

# **ENTRENAMIENTO DE UNA RED NEURONAL ARTIFICIAL PARA LA CLASIFICACIÓN DE HOJAS ASOCIADAS AL CULTIVO DE MAÍZ**

Espinosa Gualdrón, Diana Judith, Muñoz Neira Milton Javier  
UNISANGIL, Facultad de Ciencias Naturales e Ingeniería Programas de Ingeniería de sistemas e Ingeniería Electrónica  
San Gil, Colombia

dianajespinos@gmail.com  
mmunoz@unisangil.edu.co

---

-Ingeniera de Sistemas. Actualmente se desempeña como profesional de soporte a la infraestructura tecnológica de la oficina de Tránsito y Transporte de San Gil, Santander. Sus temas de interés son: Seguridad Informática, Comunicación de Datos, Visión Artificial.

- Ingeniero Electrónico. Actualmente culmina estudios de Especialización en Automatización Industrial. Desde el 2006 ha sido director del programa de Ingeniería Electrónica en UNISANGIL sede San Gil, donde ha liderado trabajos de investigación formativa en robótica. Sus temas de interés son: Control y Automatización, Inteligencia Artificial, Visión Artificial, Sistemas Caóticos y Tecnologías Emergentes. Es Professional Student Member de la IEEE y de la Robotic & Automation Society.

**Resumen**— En este proyecto se evalúa una red neuronal artificial, con un algoritmo de propagación inversa (Backpropagation), para la clasificación de imágenes de maíz entre imágenes de maleza. El cumplimiento de los objetivos conllevó el uso de diversas funciones de activación y algoritmos de entrenamiento, mediante las herramientas de la Neural network Toolbox de MATLAB, y de la toolbox Image Processing, para el tratamiento y procesamiento de las imágenes de las hojas de maíz y de maleza. El porcentaje de aciertos contra el porcentaje de error en el entrenamiento de la red se evidencia por medio de varios experimentos.

**Palabras clave**— reconocimiento de malezas, redes neuronales artificiales, backpropagation

**Abstract**— The present project evaluates an artificial neuronal network, with a back propagation algorithm, for classification of images of maize and weeds. The develop of objectives of the project included testing of various activation functions and different training algorithms, through the use of Matlab Neural Network and Image Processing Toolbox, this last tool for processing of images of maize and weeds. Percentage of successes against errors in the ANN training is evidenced with various experiments.

**Keywords** — weed recognition, artificial neural networks, back propagation

## I. INTRODUCCIÓN

El estudio de sistemas automáticos para el reconocimiento de malezas en cultivos se ha intensificado en los últimos tiempos, a consecuencia de la utilidad que representan dichos sistemas para la aplicación puntual de herbicidas, y no por aspersión general; lo que conlleva a la reducción de efectos nocivos sobre el ambiente y la salud humana que el uso indiscriminado de los mismos implica (consultar, por ejemplo, a Martínez [1] y a Nivia [2] para ampliar información referente a los perjuicios del uso indiscriminado de herbicidas).

Dichos sistemas, en asocio con móviles robotizados, pueden llegar a posibilitar un manejo de la aplicación racional de herbicidas en tiempo real sobre el cultivo, lo que contribuiría también a reducir los costos que sobre la producción agrícola acarrea el uso de aspersores uniformes [3]. Para su construcción, se estima que el uso de redes neuronales artificiales, en asocio con diversas técnicas para extracción de las características asociadas a las hojas a reconocer, es de gran utilidad [4], [5].

El trabajo desarrollado optó por clasificar de acuerdo al borde o contorno lleno de la hoja, por medio del entrenamiento de una red neuronal en configuración BP (back propagation) de tres capas. La capa de entrada se constituyó con los píxeles de la imagen previamente puesta en blanco y negro (imagen binaria), en tanto que la capa de salida se formó con tantas neuronas como clases de hojas distintas a reconocer, según el entrenamiento a realizar. Por su parte, las neuronas de la capa oculta se variaron de manera experimental y creciente, de cara a mejorar la respuesta de la red a entrenar. El coste computacional por la cantidad de entradas a procesar por la red se equilibra, en parte, con la sencillez del método empleado para la identificación de la característica (bordes) de cada hoja a reconocer.

## II. PROCESAMIENTO DE LAS IMÁGENES

Se realizó una etapa de adquisición de datos que implicó la identificación y recolección de hojas de maíz y de maleza y su posterior fotografiado. En el caso de las hojas de maíz es importante tener en cuenta que el tiempo de crecimiento no puede ser mayor a un (1) mes, pues en esta época el cultivo de maíz es más vulnerable a la maleza que lo rodea. Se obtuvo así una base de datos de 81 imágenes de maíz y 81 imágenes de maleza mediante una cámara fotográfica digital Samsung S760 de 7.2 megapíxeles, inicialmente con una resolución de 2592x1944, las cuales se convirtieron a una resolución de 256x256 para facilitar la manipulación de los datos. En efecto, la resolución inicial implicaba 5 MB por imagen aproximadamente, lo que conlleva al uso de una capacidad de memoria de 181 MB para un experimento básico. Estos tamaños, además de que serían imprácticos para una implementación en tiempo real basada en el patrón tomado para el entrenamiento, pueden limitar el normal funcionamiento del software de entrenamiento de la red neuronal artificial.

Las fotografías fueron tomadas en un ambiente controlado, con iluminación de luz incidente por medio de lámparas fluorescentes. Fueron capturadas 162 imágenes (fotografías) de maíz y de maleza (Ver Figura 1. Entorno de Adquisición).

Debido que las imágenes de las hojas de maíz y de maleza originales tienen diversas variaciones, es necesario realizar un procesamiento a las mismas. Entre los métodos más comunes está el filtrado [6]. Se pueden aplicar filtros pasa bajas para remover ruido de alta frecuencia, o pasa altas para resaltar bordes (o el contorno de la

imagen) y corregir difusión.

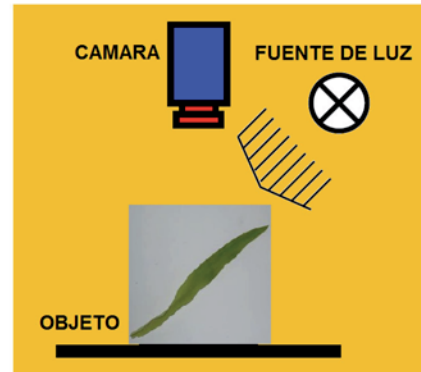


Figura 1. Entorno de Adquisición (Adaptado de [7])

Para el presente proyecto, con apoyo de Matlab (Versión de prueba R2010a, autorizada por MATHWORKS) se siguieron los siguientes pasos de procesamiento: indexado y conversión a escala de grises (función `rgb2gray`), binarización (función `graythresh` para la determinación del umbral e `im2bw` para conversión a blanco y negro) y extracción del perímetro (función `bwperim`).

En la figura 2 se pueden ver algunos ejemplos del procesamiento elaborado.



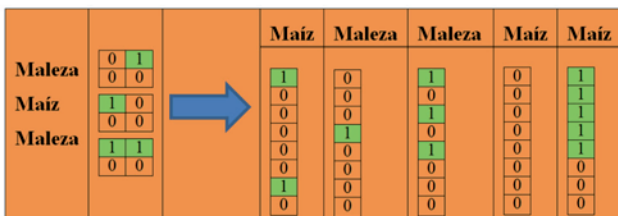


**Figura 2.** Ejemplos del procesamiento de las imágenes de maíz y maleza

El algoritmo de procesamiento construido captura N imágenes de maíz y maleza, dependiendo del entrenamiento y los grupos que se formaron para este fin. A continuación se guardaron todas las imágenes dentro de una matriz. Cada imagen posee un tamaño de 256x256 píxeles. De modo que esta matriz posee dentro de ella N matrices de cada imagen, donde cada una de estas matrices se convirtió en un vector de 65536 posiciones. Cada posición de este vector fue una entrada de la red neuronal.



**Figura 3.** Creación de la matriz de una imagen binarizada (el tamaño de esta matriz por cada imagen de 256 x 256 binaria)



**Figura 4.** Matriz de matrices a una matriz de vectores (contiene todas las matrices de las imágenes convertidas en vectores)

Terminando el procesamiento y tratamiento a las imágenes, se inició el entrenamiento de la red neuronal artificial, mediante la Neural network Toolbox de MATLAB [8].

### III. ENTRENAMIENTO DE UNA RED NEURONAL APLICADA AL RECONOCIMIENTO DE PATRONES

El comportamiento y los diversos algoritmos de entrenamiento para redes neuronales artificiales, puede consultarse en Bonifacio y Sanz Molina [9], Nilsson [10] y Delgado [11].

Para el reconocimiento de patrones en imágenes de hojas, el algoritmo de propagación inversa (backpropagation) presenta, además de buenas tasas de aprendizaje, facilidad para su posterior implementación en un sistema tiempo real [4].

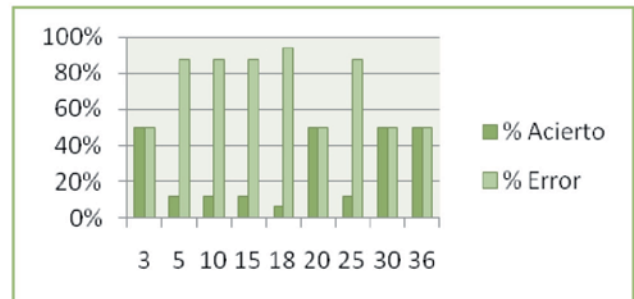
Se empleó la función newff de Matlab para crear la red neuronal a entrenar. El primer argumento fue la matriz de los vectores de entrada. El segundo argumento fue la matriz de vectores de salida. El tercer argumento fue el tamaño de la capa oculta. Mediante el cuarto argumento se modificaron las funciones de activación para la capa oculta y la de salida (la capa de entrada no requiere de función de activación porque no contiene neuronas). El quinto argumento contiene el nombre de la función de entrenamiento que se utiliza. Si sólo se proporcionan tres argumentos, la función de activación oculta por defecto para la capa de entrada es TANSIG y la función

predeterminada de la capa de salida es PURELIN. El valor por defecto de la función del algoritmo de entrenamiento es TRAINLM.

#### A. Entrenamientos

Los entrenamientos se realizaron utilizando un equipo de cómputo marca HP (Hewlett-Packard), con 250 GB, de disco duro SATA de 5400 RPM, y 2048 MB de memoria RAM, y procesador X2 dual core AMD Athlon. En total fueron realizados seis tipos distintos de entrenamientos, de los cuales cuatro serán presentados. Para los experimentos descritos a continuación se emplearon imágenes de los bordes de las hojas. Experimentos con la figura completa (contorno binarizado lleno) también fueron realizados arrojando tasas de entrenamiento similares.

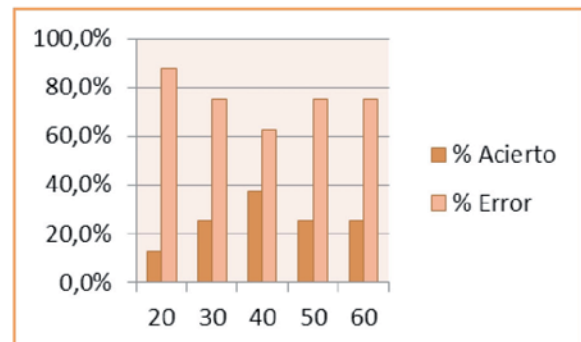
1) *Experimento Uno:* En este primer experimento se entrenó con imágenes a las que se les extrajeron los bordes, creando 9 grupos de 18 imágenes intercaladas; es decir, una imagen de maíz seguida de una de maleza y así sucesivamente. Entre estas, 7 eran para el entrenamiento y 2 para validaciones. Los resultados obtenidos fueron:



**Figura 5.** Resultados variando las capas ocultas

Los datos de la figura 5 corresponden a un entrenamiento de la red neuronal con funciones de activación purelin y tansig. Variando las capas ocultas y, con el mismo número de iteraciones, el mayor porcentaje de aciertos fue del 50%.

2) *Experimento Dos:* En este experimento se crearon 9 grupos de 4 imágenes cada uno (una de maíz y tres de maleza), 7 de ellas se utilizaron para entrenar y 2 para validación. Los resultados obtenidos fueron:



**Figura 6.** Resultados variando las iteraciones

Los datos de la figura 6 corresponden al entrenamiento de la red neuronal con las mismas funciones de activación por defecto nombradas anteriormente. Variando las iteraciones en cada entrenamiento y con el mismo número de capas ocultas, se logró un 37.5% de aciertos.

- 3) *Experimento Tres:* En este experimento se crearon 9 grupos de 4 imágenes cada uno (una de maíz y tres de maleza), 7 de ellas se utilizan para entrenar y 2 para validación. Con base en los resultados anteriores se emplearon 30 neuronas en la capa oculta (fijas), funciones de activación logsig y tansing para las neuronas de la capa oculta y de salida, respectivamente, con algoritmo de entrenamiento en contra del gradiente de error, atendiendo al cambio de signo de la variación del mismo (función trainrp en Matlab).

Cuando se validó con los grupos de entrenamiento se alcanzó un 100% de aciertos y 0% de error. Cuando se validó con otros grupos (validación), se alcanzó un porcentaje del 50% de aciertos (ver figura 7).

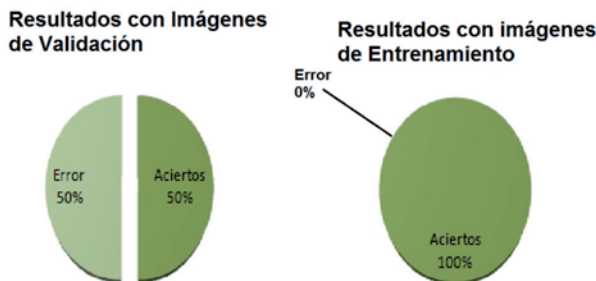


Figura 7. Resultados tercer entrenamiento

- 4) *Experimento Cuatro:* En este experimento se crearon 9 grupos de 18 imágenes intercaladas; es decir, una imagen de maíz seguida de una de maleza. Entre estas, 7 eran para el entrenamiento y 2 para validaciones. Igual que en el experimento anterior, y con base en los resultados anteriores, se emplearon 30 neuronas en la capa oculta (fijas), funciones de activación logsig y tansing para las neuronas de la capa oculta y de salida, respectivamente, con algoritmo de entrenamiento trainrp (en Matlab).

Cuando se validó con los grupos de entrenamiento se alcanzó un 100% de aciertos y 0% de error. Cuando se validó con otros grupos (validación), se alcanzó un porcentaje del 63.8% de aciertos. En la figura 8 se puede notar los alcances obtenidos en este entrenamiento.

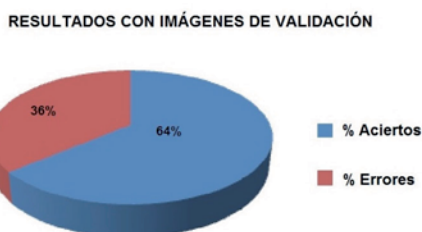


Figura 8. Resultados cuarto entrenamiento

#### IV. CONCLUSIONES

- El aprendizaje en una red neuronal artificial es sensible al algoritmo de entrenamiento empleado y a la forma en que la red backpropagation es formada. Para un algoritmo tipo resilient

backpropagation (trainrp) se obtuvieron las tasas más altas de aprendizaje (100% para los patrones de entrenamiento, 50% y 64% para patrones de verificación distintos a los de entrenamiento). Lo anterior es explicable dado que, al ser los patrones escogidos señales de tipo todo o nada (0 o 1, contorno o no contorno), el entrenamiento es más idóneo si se atiende al cambio de signo de la variación del gradiente del error.

- Entre las diversas formas de implementar la red backpropagation (backpropagation, backpropagation en cascada y backpropagation con realimentación), backpropagation ofreció los mejores resultados (100% de acierto para patrones de entrenamiento, 50% y 64% para patrones de validación). Este hecho es justificable por la topología propia de la red implementada, la cual contaba con 65536 neuronas en la capa de entrada, menos de 100 neuronas en la capa oculta y sólo 4 o 18 en la capa de salida, de acuerdo al experimento, por lo que backpropagation en cascada o en realimentación puede desorientar el entrenamiento.
- Los pesos iniciales de entrenamiento modifican sensiblemente el resultado del mismo. De esto se concluye que es preferible realizar pequeños entrenamientos (número de iteraciones bajo), hasta encontrar un aprendizaje inicial aceptable (25%) a partir del cual se realice un entrenamiento fuerte (más de 100 iteraciones), sin reinicio de la red en cada iteración.
- El entrenamiento obtiene, de igual modo, un alto número de aciertos, cuando los patrones de entrenamientos son presentados de manera variada. De hecho, la tasa de entrenamiento más alta (64%) se obtuvo con un conjunto de entrada conformado por 18 imágenes intercaladas entre imágenes de maíz y de maleza, con cada imagen de maleza distinta una de la otra.
- Las toolbox de Matlab para procesamiento de imágenes y entrenamiento de redes neuronales artificiales son herramientas de fácil manejo, que pueden economizar tiempo a la hora de ejecutar un proyecto complejo como el reconocimiento de patrones en hojas de diversas cultivos.
- Para futuros trabajos, a fin de incrementar la tasa de aciertos en la clasificación, es recomendable entrenar la red neuronal a partir de patrones descriptores de la forma de las hojas o de patrones asociados a la textura de las mismas.

#### REFERENCIAS

- [1] C. Martínez, Facultad de Ciencias, UdelaR. (2011, May 04). Taller sobre: "Impactos toxicológicos y ecotoxicológicos asociados al uso del herbicida glifosato" [en línea]. Disponible en: [http://uneamfagro.org/phocadownload/taller-glifosato\\_docs/Presentacion\\_MARTINEZ.pdf](http://uneamfagro.org/phocadownload/taller-glifosato_docs/Presentacion_MARTINEZ.pdf)
- [2] E. Nivia. (2011, May). Efectos Sobre la Salud y el Ambiente de Herbicidas que Contienen Glifosato. Ambiente Ecológico [en línea]. Edición 87 (Abril-Diciembre del 2003). Disponible en: [http://www.ambiente-ecologico.com/ediciones/2003/087\\_02.2003/087\\_Publicaciones\\_EcoPortal.php3](http://www.ambiente-ecologico.com/ediciones/2003/087_02.2003/087_Publicaciones_EcoPortal.php3)
- [3] T. Grift, Q. Zhang, N. Kondo & K. C. Ting. A Review of Automation and Robotics for the Bio-Industry. Journal of Biomechanics Engineering, Vol. 1, No. 1, 2008, pp. 37-54.
- [4] T. F. Burks, S. A. Shearer, J. R. Heath, & K. D. Donohue. Evaluation of Neural-network Classifiers for Weed Species



- Discrimination. *Biosystems Engineering*, vol. 91, no. 3, 2005, pp. 293-304.
- [5] L. Tang, L. F. Tian, B. L. Steward & J. F. Reid. Texture-based weed classification using gabor wavelets and neural network. Selangor, Malaysia. 2009. 14 p.
- [6] MATHWORKS. *Image Processing Toolbox™ 6 User's Guide*. Matlab. 1258 p.
- [7] A. Erhardt-Ferron (2010, Jun 13). *Theory and Applications of Digital Image Processing* [en línea]. Disponible en: [http://abe-research.illinois.edu/pubs/T\\_Grifi/Automation\\_and\\_Robotics\\_for\\_Bio-Industry.pdf](http://abe-research.illinois.edu/pubs/T_Grifi/Automation_and_Robotics_for_Bio-Industry.pdf)
- [8] D. Howard, M. Beale & M. Hagan. *Neural Network Toolbox™ 6 User's Guide*. Matlab. Mathworks. 901 p.
- [9] M. Bonifacio y A. Sanz Molina. *Redes Neuronales y Sistemas Difusos*. 2ª ed. México. Alfaomega, 2001, 537 p.
- [10] NILSSON. *Inteligencia artificial una nueva síntesis*. Mc. Graw Hill. España: Madrid, 2001, 458p.
- [11] A. Delgado. *Inteligencia Artificial y Minirobots*. 1ª ed. Ecoe ediciones. Colombia: Bogotá, 1998. 202 p.