

BPNⁿ-CPN. Nuevo método para segmentación de Imágenes basado en Redes Neuronales Artificiales

L. Corbalán¹, W. Hasperué², G. Osella Massa³, L. Lanzarini⁴

Universidad Nacional de La Plata - La Plata, Buenos Aires, Argentina
{corbalan,whasperue,gosella,laural}@lidi.info.unlp.edu.ar

Resumen

La segmentación es un paso muy importante en el procesamiento de imágenes digitales. Su aplicación tiene como objetivo la subdivisión de la imagen en sus partes constituyentes u objetos. Actualmente existen diferentes técnicas para realizarla.

En este trabajo se propone un nuevo método de segmentación de imágenes, denominado BPNⁿ-CPN, basado en Redes Neuronales Artificiales, que combina la utilización de las arquitecturas Backpropagation (BPN) y Contrapropagación (CPN). Si bien las características y capacidades de cada una de estas arquitecturas son ampliamente conocidas, su desempeño individual no siempre es el más adecuado para la resolución de este tipo de problemas.

El método BPNⁿ-CPN aquí propuesto basa su funcionamiento en la combinación de pequeñas redes que realizan el proceso de segmentación con un menor tiempo de entrenamiento logrando una mejora de la performance del sistema. BPNⁿ-CPN ha sido aplicado a la resolución de distintos problemas de segmentación sobre imágenes en tonos de grises con resultados muy satisfactorios demostrando poseer la capacidad de la BPN pero con un tiempo computacional mucho menor.

Finalmente se presentan algunas conclusiones así como algunas líneas de trabajo futuras.

Palabras claves: Segmentación de Imágenes, Regiones, Redes Neuronales, Aprendizaje Supervisado, Backpropagation, Contrapropagación,.

Abstract

Segmentation is a really important stage in digital image processing. Its application aims at subdividing the image into its constituent parts or objects. At present, there exist various techniques to carry this out.

This paper proposes a new image segmenting method, called BPNⁿ-CPN, based on Artificial Neural Networks, which combines the use of Back-propagation (BPN) and Counter-propagation (CPN) architectures. Even though each architecture's characteristics and capacities are widely known, their individual performance is not suitable for solving this kind of problems.

The BPNⁿ-CPN method here posed bases its performance on the combination of small networks which carry out the segmenting process with a shorter training time, thus achieving an improvement in the system performance. BPNⁿ-CPN has been applied to the solution of various segmentation problems over grey scale images with really successful results, proving itself to have the capacity of the BPN but with a much shorter computational time.

Finally, some conclusions and future lines of work are presented. .

Keywords: Image Segmentation, Areas, Neural Networks, Supervised Learning, Back-propagation, Counter-propagation,.

¹ Becario de perfeccionamiento UNLP dirigido por Ing.A.De Giusti- Fac.de Informática - UNLP

² Becario de estudio CIC - Facultad de Informática - UNLP

³ Beca de postgrado Tipo I CONICET dirigido por Ing.A.De Giusti- Fac.de Informática - UNLP

⁴ Profesor Titular DE - Facultad de Informática - UNLP

1 INTRODUCCION

El Procesamiento digital de imágenes ha cobrado gran importancia en los últimos años. En él se distinguen dos ramas fundamentales: el análisis y el mejoramiento de imágenes. El primer paso en el análisis de imágenes consiste generalmente en la segmentación. Su aplicación tiene como objetivo la subdivisión de la imagen en sus partes constituyentes u objetos [7].

Las técnicas de segmentación de imágenes monocromáticas utilizan una de las dos propiedades básicas de los valores del nivel de gris: discontinuidad y similitud. En la primera categoría, los algoritmos dividen la imagen basándose en el cambio brusco de nivel de gris localizando los bordes que existen entre las diferentes regiones de la imagen. Los principales métodos de la segunda categoría consisten en la agrupación de pixels de acuerdo con la similitud de alguna propiedad (intensidad, vecindad, etc.) y se denominan técnicas basadas en regiones. Existen varios métodos para realizar segmentación de imágenes con técnicas basadas en regiones, entre ellos se destacan el crecimiento de regiones [1][7][8][10], la umbralización [2][7][10] y la división y fusión de regiones [7].

También dentro de las técnicas basadas en similitud se puede incluir la segmentación realizada por clasificación de patrones -un pixel junto a su vecindad- con redes neuronales artificiales. Las redes neuronales son capaces de "*aprender*" por medio de un adecuado proceso de entrenamiento aquellas características distintivas de los patrones de cada región de la imagen.

Existen varios trabajos en los que se utilizan redes neuronales para realizar segmentación de imágenes [3][4][5][9]. Generalmente se las combina con alguna otra técnica para mejorar el resultado final. Ejemplo de ello es la utilización de una red Self-Organizing Map (SOM) junto a una técnica tradicional como la morfología matemática [11].

En este trabajo se presenta un nuevo método de segmentación de imágenes que combina la utilización de las redes neuronales Backpropagation (BPN) y Contrapropagación (CPN). Ambas redes presentan aprendizaje del tipo supervisado por lo que necesitan una fase de entrenamiento en la que, con la ayuda de un supervisor, "*aprenden*" a identificar la textura correspondiente al objeto de interés. Una vez concluida esta etapa de aprendizaje, las redes pueden utilizarse como dispositivos especializados en la segmentación del objeto sobre el que se ha realizado el entrenamiento pudiendo identificarlo en otras imágenes nunca presentadas a la red.

Utilizadas individualmente, la BPN suele presentar mejor desempeño en la segmentación requerida, sin embargo, la CPN es más rápida, pudiendo saltarse la etapa de entrenamiento al calcular determinísticamente el estado interno de las conexiones a fin de llevar a cabo la clasificación de manera satisfactoria. La CPN también puede utilizarse como una red auto-organizada dónde la capa competitiva aprende en la etapa de entrenamiento a clasificar patrones de forma no supervisada. Llevado a la segmentación de imágenes esto implica que la CPN es capaz de reconocer automáticamente sin intervención del usuario-supervisor texturas bien diferenciadas en una imagen. Sin embargo en el presente trabajo no se utilizan de esta manera pues se desea obtener redes capaces de identificar sólo determinados objetos de interés. Empleadas de esta forma la segmentación de imágenes tiene innumerables aplicaciones como en el análisis de imágenes de resonancias magnéticas (IRM) [3] en dónde se busca identificar tejidos patológicos como tumores, edemas, etc., o en la visión por computadora dónde puede interesar segmentar la imagen para obtener la ubicación de un objeto que se pretende manipular por medio de un robot. En todas estas aplicaciones se conoce a priori aquello que se desea identificar, por lo tanto el aprendizaje supervisado resulta adecuado.

Si bien las arquitecturas neuronales BPN y CPN son poderosas y muy útiles para identificar texturas en una imagen, este trabajo expone claramente las ventajas de utilizarlas de forma combinada en el

método BPNⁿ-CPN aquí propuesto. En este último se utilizan pequeñas redes que realizan el trabajo de una gran BPN con menor tiempo de entrenamiento y con mejores resultados.

2 REDES NEURONALES ARTIFICIALES

Varios investigadores han desarrollado desde mediados del siglo XX la teoría de las Redes Neuronales Artificiales, las cuales emulan las redes neuronales biológicas, y se han utilizado para aprender estrategias de solución basadas en ejemplos de patrones; estos sistemas no requieren que la tarea a ejecutar se programe, ellos generalizan y aprenden de la experiencia.

2.1 Backpropagation (BPN)

La Backpropagation es una red formada por capas, con propagación hacia delante, que está completamente interconectada entre capas. Por lo tanto no existen conexiones de retroalimentación, ni conexiones que salten una capa para ir directamente a otra superior.

La BPN es una red de aprendizaje supervisado, que emplea un ciclo *propagación–adaptación* de dos fases. Durante el entrenamiento, una vez que se ha aplicado un patrón a la entrada de la red como estímulo, este se propaga desde la primera capa hacia delante a través de las capas superiores hasta generar una salida. Ésta se compara con la salida deseada y se calcula una señal de error que se propaga hacia atrás, hacia todas las neuronas de las capas ocultas. Sin embargo las neuronas ocultas sólo reciben una fracción de la señal total del error, basándose aproximadamente en la contribución relativa que haya aportado cada neurona a la salida de la red. En función de esta señal de error percibida, se actualizan los pesos de conexión de cada neurona, para hacer que la red converja hacia un estado que permita clasificar correctamente todos los patrones de entrenamiento.

La importancia de este proceso de aprendizaje consiste en que, a medida que se entrena la red, las neuronas de las capas intermedias se organizan a sí mismas de tal modo que las distintas neuronas aprenden a reconocer distintas características del espacio total de entrada. Varias investigaciones han demostrado que la BPN tiende a desarrollar relaciones internas entre neuronas con el fin de organizar los datos de entrenamiento en clases. Esta tendencia se puede extrapolar, para llegar a la hipótesis consistente en que todas las unidades de la capa oculta de una Backpropagation son asociadas de alguna manera a características específicas del patrón de entrada como consecuencia del entrenamiento. Lo que sea o no exactamente la asociación puede no resultar evidente para el observador humano, lo importante es que la red encuentra una representación interna que le permite generar las salidas deseadas cuando se le dan las entradas, en el proceso de entrenamiento. Esta misma representación interna se puede aplicar a entradas que la red no haya visto antes, y la red clasificará estas entradas según las características que compartan con los ejemplos de entrenamiento.

Este último concepto es importantísimo, se trata de la capacidad para poder generalizar a través de ejemplos.

2.2 Red de Contrapropagation (CPN)

La red de Contrapropagación resulta de una combinación de tipos de redes existentes con anterioridad. Hecht-Nielsen sintetizó esta arquitectura combinando una estructura conocida con el nombre de *red competitiva* con otra denominada *outstart*.

Dado un conjunto de pares de vectores $(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_L, \mathbf{y}_L)$, la CPN puede aprender a asociar un vector \mathbf{x} en la capa de entrada con un vector \mathbf{y} en la capa de salida. Si la relación entre \mathbf{x} e \mathbf{y} puede describirse mediante una función continua, ϕ , tal que $\mathbf{y} = \phi(\mathbf{x})$, entonces esta red aprenderá a

aproximar esta correspondencia para todo valor de x en el intervalo especificado por el conjunto de vectores de entrenamiento [6].

La red consta de tres capas. La capa 1 es sólo de entrada. Para cada unidad de la capa intermedia se calcula su valor neto de entrada y se produce una competencia para ver qué unidad posee la entrada neta mayor. Esa unidad es la única que envía valores a la capa de salida.

Al igual que en la BPN, la capa de entrada se utiliza para distribuir los datos entrantes hacia otras capas de la red. Pero a diferencia de la BPN, la CPN suele hacer un preprocesamiento de los vectores de entrada consistente en su normalización (módulo 1). Al vector normalizado se lo denomina trama de reflectancia.

La trama de reflectancia es independiente de la intensidad total de la trama de entrada correspondiente. Por ejemplo, la trama de reflectancia correspondiente a la imagen de la cara de una persona sería independiente de si la persona se estuviera viendo con luz del sol muy brillante o a la sombra.

La capa competitiva (capa intermedia) está formada por un conjunto de neuronas que reciben el nombre de *instars*. Cada una de ellas responderá con un valor máximo a un cierto grupo de vectores de entrada perteneciente a una región distinta del espacio. La *instar* de mayor respuesta a un vector de entrada es la que determina la clase a la que pertenece dicho vector y es la única que comunica un valor distinto de cero a la próxima capa. En general se tendrán tantas *instars* como clases se quieran determinar.

Una sencilla regla de aprendizaje provoca que el vector de pesos de la *instar* se aproxime al promedio de los vectores pertenecientes a una misma clase.

Las unidades de la capa de salida intervienen en la formación de unas estructuras denominadas *outstars*. Una *outstar* está formada por todas las unidades de la capa de salida de la CPN y una única unidad de capa oculta, así habrá una *outstar* por cada *instar* de la capa intermedia.

La misión de las *instars*, a través de la competición, es reconocer la trama de entrada. La *outstar* en la que interviene la ganadora, es la encargada de nombrar o etiquetar a la clase que contiene dicha trama.

Debido a que en este trabajo se desea utilizar la CPN para realizar una simple clasificación de los vectores de entrada, se puede prescindir de la capa de salida. Simplemente se asocia a cada neurona de la capa competitiva un nivel de gris resultante. Así, para cada vector de entrada se obtiene un único nivel de gris para la imagen segmentada, determinado por la neurona que gana la competición. En [6] puede encontrarse un detallado análisis de esta arquitectura

3 SEGMENTACIÓN DE IMÁGENES CON REDES NEURONALES

En este trabajo se estudia la performance de las redes neuronales BPN y CPN para segmentar imágenes basándose en el aprendizaje supervisado de las texturas que componen el objeto de interés. Además se propone una combinación de ambas redes para mejorar los resultados de la segmentación en el método denominado BPNⁿ-CPN. En él se plantea la utilización de n redes Backpropagation pequeñas que realizan una pre-segmentación que mejora el rendimiento y reduce los requerimientos de cómputo.

3.1 Segmentación con BPN

Siendo la Backpropagation una red con aprendizaje supervisado, es necesario seleccionar el conjunto de vectores y la salida esperada para la fase de entrenamiento. Como se pretende identificar texturas, el nivel de gris de un pixel no brinda la suficiente información para poder realizar una clasificación eficiente. Así es necesario considerar al pixel junto con sus vecinos.

Se ha trabajado con distintos tamaños de vecindarios utilizando ventanas cuadradas de $2n+1$ pixels de lado, con $n>0$, siendo el pixel analizado el que se encuentra en el centro de la ventana, es decir el que posee coordenadas $(n+1,n+1)$. El tamaño de la capa de entrada de la red BPN queda establecido en $(2n+1)^2$ neuronas (ver Fig. 1).

El número de nodos en la capa oculta es sumamente importante y condiciona el rendimiento de la red durante la clasificación. La experimentación realizada en este trabajo ha demostrado que es necesario un número grande de neuronas ocultas directamente proporcional al tamaño del vecindario elegido.

Finalmente, cada nodo de la capa de salida representa una clase distinta. La red se entrena para que en cada instante sólo una neurona arroje el valor 1, mientras que la otra permanece en cero.

La función de transferencia utilizada en las neuronas de la capa de salida y oculta ha sido la sigmoide logística: $f(x) = 1/(1+e^{-x})$.

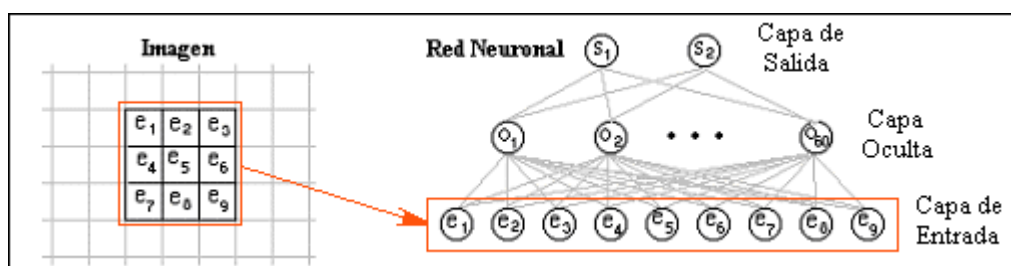


Figura 1 Ejemplo de una BPN con vector de entrada de 3x3, capa oculta de 60 neuronas y capa de salida de 2 neuronas (con capacidad de distinción entre 2 clases)

3.1.1 Fase de entrenamiento

Durante la fase de entrenamiento se produce la adaptación de los pesos de la red para que aprenda a reconocer las regiones pretendidas. La cantidad de clases a reconocer está dada por la cantidad de neuronas de la capa de salida. Dado que en este trabajo se busca la segmentación de un solo tipo de objeto, se han utilizado redes con 2 neuronas en la capa de salida.

Para conseguir el conjunto de vectores de entrenamiento, se toman dos muestras, una del objeto que desea segmentarse y otra de aquello que se considera fondo de la imagen. De cada muestra, se obtienen tantos vectores como neuronas de entrada haya. Así para un vector de dimensión 9 (3x3) se extraen los patrones de entrenamiento desde una ventana de (5x5) pixeles, consiguiendo exactamente 9 vectores para formar parte del conjunto de entrenamiento.

En general, para un vector de entrada de $n \times n$, se obtendrán exactamente n^2 patrones en una ventana de $(2n-1) \times (2n-1)$ pixels (ver Fig. 2).

El entrenamiento se prolonga hasta conseguir una cota de error global promedio aceptable. No obstante para una buena segmentación es necesario que todos los patrones de entrenamiento resulten bien clasificados.

Un factor crítico para el buen funcionamiento de una red BPN es determinar correctamente la cantidad de nodos ocultos de la misma.

Dependiendo del tipo y la cantidad de tramas a reconocer este valor puede ser distinto. Además la inicialización aleatoria de los pesos provoca que con la misma arquitectura en ocasiones se consiga entrenar exitosamente a la red y en otras no.

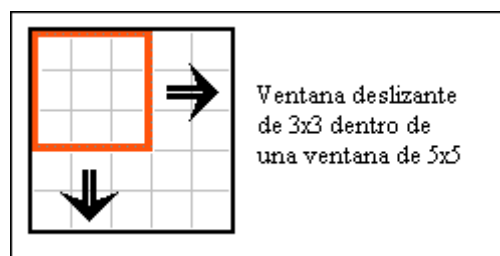


Figura 2 Una ventana deslizante de 3x3 se acomoda exactamente en 9 posiciones distintas dentro de una ventana de 5x5

3.1.2 Fase de reconocimiento

La capa de salida determina el nivel de gris del pixel resultante. Se utilizan tantos niveles de gris como neuronas de salida posea la red (relacionada con la cantidad de tipos de objetos distintos que se desea segmentar). En este trabajo se han utilizado dos neuronas para identificar sólo un tipo de objeto diferenciándolo del fondo. Así la salida de la red se corresponderá con un pixel blanco o con uno negro.

Para obtener la imagen segmentada se desplaza una ventana cuadrada -del tamaño de la capa de entrada de la red- por la imagen original, se estimula a la BPN con los valores de sus pixels y la salida de la misma determina el nivel de gris correspondiente al pixel central en la imagen transformada (Ver fig.3).

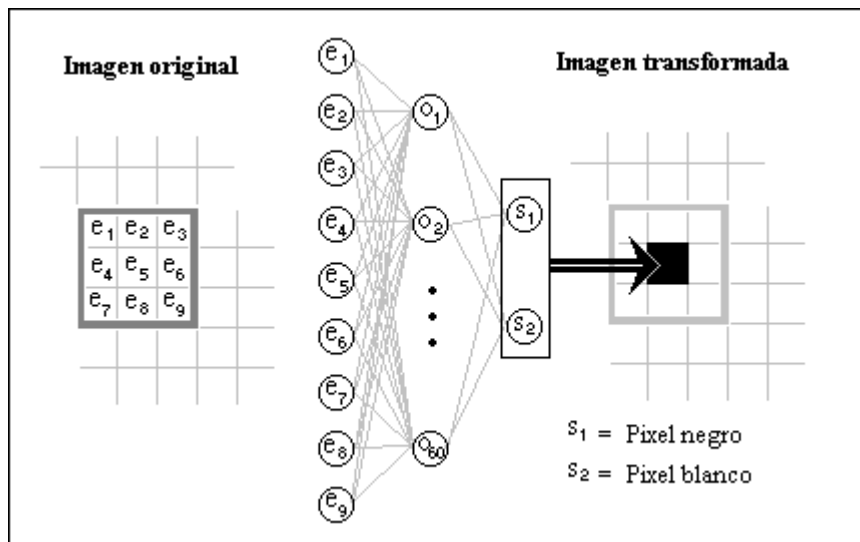


Figura 3 Ejemplo de funcionamiento de una BPN con entrada de (3x3) y 2 neuronas de salida. Si se activa la neurona S_1 el pixel de la imagen resultante será negro, de lo contrario será blanco.

3.2 Segmentación con CPN

Debido a que se desea realizar una simple clasificación de los vectores de entrada, se puede prescindir de la capa de salida. Simplemente se asocia a cada neurona de la capa competitiva un nivel de gris resultante. Así, para cada vector de entrada se obtiene un único nivel de gris para la imagen segmentada, determinado por la neurona que gana la competición.

Los vectores de entrada que estimulan la red se corresponden con una matriz cuadrada de píxeles -vecindario- valiéndose aquí las mismas consideraciones en cuanto a su conformación y tamaño ya expuestas al tratar la BPN.

Cada nodo de la capa competitiva representa una clase de textura distinta. La arquitectura de la CPN es mucho más chica que la de la Backpropagation ya que sólo cuenta con la capa de entrada y la competitiva conformada por sólo dos neuronas. Por ello los requerimientos de cómputo también son sensiblemente menores a los de la BPN.

3.2.1 Criterio de distancia

Para determinar la neurona que gana la competición -aquella cuyo vector de peso sea más parecido al vector de entrada- se han utilizado dos criterios distintos de distancia: la distancia euclídea y el producto interior. La utilización del producto interior presupone la normalización de los vectores de entrada y de pesos de la red haciendo que la clasificación de las tramas no sean sensibles a las

condiciones de iluminación. Salvo ello, no se han detectado diferencias importantes en cuanto al rendimiento de la CPN utilizando uno u otro criterio.

3.2.2 Fase de entrenamiento

Valen aquí las mismas consideraciones realizadas para la Backpropagation. Para cada imagen será necesario que la red “*aprenda*” a reconocer las texturas pretendidas.

Pero a diferencia de la red Backpropagation, en la CPN se pueden calcular los vectores de pesos de las unidades competitivas (asociadas a cada una de las texturas que se pretende reconocer) simplemente como el promedio de los vectores de entrenamiento de la clase correspondiente.

De esta manera la utilización de la CPN es significativamente más rápida que la Backpropagation puesto que se puede obviar la fase de entrenamiento. Además ya se ha dicho que su arquitectura es mucho más pequeña ahorrándose el procesamiento que realiza la BPN para evaluar sus numerosas neuronas ocultas.

La variante auto-organizada de la CPN, donde la capa competitiva aprende a identificar distintas regiones durante el entrenamiento, no ha sido contemplada en este trabajo que se focaliza en la segmentación de objetos aprendida en forma supervisada. Por ello, se seleccionan las muestras correspondientes a cada textura para calcular directamente y sin entrenamiento los pesos de la red.

3.2.3 Fase de reconocimiento

La capa competitiva determina el nivel de gris del pixel resultante. Se utilizan dos neuronas -niveles blanco y negro- para diferenciar objeto de fondo.

De igual forma que la expuesta al tratar la BPN, para obtener la segmentación deseada se desplaza por la imagen original una ventana cuadrada con las dimensiones apropiadas para estimular la entrada de la red. Una vez evaluada la CPN se determina el nivel de gris del pixel central en la imagen resultado según sea la neurona ganadora de la competición.

3.3 Segmentación con BPNⁿ-CPN

Se propone este método para mejorar la relación entre costo de procesamiento y calidad de la segmentación de los dos casos anteriores. Como se expone en la sección de experimentación mientras que la BPN produce mejores segmentaciones que la CPN, la cantidad de neuronas ocultas necesarias es muy grande y por lo tanto la fase de entrenamiento y aún la de reconocimiento es sumamente costosa. La CPN es muy rápida comparada con la BPN pero ha demostrado una performance más baja para la segmentación.

El método BPNⁿ-CPN consiste en utilizar una o más redes BPN para obtener una primera segmentación. Sin embargo no es necesario completar el entrenamiento de estas redes hasta que todos los vectores sean clasificados correctamente, ahorrando así mucho tiempo de cómputo. La imagen semi-segmentada alimenta a una CPN que completa la tarea. Los pesos de las neuronas competitivas de esta arquitectura se calculan utilizando la misma posición de las ventanas de entrenamiento definidas durante el aprendizaje de las BPN pero tomando los patrones de la imagen semi-segmentada en lugar de la imagen original. La figura 4 muestra esquemáticamente una solución para $n=1$.

Sin embargo la ganancia suele ser mayor si se utilizan n redes BPN (con n número par) entrenadas de manera independiente para reconocer un determinado objeto en las imágenes a segmentar. Los pesos iniciales de las BPN son aleatorios por lo que, partiendo de un lugar distinto en el espacio de búsqueda, al utilizar el algoritmo de descenso más pronunciado sobre la superficie de error, suelen converger a soluciones ligeramente distintas.

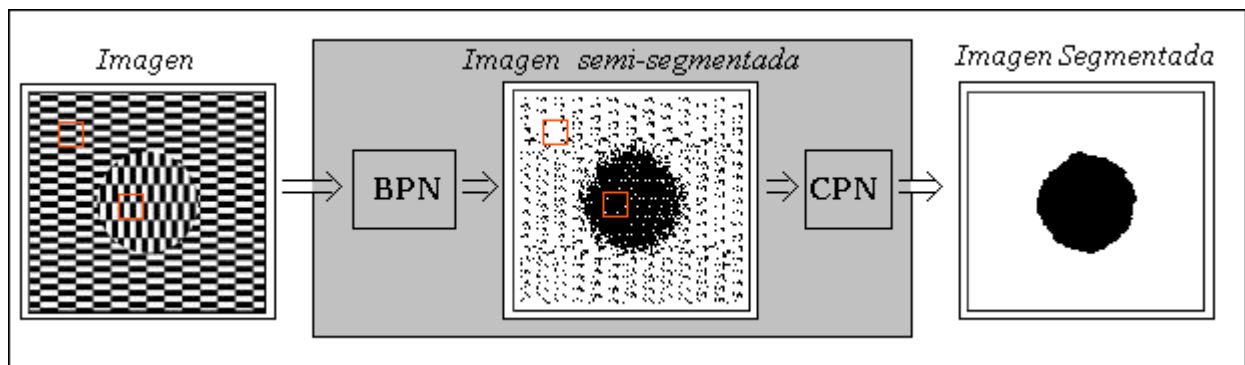


Figura 4 Esquema BPN¹-CPN. Las marcas rojas identifican las regiones sobre las que se extraen los vectores de entrenamiento

Una vez entrenadas independientemente cada BPN, se presenta al sistema la imagen a segmentar. Se obtiene una imagen intermedia semi-segmentada aplicando todas las BPN a la imagen original. Para cada pixel analizado, el resultado en la imagen intermedia será aquel identificado por la mayoría de las BPN intervinientes. En caso de empate el valor del pixel resultante será un nivel de gris intermedio. En este trabajo se han utilizado los siguientes niveles de gris: 0 cuando la mayoría de las redes identifica que es un pixel perteneciente al objeto a segmentar, 255 cuando la mayoría de las BPN identifican al pixel como perteneciente al fondo y 128 en caso que no haya mayoría en la identificación del pixel.

Esta imagen intermedia suele ser mejor, y por ende también el resultado final, que la obtenida con BPN¹-CPN, es decir con una única red BPN. Esta mejora se observa aún cuando se utilicen redes BPN lo suficientemente chicas como para mantener los requerimientos de cómputo similares entre los BPNⁿ-CPN con distintos valores del parámetro n .

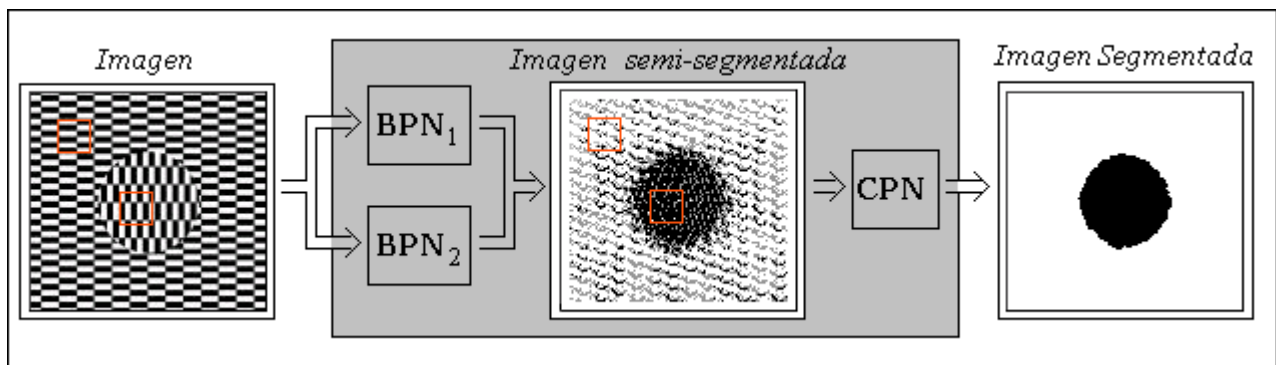


Figura 5 Esquema BPN²-CPN. Las marcas rojas identifican las regiones sobre las que se extraen los vectores de entrenamiento

4 EXPERIMENTACIÓN REALIZADA

Se han probado los tres métodos (BPN, CPN, y BPNⁿ-CPN con $n=1,2,4,6$) sobre un conjunto de imágenes monocromáticas con 256 niveles de grises. También se han utilizado en la experimentación imágenes con 2 niveles -blanco y negro-, útiles para poner en evidencia en forma clara y sencilla las limitaciones y potencialidades de los métodos.

La experimentación ha sido cuidadosamente diseñada pues la comparación entre los métodos debe realizarse con especial atención debido a que todos ellos son sensibles al tipo de imagen a segmentar y al conjunto de patrones seleccionados para la fase de entrenamiento.

En el análisis comparativos del método BPNⁿ-CPN con distintos valores del parámetro n , se han utilizado sistemas con similares requerimientos de cómputo. Así, si se utilizaron m neuronas ocultas

en la red Backpropagation del método BPN¹-CPN, se usaron a lo sumo m/n neuronas en la capa oculta de las n redes Backpropagation del método BPNⁿ-CPN.

También se ha tenido especial cuidado en utilizar el mismo conjunto de patrones de entrenamiento para los distintos métodos, intentando así determinar la performance de cada uno de ellos evaluados en iguales condiciones.

4.1 Resultados obtenidos

Las figuras 6, 7, 8, y 9 muestran sólo algunos de los resultados obtenidos durante la experimentación. Los métodos han sido probados en una amplia gama de imágenes reales y sintéticas. Para la presentación de los resultados se han elegido aquellos que evidencian con más claridad la potencia y debilidad de cada estrategia sin que ello implique pérdida de generalidad.

Las pruebas realizadas confirmaron que, para una gran cantidad de texturas, la red CPN se comporta adecuadamente segmentando el objeto de interés en forma apropiada (fig. 6a). Para estos casos la utilización de la CPN es siempre preferible a los otros métodos ya que el tiempo de procesamiento requerido es muy inferior al de los demás. Esta diferencia se debe al cálculo rápido y directo de los pesos de conexiones entre neuronas y sobre todo al tamaño reducido de su arquitectura.

Sin embargo, la capacidad de la CPN para distinguir algunos objetos en ocasiones es inferior a la demostrada por la red BPN. Ejemplo de ello lo constituye la segmentación de la imagen que se observa en las figuras 6b y 7. La razón es que ciertos objetos, claramente distinguibles para el ojo humano, pueden estar conformados por texturas que originan conjuntos de vectores con centroides -vector promedio- muy cercanos entre sí. Esto hará que la red CPN sea incapaz de clasificarlos correctamente.

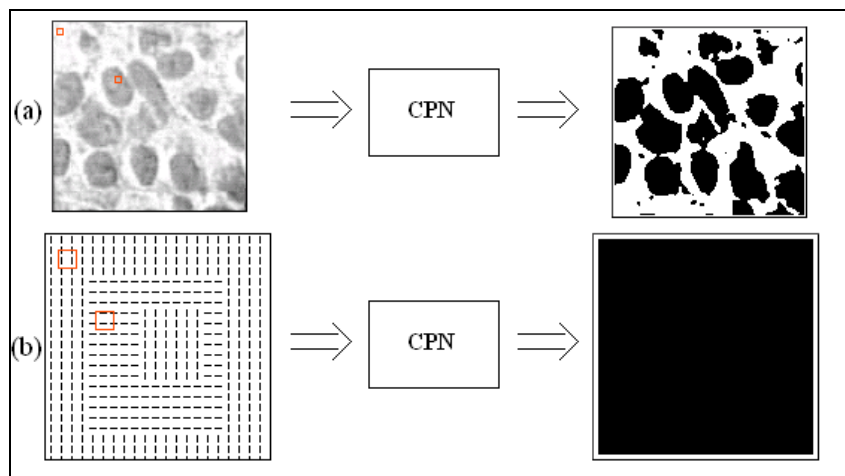


Figura 6 Las marcas rojas identifican las regiones sobre las que se extrajeron los vectores de entrenamiento. La CPN segmenta adecuadamente la imagen (a) pero es incapaz de hacerlo para la imagen (b)

La red BPN ha mostrado mayor poder de segmentación que la CPN. Siguiendo el ejemplo anterior la figura 7 muestra una segmentación bastante precisa de la figura que no puede ser identificada por la CPN. Sin embargo el tiempo de cómputo necesario para la segmentación con BPN es sensiblemente mayor al demandado por la CPN. Aquí no puede evitarse la etapa de entrenamiento cuyo éxito y duración depende del tamaño de la capa oculta de la red. La tabla 1 presenta los datos extraídos durante la realización de 20 pruebas de segmentación de la imagen de la figura 7 utilizando una BPN con 49 neuronas de entrada (matriz de 7x7 pixels) con distintos tamaños de capa oculta.

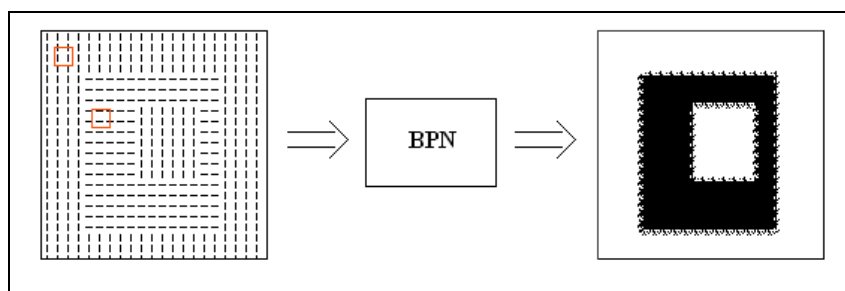


Figura 7. Las marcas rojas identifican las regiones sobre las que se extrajeron los vectores de entrenamiento.

La gran cantidad de iteraciones utilizada en el entrenamiento de la BPN está dada por la necesidad de aprender a clasificar correctamente todos los patrones del conjunto de entrenamiento, de lo contrario no se obtiene una buena segmentación.

Tabla 1 Performance BPN

Neuronas Ocultas	Entrenamientos exitosos (%)	Promedio de Iteraciones
100	35	> 1000
200	80	130
300	100	80

Se han obtenido mejores resultados que los anteriores al aplicar el método aquí propuesto con parámetro $n=1$. BPN¹-CPN hace innecesario invertir tanto esfuerzo en el aprendizaje de la red BPN. La figura 8 muestra los resultados obtenidos por este método en la segmentación de la misma imagen de la figura 7. Para realizarla se utilizaron 80 neuronas ocultas en la red BPN y se entrenó durante sólo 20 iteraciones. Aún no habiendo conseguido la clasificación correcta de todos los patrones de entrenamiento, la BPN generó una imagen semi-segmentada que no representó inconveniente para que la CPN completara adecuadamente la segmentación. Similares resultados se han obtenido utilizando 50 neuronas ocultas en la BPN y 200 iteraciones de entrenamiento. Compárense estos resultados con los datos descritos en la tabla 1 para determinar cuánto mejor a la red Backpropagation resulta el método BPN¹-CPN.

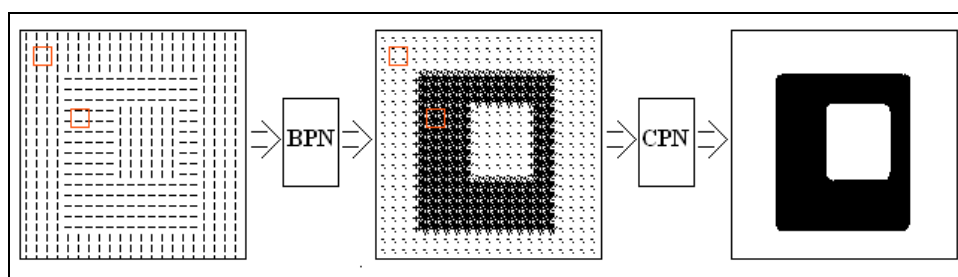


Figura 8. Las marcas rojas identifican las regiones sobre las que se extrajeron los vectores de entrenamiento. El método BPN¹-CPN consigue una mejor segmentación y en menor tiempo que la BPN

Para apreciar el rendimiento del método BPNⁿ-CPN con distintos valores para el parámetro n , es necesario utilizar imágenes con alguna dificultad extra. Por ejemplo en la figura 9 se utilizan los métodos BPN¹-CPN y BPN⁴-CPN para segmentar una ventana sobre una pared de piedras (fig. 9a.). La dificultad aquí es que el objeto a identificar -la ventana- no está compuesto por una textura uniforme. Se han realizado pruebas para $n = 1, 2, 4$ y 6 encontrándose los mejores resultados para $n=4$. No obstante, se ha observado que el valor más adecuado para este parámetro depende fuertemente del tipo de textura a segmentar y del conjunto de vectores de entrenamiento

seleccionado. Como ocurre con otros parámetros importantes de la teoría de redes neuronales, la experimentación previa en cada caso puede arrojar el mejor indicio sobre cuál será el valor más apropiado para n .

Para los ejemplos de la figura 9 se han utilizado 300 neuronas ocultas en la red backpropagation del método BPN^1 -CPN y sólo 50 en las correspondientes al método BPN^4 -CPN. Así este último, aún mostrando los mejores resultados, requirió menor tiempo de cómputo.

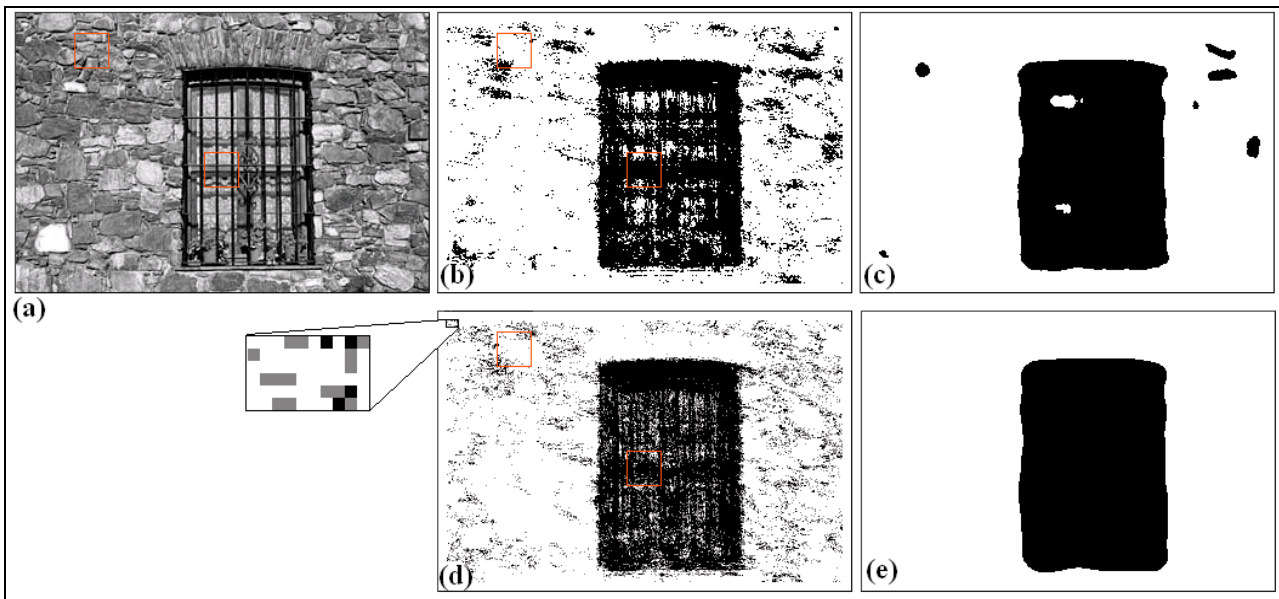


Figura 9. (a) Imagen a segmentar. (b) Imagen semi-segmentada método BPN^1 -CPN. (c) Imagen segmentada método BPN^1 -CPN. (d) Imagen semi-segmentada método BPN^4 -CPN. (e) Imagen segmentada método BPN^4 -CPN.

5 CONCLUSIONES

Se ha propuesto un nuevo método basado en redes neuronales para realizar segmentación de imágenes digitales: BPN^n -CPN. Se ha evaluado su rendimiento para distintos valores del parámetro n y el de las redes BPN y CPN utilizadas individualmente.

Se concluye que la red CPN constituye el método más rápido con menor requerimiento de cómputo. Sin embargo, puede no hallar la mejor segmentación en algunos casos.

La red BPN puede segmentar mejor que la CPN pero a costa de un largo entrenamiento y gran cantidad de neuronas ocultas por lo que se desaconseja su utilización. En su lugar, es recomendable el método BPN^1 -CPN que muestra mejor rendimiento que la BPN pero con mucho menos carga computacional.

Finalmente la utilización del método BPN^n -CPN para $n > 1$ ha demostrado ser aún más eficiente que BPN^1 -CPN obteniendo resultados de mejor calidad con el mismo o menor requerimiento de cómputo.

En cuanto al valor más apropiado para el parámetro n puede afirmarse que el mejor rendimiento aparece para $n > 1$. Sin embargo, aún no se ha podido establecer algún criterio aplicable a todos los casos para determinar exactamente el valor más adecuado para este parámetro. Por el contrario, dicho valor parece depender fuertemente del problema particular a resolver. Actualmente se está trabajando en esa dirección intentando realizar una clasificación de distintas texturas que permita, conociendo la clase a la que pertenece la que se quiere segmentar, aproximar el valor más recomendable para el parámetro n .

REFERENCES

- [1] Campbell, N. W., Thomas B. T. and Troscianko T.: A Two-Stage Process for Accurate Image Segmentation. *Sixth International Conference on Image Processing and its Applications*. IEEE, (1997).
- [2] Castleman, K.: Digital Image Processing. *Prentice Hall*, New Jersey, Cap. 5, 15. Pp. 68-83, 302-315. (1979).
- [3] Drozdowicz B., Bernasconi G., Reyes M., Saba F. Simon G.: Segmentación Semiautomática de Imágenes de Resonancia Magnética, basada en Redes Neuronales Artificiales. *Ciencia, Docencia y Tecnología. Universidad Nacional de Entre Ríos. Issn 0327-5566* pp.117-155 (2005).
- [4] Fernandez Muñoz J.A. Red Neuronal Celular para el Recuento de objetos en Imágenes Aéreas. *XXII Jornadas de Automática*. Barcelona. (Septiembre de 2001).
- [5] Nattkemper T. W., Wersing H., Schubert W., Ritter H.: A neural networkarchitecture for automatic segmentation of fluorescence micrographs. *ELSEIVER. Neurocomputing* 48 357–367. (2002).
- [6] Freeman, James A. & Skapura, David M.: Redes neuronales Algoritmos, aplicaciones y técnicas de programación. Addison–Wesley, 1991. Versión en español de: Rafael García -Bermejo Giner. Addison–Wesley Iberoamericana (1993).
- [7] Gonzalez Rafael C. Richard E. Woods: Tratamiento digital de imágenes. Addison–Wesley/Diaz de Santos.
- [8] Pavlidis, Th.: Algorithms for Graphics and Image Processing. *Computer Science Press*. Berlin, Cap. 4. Pp. 65-72. (1982).
- [9] Reyes Aldasoro C. Algorri ME.: A combined algorithm for image segmentation using neural networks and 3d surface reconstruction using dynamic meshes. *Revista Mexicana de Ingeniería Biomédica*. Vol XXI Núm. 3. pp 73-81 (Julio-Septiembre 2000).
- [10] Rosenfeld, A. and Kak, A.: Digital Picture Processing. *Academic Press INC*. New York, 1976, I: Cap. 10. Pp. 55-151. (1976).
- [11] Yalin Miao, Xianglin Miao, Zhengzhong Bian, Kai Chen and Gang Yu: A New Effective and Powerful Image Segmentation Method. *Lecture Notes in Computer Sciencie*. ISSN: 0302-9743 Volumen 3497 (2005).