

Sistema de lectura automática de formularios manuscritos

Lucio MARTINEZ GARBINO^{1,2}, Augusto J. CARIMATTO^{1,2}, Leandro DI MATTEO¹,
Juan C. GÓMEZ^{1,3}, Claudio A. VERRASTRO^{1,2}

¹ Grupo de Inteligencia Artificial UTN-FRBA, Buenos Aires, Argentina,

² CNEA, Centro Atómico Ezeiza. ³ INTI, Instituto Nacional de Tecnología Industrial.
luciojmg@cae.cnea.gov.ar; carimatto@cae.cnea.gov.ar; Leandro.DiMatteo@ieee.org;
juanca@inti.gov.ar; cverra@cae.cnea.gov.ar

RESUMEN

La construcción de sistemas de lectura automática requiere la puesta a punto de una gran cantidad de parámetros en las diferentes etapas de su diseño y aún durante su utilización. En este trabajo se presenta, por un lado, una metodología de trabajo para agilizar el desarrollo de sistemas de lectura automática, por otro, utilizando esa metodología y las herramientas desarrolladas, se analizan típicas alternativas de diseño y luego se implementa un sistema automático de lectura de formularios manuscritos. Se realiza una experiencia concreta. Finalmente se presentan y comparan los resultados obtenidos, y se extraen conclusiones, tanto sobre la utilización de las herramientas de desarrollo como sobre el sistema implementado.

Palabras Claves: Reconocimiento de caracteres, OCR, Redes Neuronales Artificiales.

1 INTRODUCCIÓN

En sentido amplio, el área del reconocimiento de patrones (RP), es la disciplina que se ocupa de la descripción y clasificación de objetos a partir de mediciones de determinadas características de los mismos [1]. Dentro del área del RP, el reconocimiento óptico de caracteres (OCR “Optical Character Recognition”) tiene por objetivo extraer información en formatos codificados (por ejemplo caracteres en código ASCII) a partir de imágenes de texto, que pudieron ser obtenidas de muy diferentes fuentes, por ejemplo de formularios, cheques, sobres de correo [2], típicamente manuscritos, o de etiquetas de identificación, chapas patentes [3][4], etc. de fuentes predefinidas.

Para el cumplimiento de sus objetivos, en esta disciplina se emplean diferentes técnicas y herramientas, entre ellas: estadísticas, inteligencia artificial y procesamiento digital de señales e imágenes.

Desde el comienzo del trabajo, en la investigación bibliográfica, se detectó la necesidad de contar con una herramienta que permita agilizar las pruebas, la sintonía de parámetros y la evaluación de la gran cantidad de alternativas presentes en la etapa de desarrollo de sistemas de esta naturaleza.

En estos problemas debe distinguirse entre la etapa de diseño y la de producción. En la primera se establecen los procesos a realizar y sus parámetros. En la segunda se utiliza el sistema para el RP propiamente dicho.

El proceso completo puede ser dividido en un conjunto de tareas que se realizan una a continuación de otra, a saber:

1. Obtención de la imagen en formato digital donde se encuentra el texto buscado.
2. Segmentación en subimágenes que contengan un solo carácter a reconocer.

3. Procesamiento de las subimágenes para eliminar posible ruido y adecuarlas a la posterior extracción de características.
4. Extracción de características o descriptores.
5. Clasificación, a partir de los descriptores, de los caracteres de las subimágenes.

Una vez terminada la etapa de diseño, se está en condiciones de construir un OCR simple y evaluar su resultado.

Para cada una de las etapas hay que definir una serie de subprocesos, el orden de aplicación y los parámetros de los mismos. Todo esto depende también del problema específico que se esté abordando y de sus condiciones particulares. No es lo mismo “leer” la matrícula de la placa de un auto a partir de una foto desde la cabina de un estacionamiento que los números en un cheque bancario escaneado.

En este trabajo se presenta una metodología de diseño y su forma de utilización. Se desarrolla un sistema de reconocimiento de números extraídos de formularios manuscritos analizando diferentes alternativas de diseño y se realiza una experiencia concreta. Como descriptores se utilizan combinaciones de los momentos de Hu [5], Zernike [6] y características conocidas como Zoning [7]. En todos los casos se emplean redes neuronales artificiales tipo *backpropagation* [8] de diferentes topologías como método de clasificación o de reconocimiento propiamente dicho. El objetivo es seleccionar la mejor topología y conjunto de descriptores para las diferentes redes de cada símbolo a reconocer. Finalmente se analizan sus resultados y se extraen conclusiones. Hay que destacar que la información obtenida para cada campo no necesariamente consta del carácter mejor calificado, sino que puede ser un conjunto ordenado de los caracteres mejor calificados para cada campo. De esta manera, una etapa posterior de ejecución, de nivel más alto, que cuente con más información, podría mejorar el resultado que se obtendría considerando solamente al mejor calificado. Ejemplo: en castellano luego de una letra ‘Q’ debe seguir una letra ‘U’. Entonces ante valores de salida similares del clasificador para las letras ‘Q’ y ‘O’ se podría analizar el carácter siguiente y tomar la decisión en función de esta información. Se mejora aún más considerando el resto de la palabra y así salvar, en el mismo ejemplo, el caso del nombre propio “Lourdes”. El tipo de sistemas que hace uso de información contextual, de diccionario, gramática, etc. se conoce en la literatura como reconocedor de caracteres inteligente (ICR) “Intelligent Character Recognition” [12].

2 METODOLOGÍA DE DISEÑO

Para facilitar el diseño de manera que se pueda poner a punto un sistema de reconocimiento de caracteres, se estableció una metodología de trabajo que permite indicar los subprocesos a reali-

zar, su orden, fijar sus parámetros y documentar de manera automática las pruebas realizadas. Además, se estableció una construcción modular de los subprocesos de forma que cada bloque sea construido y mantenido independientemente del resto.

Resulta así un sistema de módulos de software que permite la construcción de un sistema de RP en general, y en particular para este caso un OCR de caracteres manuscritos. La construcción modular, permite que se trabaje con la herramienta más adecuada para cada módulo y se ajuste a cada caso particular de desarrollo.

Para lograr esos objetivos se definió un estándar para indicar a los diferentes módulos los parámetros de entrada y salida. Se implementó mediante el uso de un archivo de configuración (AC) del experimento en formato “.ini” y en ellos se informa: la ubicación de los archivos de entrada, el/los archivos de salida y los procesos a realizar junto a sus parámetros. De esta manera cada experimento de un desarrollo particular es una serie de procesos encadenados, guiado y parametrizado con la información contenida en el AC.

La historia de los procesos aplicados por un módulo y sus parámetros se dejan en un archivo. De esta forma quedan bien documentadas las pruebas realizadas. Esto le otorga trazabilidad al sistema.

Esta metodología de trabajo junto a los módulos básicos para el tratamiento de la información, integran una herramienta de desarrollo que facilita las pruebas de diferentes procesamientos de imagen, descriptores y métodos de clasificación.

El desarrollo de un sistema de reconocimiento de caracteres es en esencia un proceso iterativo. La metodología empleada permite comparar fácilmente los resultados obtenidos con diferentes procesos, parámetros y/o topologías del sistema de clasificación. De esta manera se puede seleccionar el mejor conjunto de procesos, parámetros y método de clasificación para cada caso, inclusive para un mismo sistema, aplicar diferentes alternativas para cada clase a clasificar i.e. podría haber un método diferente para cada caracter.

Con el objeto de probar adecuadamente cada alternativa de sistema desarrollado se realiza una validación cruzada. Para eso se dividen los datos de entrada en tres conjuntos: *entrenamiento*, *validación* y *prueba*, los dos primeros para utilizarse durante la etapa de diseño y el último para prueba final de donde se extrae el resultado para la evaluación.

3 CONSIDERACIONES PRELIMINARES

En todos los casos prácticos, siempre que sea posible, hay que comenzar con un buen diseño del objeto base a partir del cual se pretende extraer la información. En el caso que aquí se presenta: el formulario. Ver figura 1.

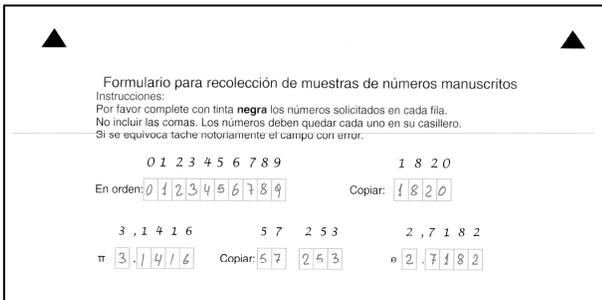


Figura 1: Parte superior de la imagen de un formulario de prueba utilizado en la experiencia

Además de características específicas respecto del contenido de formularios, hay que hacer una serie de consideraciones de formato a los efectos de su posterior tratamiento automático.

Es necesario colocar marcas de anclaje en la hoja que permita corregir ligeras desviaciones y/o inclinaciones de los campos de interés. Utilizar, para definir los campos, colores diferentes del de la tinta con que se completan los datos, ayuda significativamente en la segmentación.

4 DESARROLLO

Como se mencionó en la introducción, el sistema consta de 5 procesos: Adquisición, segmentación, procesamiento de la imagen, extracción de características y finalmente la clasificación.

Durante el desarrollo de un sistema de reconocimiento como el que se describe, es imprescindible evaluar y comparar los resultados siguiendo algún criterio, que en general depende del problema particular que se esté abordando. Por tal motivo se agrega un proceso extra que permite el análisis y comparación de los resultados obtenidos.

A continuación se describen los diferentes procesos.

4.1 Adquisición

Este proceso y el de segmentación están fuertemente influenciados por el diseño del formulario, tal como se expresó en la sección 3. En este trabajo se realizó el diseño del formulario con la restricción de que fuera en escala de grises. Se agregaron marcas de anclaje en las esquinas de la hoja y se delimitaron los campos con líneas de un gris claro para facilitar el llenado primero y la segmentación después.

Para la etapa de diseño, se desarrolló una aplicación que permite al usuario, a partir de la imagen correcta de uno de los formularios, identificar la posición (coordenadas) de marcas de anclaje y de los campos numéricos de interés. Esta información se almacena en un archivo de ubicación de campos (AUC), con un formato predefinido.

Tanto para la etapa de diseño, como para la de producción, la adquisición de datos se realiza digitalizando las imágenes de los formularios ya completados, por medio de un escáner. La resolución empleada para dicha operación fue de 600x600 píxeles por pulgada y el formato de archivo de salida “.tif”.

4.2 Segmentación

Los formularios ya tienen campos específicos para los datos que se desean obtener. La segmentación es el mecanismo que genera una subimagen de cada caracter de interés que fue escrito dentro de estos campos. Este proceso funciona prácticamente de la misma manera tanto en fase de diseño como en fase de producción.

Este módulo toma del AC del experimento la ubicación de los archivos con las imágenes de los formularios, el AUC con las coordenadas de interés y la ubicación para los archivos de salida.

A partir de los datos del AUC generado durante el diseño, el cual contiene las coordenadas de las marcas de anclaje, se realiza cualquier corrección necesaria tanto por desplazamiento y/o por rotación respecto de la imagen tomada como referencia. Esto es necesario ya que hay diferencias de impresión entre partidas distintas de un mismo formulario o por diferencias en el proceso de escaneado de los mismos.

Una vez corregido se procede a separar en subimágenes a cada campo de la imagen del formulario, empleando las coordenadas establecidas con la imagen del formulario de referencia.

Como resultado de esta etapa se tiene un archivo “.bmp” para cada uno de los campos contenidos en el formulario. En la figura 2 se muestran como ejemplo las imágenes de cuatro campos. El nombre del archivo incluye un identificador de campo y formulario al que pertenece.

En particular, durante la etapa de desarrollo, donde se conoce el carácter contenido en cada campo, se agrega este carácter a la codificación del nombre del archivo para su empleo en etapas posteriores de diseño.

4.3 Procesamiento de la imagen

El objetivo de esta etapa es adecuar la imagen del campo de cada carácter para su posterior extracción de características. Este módulo toma del AC la ubicación de la carpeta que contiene los archivos con las imágenes segmentadas en la etapa anterior.

Debe minimizarse el ruido agregado en los procesos que sufre el formulario, desde su impresión, pasando por el llenado con la información de interés y los procesos de digitalización y segmentación. Se busca también reducir algunas diferencias introducidas por los que completaron los formularios.

Las operaciones realizadas por los módulos desarrollados, y sus parámetros, indicados en el archivo AC, para los casos que se presentan como resultado en este trabajo son:

- 1 BorrarBorde
- 2 Dilatar, 2
- 3 Erosionar, 2
- 4 QuitarRuido, 4
- 5 CentrarGeométricamente
- 6 Expandir.
- 7 FiltroPB, 5
- 8 Esqueletizar.
- 9 Dilatar, 2
- 10 Dilatar, 2
- 11 Dilatar, 2

Como resultado del proceso de segmentación queda, en algunos casos, algo del borde gris claro que delimita el campo en la imagen a analizar. La operación BorrarBorde, calcula las proyecciones de la imagen contenida sobre los ejes horizontal y vertical y decide, con un algoritmo que emplea dos umbrales, si pone en blanco las columnas/filas de la imagen que pudieran contener algo del borde impreso.

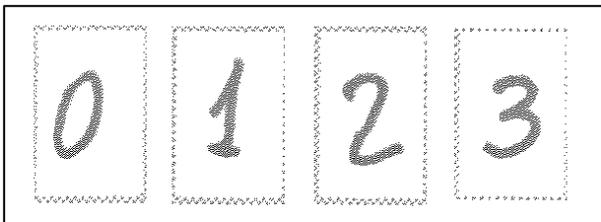


Figura 2: Ej. de caracteres segmentados de 4 campos

Los procesos siguientes de dilatación y erosión [9] tienen por objeto completar el trazo y eliminar posibles huecos pequeños en el interior del trazo o entre dos trazos muy próximos. El parámetro adjunto indica el tamaño en píxeles de la matriz a utilizar.

QuitarRuido analiza los píxeles de la imagen que pudieran pertenecer a un trazo, y si se encuentran aislados, los elimina. El grado de conexión, se pasa como parámetro.

Luego se centra la imagen del carácter haciendo coincidir el centro geométrico con el centro de la imagen.

Posteriormente se expande, conservando la relación de aspecto, hasta hacer que la imagen toque el borde horizontal o vertical.

Se le aplica luego, un filtro pasa bajos del tipo media móvil. El parámetro indica el tamaño del filtro.

A continuación se emplea un algoritmo de esqueletizado [9] y luego se ensancha el trazo mediante la aplicación de una serie de dilataciones.

En la figura 3 se observan los caracteres segmentados luego de aplicarle la serie de procesos recién indicados. Estos se corresponden con los de la figura 2.

La salida de este módulo es también un conjunto de archivos con las imágenes ya procesadas, una por cada archivo de entrada. Se almacenan en una carpeta de imágenes de salida indicada también en el AC.

4.4 Extracción de características

Una vez que las imágenes de los caracteres han sido segmentadas y preprocesadas, se debe extraer de las mismas la información necesaria para diferenciar un carácter de otro y así reducir la cantidad de información que se vuelca a la próxima etapa, la de identificación.

Este módulo toma del AC la ubicación de las carpetas para obtener los archivos de entrada y depositar el de salida. Funciona igual en la fase de diseño, que en la de producción.

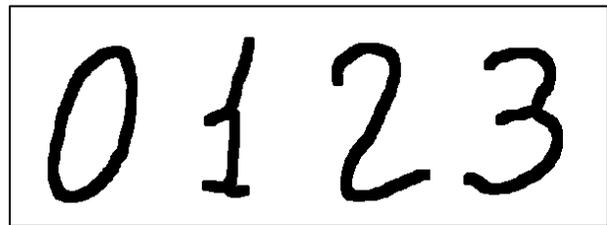


Figura 3: Imágenes de los caracteres procesados

Para este caso, las alternativas de descriptores a evaluar para los clasificadores de cada número son: los momentos de Hu [5], los momentos de Zernike[6] y unas características Zoning [7]. Tanto de los momentos de Hu, como de los de Zernike (donde se utiliza sólo el módulo) se extraen características invariantes a la rotación. Dado que, previamente, las imágenes se escalan y se centran, resultan características invariantes a la escala, a la posición y a la rotación. Si bien esta es una característica deseada, presenta inconvenientes a la hora de discernir entre caracteres como el ‘6’ y el ‘9’, que son iguales si no se considera su orientación. Por lo tanto es necesario fortalecer el sistema agregando características variantes a la rotación. En este caso se utilizaron las conocidas como zoning, que resultan de dividir a la imagen en una cuadrícula, y asignar a cada cuadrado el promedio del nivel de gris de de los píxeles que contiene cada uno.

4.5 Clasificación

El proceso de clasificación es un mecanismo que, a partir de los valores de los descriptores de la imagen, permite identificar a qué clase pertenece el carácter i.e. ‘0’, ‘1’..‘9’. En el presente proyecto se optó por el paradigma de redes neuronales artificiales (ANN) para la clasificación, por su buen desempeño como identificador de patrones, aprovechando su capacidad de genera-

lización. Se utiliza una red para cada clase a identificar. Estas redes neuronales son del tipo *backpropagation*, que es un tipo de red *feedforward* de aprendizaje supervisado [10].

La fase de diseño, para las redes neuronales, consiste en definir una topología, entrenarla y obtener los pesos sinápticos de la misma. El aprendizaje consiste en presentarle a la red pares entrada-salida conocidos (conjunto de entrenamiento). Esto es: los descriptores de imágenes como entrada y la clase a la que pertenecen como salida deseada. El algoritmo de entrenamiento va ajustando los pesos sinápticos minimizando el error cuadrático medio entre la salida obtenida y la deseada. Con el fin de evaluar la condición de finalización del entrenamiento y evitar el sobreentrenamiento “overfitting” u “overtraining”, se utiliza el segundo conjunto de datos, llamado conjunto de validación. Con este último se evalúa a la red, luego de cada pasada de datos de entrenamiento. Se finaliza el proceso cuando la red deje de mejorar la generalización [10].

El módulo programado permite ingresar un conjunto de topologías a ensayar. Éstas se definen por el número de capas ocultas, la cantidad de neuronas que conforman cada capa y las funciones de transferencia de cada una de ellas. Los datos de entrada a este módulo están compuestos por dos conjuntos de pares de entrada-salida. La salida es un archivo con los pesos sinápticos para cada red entrenada, de cada clase a identificar, para todas las topologías solicitadas.

En la fase de producción se utiliza la red con los pesos hallados para clasificar los caracteres a través de sus descriptores. La entrada es un archivo con un conjunto de características para cada campo a analizar y la salida es un archivo, para cada topología, con los resultados que produjo cada una de las redes para cada dato ingresado.

En el sistema OCR propuesto, cada clase a identificar tiene su red correspondiente y, ante la presentación de los descriptores elegidos para cada una de ellas, todas las redes responden con valores entre 0 y 1. Un sistema OCR simple, puede quedarse con la identificación hecha por la red que obtenga el mayor valor.

4.6 Evaluación de resultados

Este proceso se agrega para generar indicadores que permitan el posterior análisis y evaluación del comportamiento de la alternativa de diseño de sistema bajo estudio.

Por ejemplo, dejando fijos los procesos de segmentación, el procesamiento de imágenes y la utilización de ANN como clasificador, se analizan diferentes descriptores y topologías de redes. De cada alternativa de diseño se obtiene entonces, como producto, un conjunto de redes entrenadas y sus pesos sinápticos, listos para utilizarse en producción. Por cada clase a identificar (caracteres del ‘0’ al ‘9’) se obtienen tantas redes como topologías se hayan indicado en el AC. Para analizar otros descriptores (o combinaciones de ellos) se deben realizar nuevos experimentos.

El objetivo es construir un OCR, con la mejor combinación de topología-descriptores para cada clase a identificar. Para ello es necesario establecer una figura de mérito que permita elegir, por clase, la mejor combinación topología-descriptores. De esta manera se genera un resultado por clase, evitando realizar todas las pruebas cruzadas con todas las combinaciones de configuraciones posibles.

La figura de mérito seleccionada para este trabajo está conformada por el promedio de dos índices. Estos índices son el de especificidad y sensibilidad [11] que son típicas de la evaluación de métodos de diagnósticos.

Para calcular estos índices es necesario definir previamente un umbral tal que valores de salida de una red, por encima del mismo indiquen que los descriptores presentados corresponden a la

clase que aprendió a identificar, mientras que valores por debajo de ese umbral indican que no corresponden.

Dado que estos índices son función del nivel de umbral elegido para cada red, se selecciona el umbral que maximiza la figura de mérito establecida.

Los índices de especificidad y sensibilidad se calculan como:

$$\begin{aligned} \text{Sensibilidad} &= \text{cVP}/(\text{cVP}+\text{cFN}) \\ \text{Especificidad} &= \text{cVN}/(\text{cVN}+\text{cFP}) \end{aligned}$$

Donde cVP es la cantidad de Verdaderos Positivos que es la número de caracteres del conjunto de prueba que perteneciendo a la clase que se está evaluando i.e. ‘0’ son reconocidos como tales. cFN cantidad de Falsos Negativos, que son aquellos que, perteneciendo a la clase, se evalúan como ajenos a la misma. cVN cantidad de Verdaderos Negativos, esto corresponde a aquellos caracteres que no siendo de la clase evaluada son correctamente clasificados como ajenos. cFP cantidad de Falsos Positivos y corresponde a los caracteres que no perteneciendo a la clase evaluada son clasificados como pertenecientes a la misma.

La sensibilidad indica cuán bueno es el clasificador para identificar a los caracteres de la clase para la que fue entrenado, mientras que especificidad indica la cualidad de rechazo de los individuos que no pertenecen a la clase para la que fue entrenado.

Para la evaluación, se desarrolló un módulo que, mediante el conjunto de datos de prueba, permite presentar los resultados, y calcular los índices de desempeño de las alternativas bajo consideración.

5 EXPERIENCIA DE PRUEBA

Con el objeto de probar la metodología de diseño, las herramientas y módulos desarrollados que se presentan en este trabajo, se realizó la experiencia de diseñar y construir los procesos centrales de un sistema de lectura automática de formularios manuscritos. Se analizaron diferentes alternativas de diseño.

Para la obtención de una base de datos propia, se diseñó el formulario que se muestra en la figura 1 y con voluntarios se completaron 149 ejemplares del mismo. Cada formulario consta de 110 campos. 11 para cada símbolo, y distribuidos de diversas formas, de manera de simular un llenado de campos reales.

A partir de allí se escanearon los formularios ya completados obteniendo los archivos “.tif” para cada uno.

Se tomó un formulario patrón y con él se utilizó la aplicación desarrollada, que permite ubicar la posición de las marcas de anclaje, en este caso cuatro triángulos en las esquinas de la hoja, y también las ubicaciones de los campos definidos con rectángulos en gris claro. Se generó así el AUC con los datos necesarios para automatizar el proceso de segmentación

Se obtuvieron así más de 15000 imágenes que se separaron en tres conjuntos, entrenamiento (60%), validación (20%) y prueba (20%).

El procesamiento de imágenes se mantuvo tal como se describió previamente en la sección 4.

Como alternativas de diseño se propuso analizar seis combinaciones de descriptores y ocho diferentes topologías de redes.

Se editaron, según una plantilla, los AC con todos los procesos a realizar y sus parámetros para cada uno de los experimentos propuestos. Se estableció la configuración de las redes, con funciones de transferencia tangente hiperbólica y sigmoideal para la capa oculta y la de salida respectivamente y con diferentes alternativas para la cantidad de neuronas de la capa oculta.

Las combinaciones de descriptores utilizadas son: Hu, Zernike, Hu+Zoning, Zernike+Zoning, Zernike+Hu y Zernike+Hu+Zoning. De Hu se utilizaron 7 momentos, de Zernike 11 y 2 de Zoning.

Luego se ejecutaron el resto de los procesos y se obtuvieron los pesos sinápticos de las redes neuronales utilizadas. Se evaluaron todas las redes con el conjunto de datos de prueba y se almacenaron los índices de sensibilidad y especificidad en función del valor de umbral.

En la figura 4 se pueden apreciar los índices de sensibilidad, especificidad y el promedio de ambos, para el caracter '9' para una topología de 20 neuronas en la capa oculta y con los descriptores Hu+Zernike+Zoning. Para cada red se calcula el índice que servirá de figura de mérito para su evaluación.

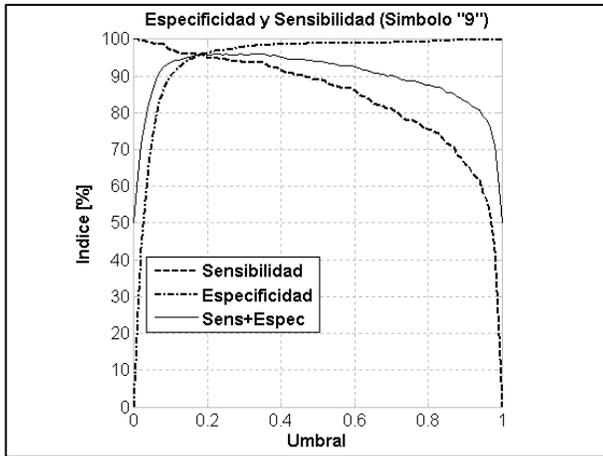


Figura 4: Especificidad y sensibilidad en función del umbral empleado

Finalmente, se agrupan en una matriz todas las redes que están dedicadas a reconocer la misma clase. En la tabla 1, a modo de ejemplo, se observan los valores del índice propuesto para el caso de las redes dedicadas a reconocer al símbolo '9'. Según este índice la mejor red tiene 20 neuronas en la capa oculta y como descriptores los momentos de Hu, Zernike y Zoning. De igual forma se hace para cada conjunto de redes de cada clase a identificar.

n° N	H	Z	H Zo	Z Zo	H Z	H Z Zo
5	86,4	87,6	94,9	96,5	90,7	96,4
8	83,1	89,0	96,6	97,0	91,5	97,0
10	83,9	86,8	96,8	96,3	91,2	97,1
12	84,1	89,0	97,0	97,0	89,1	97,1
15	88,0	86,3	96,5	96,7	87,5	96,1
20	87,4	87,3	95,9	96,1	88,8	97,3
25	86,0	86,8	95,3	96,7	88,7	93,7
30	86,8	87,5	95,6	96,3	88,3	95,3

Tabla 1: Figura de mérito para cada red dedicada a reconocer al '9'

Con este conjunto de redes seleccionadas según el máximo promedio de los índices de sensibilidad y especificidad, se construye un OCR simple, que se distinguirá como OCR*. Este, da como resultado de la clasificación al caracter cuya red indique el valor más alto de salida. Se le presenta todo el conjunto de prue-

ba y se calcula el nivel de reconocimiento como el porcentaje de caracteres bien clasificados por el OCR.

Para realizar una comparación se construyó otro OCR simple, OCR1, donde, para cada clase se emplearon redes de igual topología y la misma combinación de descriptores. Se seleccionó la que obtuvo el mejor resultado promedio de todos sus índices, siempre a igualdad de topología y descriptores. En esta experiencia resulta que las redes tienen la topología de 12 neuronas en la capa oculta y que utiliza los mismos tres descriptores que el OCR*.

6 RESULTADOS

Una vez realizados los experimentos el módulo de evaluación obtuvo los siguientes resultados:

Nivel de reconocimiento (NR)	
OCR*	OCR1
89.24%	87.48%

Tabla 2: Nivel de reconocimiento obtenido para el OCR* y OCR1

Se observa una mejora interesante al emplear redes de diferentes topologías para cada símbolo a reconocer. No hubo en cambio, variación en los descriptores.

7 CONCLUSIONES

La metodología presentada, que automatiza y documenta las pruebas, junto a los módulos desarrollados, facilitaron la construcción de un OCR, siguiendo un determinado criterio.

Establecer la figura de mérito, en este caso el promedio de los índices de especificidad y sensibilidad, permitió un análisis por separado de las redes de manera de elegir los mejores descriptores y el mejor clasificador para cada clase. Esto abre una nueva perspectiva en el diseño de OCR.

De la misma forma que se hizo en el experimento, es posible hacer pruebas para establecer descriptores muy específicos para subconjuntos de los símbolos a clasificar. Por ejemplo estudiando los descriptores del '2' y el '5' se encuentra cierta correlación, por lo tanto es prometedor encontrar un descriptor que sirva para separarlos.

Si bien las redes neuronales demostraron ser efectivas para la tarea de clasificación/reconocimiento, en trabajos futuros se implementarán otro tipo de clasificadores sugeridos en la bibliografía por ejemplo *support vector machines* [13].

8 REFERENCIAS

- [1] Robert J. Schalkoff, Pattern Recognition: Statistical, Structural and Neural Approaches, John Wiley, 1991
- [2] C.G. Leedham, Historical perspectives of handwriting recognition systems, University of Essex. The Institution of Electrical Engineers, London 1994.
- [3] Parisi, Di Claudio, Lucarelli and Orlandi, Car Plate recognition by neural network and image processing, University of Rome "La Sapienza"
- [4] Juanlan Feng et. al., Research of vehicle license plate character recognition method based on artificial neural network,

- 2nd International conference on informatics in control, automation and robotics, 2010
- [5] Hu, Visual pattern recognition by moment invariants, IRE Trans. Inform. Theory. vIT-8. 179-187, 1962
 - [6] O. D. Trier; A.K. JAIN; T.TAXT, Feature Extraction Methods for Character Recognition - A Survey.
 - [7] 1996V. Di Lecce, G. Dimauro, A. Guerriero, S. Impedovo, G. Pirlo, A. Salzo, Zoning design for hand-written numeral recognition, 2000
 - [8] Simone Marinai et. al., Artificial Neural Networks for Document Analysis and Recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence. Vol. 27 n°1 January 2005.
 - [9] Gonzalez & Woods, Digital image processing, Addison Wesley.
 - [10] Tsoukalas Lefteri, Fuzzy and neural approaches in engineering, Purdue University. John Wiley & sons. 1997
 - [11] http://www.fisterra.com/mbe/investiga/pruebas_diagnosticas/pruebas_diagnosticas.asp (05/2011)
 - [12] <http://www.haifa.il.ibm.com/projects/image/ocr/icr.html> (03/2011)
 - [13] Chen-Lin Liu et. al., Handwritten digit recognition: benchmarking of state of the art techniques, Pattern Recognition 36 (2003) pág. 2271-2285