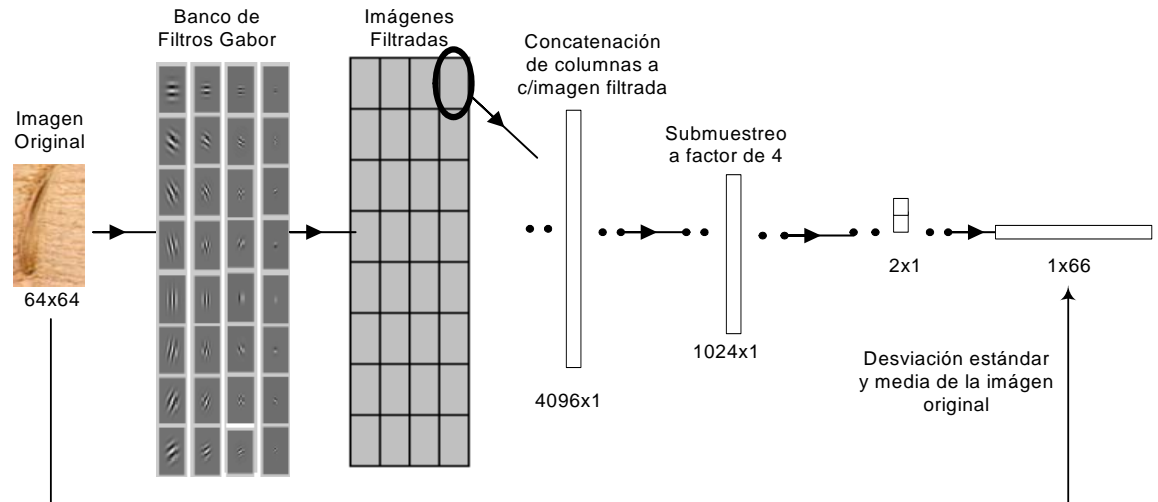


Fig. 2. Diagrama a bloques de la etapa de extracción de características de la imagen

4 Redes Neuronales Artificiales

Uno de los tipos de redes más utilizado en reconocimiento de patrones es la de Red Perceptrón Multicapa (RPM) entrenada con el algoritmo de retropropagación, RP.

Los problemas que se tiene con este tipo de redes es definir el número de capas en la red, el número de neuronas en cada una de las capas y las funciones de activación de las neuronas. Haciendo un estudio de varios métodos de reconocimiento de patrones que utilizan RNA [2], [12], [13] se llegaron a las siguientes conclusiones:

- Se puede llegar a un buen resultado de clasificación si se utilizan 3 capas en la red.
- Las funciones de activación de las neuronas son comúnmente la función sigmoideal logarítmica y la sigmoideal tangencial.
- Cuando la función de activación de la capa de salida es la sigmoideal logarítmica se sugiere que los valores de la salida deseada sean de 0.1 y 0.9 para que no trabaje en la zona saturación de la función de activación. Cuando la función de activación es sigmoideal tangencial se recomienda que los valores de salida deseada sean de -0.8 y 0.8 para evitar saturación.
- Se recomienda no introducir a la red todos los vectores que correspondan a una misma clase juntos. Es mejor que las entradas varíen de acuerdo a la clase que representan para lograr que la red obtenga una buena generalización en la clasificación.
- Comúnmente el número de neuronas en la capa de entrada de la red corresponde al número de características que se extrajeron de la imagen.
- El número de neuronas en la capa de salida corresponde al número de clases.
- El número de neuronas en la capa intermedia se puede determinar por dos métodos [12], [13].
 1. El número de neuronas es igual al doble de neuronas que se tienen en la capa de salida más uno.
 2. El número de neuronas es igual al número de neuronas de la capa de salida multiplicado por 10.

La red con la que se trabajó es una RPM de 3 capas entrenada con una modificación al algoritmo de RP llamado *Resilient Backpropagation*. El número de neuronas en cada capa fue de 10-19-9 con funciones de activación sigmoideal tangencial, en

las primeras dos capas, y sigmoideal logarítmica para la última capa. El algoritmo de *Resilient Backpropagation* toma el signo de la derivada del error con respecto al peso de la neurona para determinar la dirección de la actualización de pesos. La magnitud de la derivada no tiene efecto en la actualización de pesos. El cambio de los pesos se determina de acuerdo a un factor incremental o decremental. Si la derivada del error con respecto al peso tiene el mismo signo en dos iteraciones sucesivas el valor de la actualización de pesos se incrementará por un factor Δinc . De otra manera, si la derivada del error con respecto al peso cambia de signo con relación al valor de la iteración anterior el cambio de pesos se decrementará por un factor Δdec . Si la derivada es cero el valor para la actualización de pesos será el mismo. Si los pesos se encuentran oscilando el cambio de pesos se reducirá. Si la actualización de pesos se realiza en una misma dirección por varias iteraciones la magnitud del cambio de pesos se incrementará. El valor que se utilizó para Δinc es igual a 1.2 y para Δdec es 0.5. La fórmula para la actualización de pesos queda definida de la siguiente manera:

$$\begin{aligned} \text{Si } \left(\frac{\partial E}{\partial w_{ij}}(t-1) * \frac{\partial E}{\partial w_{ij}}(t) > 0 \right) \text{ entonces } & \begin{cases} \Delta w_{ij}(t) = \Delta w_{ij}(t-1) + \Delta inc \\ w_{ij}(t) = w_{ij}(t-1) + \Delta w_{ij} \end{cases} \\ \text{Si } \left(\frac{\partial E}{\partial w_{ij}}(t-1) * \frac{\partial E}{\partial w_{ij}}(t) < 0 \right) \text{ entonces } & \begin{cases} \Delta w_{ij}(t) = \Delta w_{ij}(t-1) + \Delta dec \\ w_{ij}(t) = w_{ij}(t-1) + \Delta w_{ij} \end{cases} \\ \text{Si } \left(\frac{\partial E}{\partial w_{ij}}(t-1) * \frac{\partial E}{\partial w_{ij}}(t) = 0 \right) \text{ entonces } & \begin{cases} \Delta w_{ij}(t) = \Delta w_{ij}(t-1) \\ w_{ij}(t) = w_{ij}(t-1) + \Delta w_{ij} \end{cases} \end{aligned} \quad (7)$$

Donde w_{ij} representa los pesos de conexión entre la neurona j y la neurona i , Δw_{ij} indica la modificación que se le hará a ese peso, E es el error de la red y $\frac{\partial E}{\partial w_{ij}}(t)$ es la derivada del error con respecto al peso en un tiempo t . Riedmiller y Braun [14] recomiendan que el valor inicial para la actualización de pesos sea igual a 0.07.

5 Desarrollo

5.1 Filtros Gabor

Para la extracción de características en la imagen se utilizan filtros Gabor. El proceso es el siguiente. Se diseñan varios filtros pasa banda (banco de filtros) los cuales responden a diferentes frecuencias y orientaciones definidas por las características de la onda Gaussiana (ancho y orientación de la onda). La imagen será filtrada con cada uno de estos filtros resultando en varias imágenes las cuales guardaran información relacionada a la frecuencia y orientación a la que se encuentra el filtro.

Primero se calcula la respuesta de los filtros Gabor utilizando las ecuaciones 1, 2 y 3. La Fig. 3 muestra la parte real de estos cálculos. La magnitud de los filtros Gabor con 4 frecuencias en una sola orientación se aprecia en la Fig. 4.

Teniendo la respuesta de los filtros Gabor, banco de filtros con 4 diferentes frecuencias y 8 orientaciones, éstos se convolucionan con la imagen original formando 32 (4x8) imágenes filtradas de tamaño 64x64. A cada una de estas imágenes filtradas se le concatenan sus columnas formando una matriz de 4096x1. Luego se submuestra esta matriz por un factor de 4 formando una matriz de 1024x1. Esto se realiza en cada una de las 32 imágenes filtradas formando por último una matriz de 1024x32. A cada una de las 32 columnas se le calcula la media y la desviación estándar obteniendo una matriz de 2x32. Luego se concatenaron los renglones formando una matriz de 1x64, siendo las primeras 32 las características de desviación estándar y las siguientes 32 las características de la media. La Fig. 2 muestra estos pasos.

Hasta este momento se tienen 64 (32x2) características por cada imagen a clasificar. Según mencionan Lampinen y Smolander [8] la distribución de los niveles de gris (histograma) en las imágenes originales de los defectos es una buena característica que se pueden añadir al clasificador, por esta razón se decidió utilizar la media y desviación estándar de las imágenes originales como características adicionales teniendo ahora 66 (64+2) características por cada imagen a clasificar.

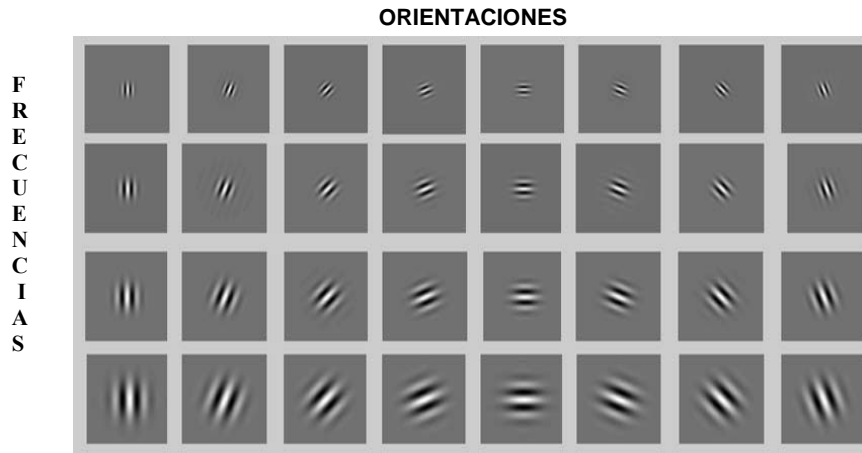


Fig. 3. Parte real de los filtros Gabor con 4 diferentes frecuencias y 8 orientaciones (dominio espacial)

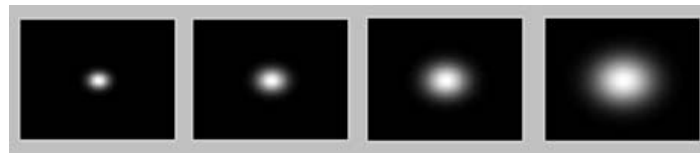


Fig. 4. Magnitudes del filtro con 4 diferentes frecuencias a una sola orientación (dominio de la frecuencia)

5.2 Reducción de la Dimensión del Vector de Características

Como la dimensión del vector de entrada es grande se utiliza el método de Análisis de Componentes Principales (ACP) para disminuir la dimensión del vector. Para esto se siguen los pasos ya descritos en la sección 3.2.

La matriz de covarianza es calculada con todos los vectores de entrada, en este caso se tienen 438 muestras cada una con 66 elementos teniendo así que la dimensión de la matriz de covarianza es de 66×66 . Se calculan los mayores 7 valores característicos de la matriz de covarianza. Con 7 valores característicos se tienen 7 vectores característicos, la dimensión de la matriz de transformación W es de 66×7 . El cálculo de los nuevos vectores de entrada se realiza con la ecuación 6 teniendo ahora una dimensión de 7×1 por cada vector de entrada.

5.3 Clasificación

El número de defectos que se clasifican son 7 pero como se tienen variaciones en la forma del defecto **SO**, este se subdividió en 3 clases, una clase consta de las muestras que tienen rasgaduras dentro del botón, otra clase presenta un poco de porosidad dentro del botón y la ultima es la clase que apenas se nota que forma un botón. Por lo tanto el número de neuronas en la capa de salida es de nueve, en donde tres neuronas representan a la misma clase. El número de neuronas en la capa intermedia se determino por los dos métodos ya descritos en la sección 4. El número de muestras que se tienen para cada defecto se especifican en la tabla 1:

Tabla 1. Numero de muestras disponibles para cada defecto

Nombre del defecto	Num. de muestras disponibles
ED	65
EN	29
LE	47
DR	69
HO	35
SO(leves)	49
SO (porosos)	57
SO (rasgados)	73
DE	14
Total	438

Como se puede observar el número de muestras para cada clase no es uniforme, en la clase **DE** se tienen muy pocas muestras, en cambio en la clase **SO** se tienen bastantes.

Al momento de entrenar la red es importante que se tenga un número homogéneo de muestras que representen a cada clase, esto es para que se pueda llegar a una buena generalización evitando que la red solo pueda representar correctamente aquellas clases que tuvieron más muestras en el entrenamiento. Es por eso que se decidió que el número de muestras para cada clase en el entrenamiento de la red fuera de 25. Algunas clases como la **ED** y la **DR** son muy difíciles de clasificar con solo 25 muestras ya que se confunden fácilmente con otros tipos de defecto, aún para un inspector humano. Así que se decidió que la clase **ED** y **DR** contaran con más de 25 muestras para el entrenamiento de la red. El defecto **DE** solo tiene 14 muestras, para completar las 25 muestras se tuvieron que repetir algunas de ellas. La tabla 2 muestra el número de muestras para cada clase en el entrenamiento y prueba de la red.

Tabla 2. Distribución del número de muestras para el entrenamiento y prueba de la red

Nombre del defecto	Num. muestras entrenamiento	Num. muestras prueba	Total muestras
ED	37	28	65
EN	25	4	29
LE	25	22	47
DR	38	31	69
HO	25	10	35
SO (leves)	25	24	49
SO (porosos)	25	32	57
SO (rasgados)	25	48	73
DE	25 (11 se repiten)	0	14
Total	250	199	438

6 Resultados

La red que se utilizó fue una RPM de 3 capas entrenada con una modificación al algoritmo de RP llamada *Resilient Backpropagation* [14].

El número de neuronas de la capa de salida es igual al número de clases que se quieren clasificar, por lo tanto es igual a 9. Siguiendo la propuesta que hacen Tsoukalas L. H. y Uhrig R. E [12] el número de neuronas de la capa intermedia es igual al doble de las que se tiene en la capa de salida más 1, para este caso es igual a 19. Y el número de neuronas en la capa de entrada es igual a 10, aunque el vector de características está formado por 7 elementos utilizando 10 neuronas en la capa de entrada dio mejores resultados que el utilizar solamente 7.

El número de épocas en el entrenamiento fue de 200. No se observaron mejoras en la clasificación si se utilizaban más de 200 épocas. Las funciones de activación de las capas de entrada e intermedia fue la sigmoideal tangencial y para la capa de salida se utilizó la función sigmoideal logarítmica con valores de salida deseada de 0.1 y 0.9 para evitar llegar a saturación en la red.

El resultado del entrenamiento se muestra en la tabla 3. Esta tabla muestra la matriz de confusión de clasificación de las 250 imágenes que se utilizaron para entrenar la red. La primera columna indica la clase de defecto que se está analizando y la primera fila señala hacia donde fue clasificado el defecto. La penúltima columna indica el número de muestras que se utilizaron en cada clase y la última columna señala el número de errores que se obtuvieron en cada clase. La última fila tiene el número de errores totales que se obtuvieron en el sistema.

La clase **SO** consta de 3 subclases por lo que las muestras que se tienen para el entrenamiento son 75 (25x3) y para la prueba son 179 (49+57+73) muestras.

Tabla 3. Resultados de la clasificación en el entrenamiento de la red

	<i>ED</i>	<i>EN</i>	<i>LE</i>	<i>DR</i>	<i>HO</i>	<i>SO</i>	<i>DE</i>	# muestras	Error
<i>ED</i>	37	0	0	0	0	0	0	37	0
<i>EN</i>	0	19	1	2	0	3	0	25	6
<i>LE</i>	0	0	21	0	0	4	0	25	4
<i>DR</i>	1	0	1	31	0	5	0	38	7
<i>HO</i>	1	0	0	0	24	0	0	25	1
<i>SO</i>	2	1	1	5	0	64	2	75	11
<i>DE</i>	0	1	0	3	0	2	19	25	6
Error Total									35

En la tabla 4 se muestran la matriz de confusión utilizando las 199 muestras para prueba de la red.

Por lo tanto el reconocimiento que se obtiene con las muestras de entrenamiento es del 86% y un reconocimiento del 83.91% utilizando las muestras de prueba. Este resultado se considera aceptable teniendo en cuenta de que un inspector humano alcanza un reconocimiento entre el 75 y 85%. La clase que logró mejores resultados fue la de **ED** ya que solamente 2 de 28 muestras fueron clasificadas erróneamente. En cambio las clases que más errores obtuvieron fueron la **DR** con 9 errores de 31 muestras y la **EN** que clasifico erróneamente la mitad de sus muestras. Estos resultados son comprensibles ya que algunas de las muestras de la clase **DR** son muy parecidas a las de la clase **EN**, **SO** y **DE**. Incluso a simple vista es difícil distinguir cuales características son las que debe de tomar en cuenta un inspector humano para la correcta clasificación de este tipo de defectos.

Table 4. Resultados de la clasificación con los vectores de prueba

	<i>ED</i>	<i>EN</i>	<i>LE</i>	<i>DR</i>	<i>HO</i>	<i>SO</i>	<i>DE</i>	# muestras	Error
<i>ED</i>	26	0	0	0	1	0	1	28	2
<i>EN</i>	0	2	0	0	0	1	1	4	2
<i>LE</i>	0	0	19	1	0	0	2	22	3
<i>DR</i>	1	2	4	22	0	0	2	31	9
<i>HO</i>	2	0	0	0	8	0	0	10	2
<i>SO</i>	0	2	1	3	0	90	8	104	14
Error Total									32

7 Conclusiones

El buen resultado de un clasificador depende en gran medida de sus entradas. Si los objetos que se van a clasificar varían en tamaño, color y orientación dentro de una misma clase se necesita que la etapa de extracción de características en la imagen sea robusta e invariante a la dimensión y orientación del objeto que se quiere analizar.

Los filtros Gabor cuentan con esas características, el problema es que aumenta las dimensiones del vector extraído de la imagen conforme se cambien las orientaciones y frecuencias en el filtro. El método usado en este trabajo para la reducción de dimensiones de espacio fue el de ACP. Una vez que se seleccionaron las mejores características mediante ACP el problema que se resolvió fue la elección de una buena red neural capaz de clasificar entradas muy parecidas en clases diferentes. En este trabajo se estudiaron varias técnicas para el diseño de una red neural. El tipo de red que se implementó

fue una RPM de 3 capas entrenada con una modificación al algoritmo de RP llamada *Resilient Backpropagation*. El número de neuronas en cada capa fue de 10-19-9 con funciones de activación sigmoideal tangencial, en las primeras dos capas, y sigmoideal logarítmica para la última capa.

Analizando de nuevo los resultados de la clasificación nos damos cuenta de que los errores obtenidos en el clasificador se deben a la complejidad natural del problema. Algunas muestras tienen características que caen dentro de dos distintas clases, por lo que es de esperarse que la red cometa errores que un humano también cometería.

La etapa más importante en el diseño un sistema es decidir cuales características serán tomadas como entrada a la red neural, en nuestro caso determinar cuales salidas de los filtros Gabor serán las entradas a la red o el tipo de transformación que debe de sufrir el vector de características para disminuir su dimensionalidad. En este trabajo se utilizó el método de Análisis de Componentes Principales para la transformación del vector de entrada. En trabajos futuros se piensa utilizar otra técnica que realice esta misma tarea esperando obtener mejores resultados en la clasificación.

Dentro de los trabajos futuros podemos comentar que actualmente se están realizando estudios para la clasificación de este tipo de defectos pero ahora utilizando un Sistema de Inferencia Difusa o algoritmos de Agrupamiento Difuso para comparar los resultados con los de la Red Neural propuesta. También se está trasladando el clasificador descrito en este trabajo a un DSP para tener una velocidad de proceso en tiempo real [15].

Agradecimientos

Los autores agradecen a CONACYT por el financiamiento de este trabajo bajo el convenio PFPN-03-29-05.

Referencias

1. **Haykin S.:** Neural Networks A comprehensive foundation. Prentice Hall (1999)
2. **Lin C. y George Lee S.C.:** Neural Fuzzy Systems. A Neuro-Fuzzy Synergism to Intelligent Systems. Prentice Hall (1996)
3. **Kohonen T.:** Self Organizing Maps. Springer-Verlag, Berlin, Germany (1995)
4. **Silven O., Niskanen M. and Kauppinen H.:** Wood inspection with non-supervised clustering. Machine Vision and application. Vol. 13 (2003) 275-285
5. **Silven O., Niskanen M. and Kauppinen H.** University of Oulu. Department of electrical engineering. URL: <http://www.oulu.fi/research/imag/gsom/>
6. **Lampinen J. and Smolander S.:** Word defect recognition: A comparative study. Proc. of workshop on Machine vision in advanced production. Oulu, Finland (1994)
7. **Lampinen J., Smolander S. and Korhonen M.:** Wood surface inspection system based on generic visual features. International Conference on artificial neural networks ICANN'95, Paris, October 9-13(1995)
8. **Lampinen J. and Smolander S.:** Self organizing feature extraction in recognition of wood surface defects and color images. International Journal of Pattern Recognition and Artificial intelligence. Vol. 10, No. 2 (1996) 97-113
9. **Kauppinen H., Silven O. and Piirainen T.:** Self Organizing Map based user interface for visual surface inspection. Proc. 11th Scandinavian conference on image analysis. Kangerlussuaq, Greenland, June 7-11 (1999) 801-808
10. **Aznar-Casanova J. A.:** Análisis multiescala y multiorientación de imágenes mediante un banco de filtros de Gabor-2D. Revista Cognitiva; Vol.12, no.2, 2002, pp 223-246
11. **Liu C. and Wechsler H.:** Independent component analysis of Gabor features for face recognition. IEEE Trans. Neural Networks. Vol. 14 (2003) 919-928
12. **Tsoukalas L. H., Uhrig R. E.:** Fuzzy and Neural Approaches in Engineering, Wiley Interscience (1997)
13. **Somaie A.A., Badr A., Salah, T.:** Aircraft image recognition using back-propagation. Radar, CIE International Conference on, Proceedings, 15-18 Oct (2001) 498-501
14. **Riedmiller M., Braun H.:** A direct adaptive method for faster backpropagation learning: the RPROP algorithm. Neural Networks, IEEE International Conference on. Vol. 1, 28 March - 1 April (1993) 586-591.
15. **Chacón M. M., Valenzuela I.:** Fast image processing application development scheme for the DSK C6711 using Matlab and Simulink. IEEE Workshop on Signal Processing Education, Proceedings, Ago. (2004), 79-83.



Graciela María de Jesús Ramírez Alonso. Recibió el grado de Ingeniero Electrónico y Maestría en Ciencias en Ingeniería Electrónica del Instituto Tecnológico de Chihuahua en 2002 y 2004 respectivamente. Sus intereses de investigación actuales son Redes Neuronales, Lógica Difusa, Procesamiento Digital de Señales, Reconocimiento de Patrones y Visión por Computadora.



Mario Ignacio Chacón Murguía. Recibió el grado de Ingeniero Industrial en Electrónica y el grado de Maestro en Ciencias en Ingeniería Electrónica del Instituto Tecnológico de Chihuahua en 1982 y 1985 respectivamente y el grado de Doctor en Filosofía en Ingeniería Eléctrica de New Mexico State University, EEUU, en 1998. Ha realizado investigaciones para la industria y otras universidades y cuenta con publicaciones internacionales científicas y tecnológicas. Es miembro Senior de la IEEE, miembro de la Pattern Recognition Society, miembro de la IEEE Signal Processing Society y miembro de la IEEE Computational Intelligence Society. Sus intereses de investigación incluyen Visión por Computadora, aplicación de teoría de redes Neuronales y Lógica Difusa a problemas de Procesamiento Digital de Señales, Reconocimiento de Patrones y Diseño de Sistemas Digitales.