

# Reconocimiento de caracteres escritos a mano mediante métodos de tratamiento de imagen y redes neuronales

Alexander Maestre Torres

Facultad de ingeniería, Universidad Popular del Cesar, Colombia

[ajmaestretorres@gmail.com](mailto:ajmaestretorres@gmail.com)

**Resumen.** La investigación en Reconocimiento Óptico de Caracteres (ROC) constituye una rama integral del procesamiento de imágenes, donde se han concebido diversas técnicas y metodologías. Su principal finalidad radica en la identificación de caracteres a partir de imágenes digitalizadas, representadas como conjuntos de píxeles. En este estudio, llevamos a cabo un proceso iterativo para el ROC, el cual se compone de cinco fases. Para ello, implementamos varios métodos de procesamiento de imágenes, y exploramos la aplicación de un modelo de red neuronal Perceptron Multicapa.

**Palabras clave:** reconocimiento óptico de caracteres (ROC), modelos predictivos, métodos de clasificación, redes neuronales, métodos supervisados de aprendizaje automatizado.

## Introducción

El Reconocimiento Óptico de Caracteres (ROC) es una fascinante área de investigación en el procesamiento de imágenes, abordando la tarea de identificar caracteres a partir de representaciones digitales compuestas por conjuntos de píxeles. En nuestro trabajo, hemos emprendido un proceso iterativo en el ámbito del ROC, estructurado en cinco fases distintas.

En particular, cabe destacar nuestra atención hacia el modelo de red Perceptron Multicapa. Este modelo no solo captura la novedad de las últimas tendencias en aprendizaje profundo, sino que también exhibe un potencial excepcional en la resolución de los desafíos asociados al reconocimiento de caracteres. La capacidad de estos modelos para aprender representaciones complejas y patrones sutiles en las imágenes digitalizadas ofrece una perspectiva emocionante para avanzar en la eficacia del ROC.

A través de este enfoque integral, nuestro objetivo es contribuir al avance de las metodologías aplicadas en el Reconocimiento Óptico de Caracteres, con la aspiración de mejorar la precisión y robustez en la identificación de caracteres en imágenes digitales, respaldando así el progreso continuo en este campo de investigación.

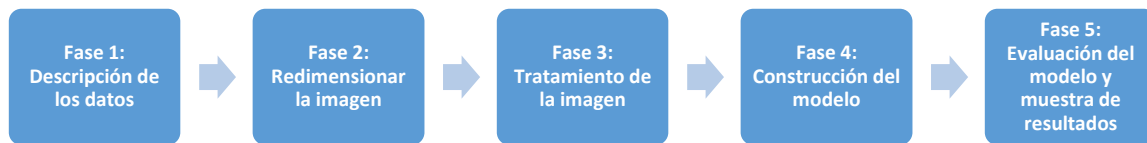


Fig. 1. Fases de la metodología del proyecto.

El reconocimiento de caracteres escritos a mano es un procedimiento que consiste en identificar un símbolo, ya sea bien un número o una letra, a partir de la imagen de cierta escritura por medio del uso de herramientas de inteligencia artificial. Actualmente existen múltiples aplicaciones para el procedimiento de reconocimiento de caracteres. Muchas de estas aplicaciones incluyen: la automatización de la redirección de cartas en el correo postal, la identificación de las matrículas de los coches en los radares o la digitalización de los apuntes escritos a mano.

En la era digital, la capacidad de las computadoras para comprender y procesar la escritura a mano es una habilidad fundamental con aplicaciones en campos que van desde el reconocimiento de texto en documentos hasta la mejora de la interfaz hombre-máquina. El reconocimiento de caracteres escritos a mano ha sido un desafío continuo en el campo de la inteligencia artificial y el procesamiento de lenguaje natural. Para abordar este desafío, se han desarrollado diversas técnicas, y una de las más notables es la red neuronal de Hopfield.

Nuestro proyecto consiste en desarrollar una red neuronal de Hopfield para la identificación de caracteres numéricos a partir de un conjunto de píxeles que representan estos caracteres escritos a mano. El desarrollo de este proyecto lo pretendemos realizar por medio de una metodología general compuesta de 3 fases para la identificación de dichos caracteres. Para ello, realizamos la adquisición previa de la imagen, el pre-procesamiento de la imagen, y el reconocimiento o clasificación de la imagen.

### **Fase 1: Descripción de los datos**

Normalmente, el proceso de reconocimiento óptico de caracteres inicia con la digitalización de las imágenes de escritura realizada a mano que contienen los dígitos, pero como hemos ya mencionado en este proyecto partimos de la necesidad de identificar o encontrar un conjunto de datos público que sea lo suficientemente robusto.

Este conjunto de datos será creado por nuestro propio criterio, utilizando la captura de imágenes de nuestra propia escritura que luego hemos separado en las muestras necesarias. Cada muestra tiene 1601 campos. Un campo que consideramos la salida del modelo representa el carácter y el resto de los campos son las entradas. Las entradas son los píxeles de la imagen que representa el dígito escrito a mano. Cada imagen tiene una resolución de 40x40, lo que hace un total de 1600 píxeles. Cada pixel se representa con un número entre el intervalo de 0 y 255 que indica el nivel de gris. De forma que el valor 0 representa el color blanco y el valor 255 el color negro.

## Fase 2: Redimensionar la imagen

La resolución de una imagen se refiere a la cantidad de detalles que contiene. Cuantos más píxeles se utilicen para representar una imagen, mayor será su resolución. Cada imagen se compone de una cuadrícula de píxeles, y si la imagen es en color, cada píxel se describe mediante tres valores que representan los colores rojo, verde y azul (RGB). En nuestro caso, la imagen es en escala de grises, por lo que cada píxel tiene un valor entre 0 y 255.

Para que las imágenes se ajusten a los modelos, que tienen un número fijo de entradas, es necesario que todas las imágenes tengan el mismo tamaño. En algunos casos, esto implica reducir el tamaño de la imagen, lo que puede resultar en la pérdida de información, mientras que, en otros casos, se debe ampliar la imagen y rellenar los píxeles adicionales a través de un proceso de interpolación.

Una vez que hemos procesado la imagen, podemos extraer características de ella que son útiles para mejorar la precisión de las predicciones del modelo. Esto implica medir cosas como el promedio de los valores de los píxeles, contar cuántos píxeles tienen un valor por encima de cierto umbral o identificar regiones dentro de la imagen. También podemos aplicar algoritmos para detectar características como la textura o determinar cuántos agujeros hay en ciertos patrones. Estas características adicionales ayudan al modelo a comprender mejor la imagen y a realizar predicciones más precisas.

## Fase 3: Tratamiento de la imagen

El tratamiento de imágenes es un conjunto de técnicas que se utiliza para mejorar la calidad de la imagen o para facilitar la búsqueda de información en ellas. Algunos ejemplos podrían ser eliminar el ruido, localizar bordes o suavizar la imagen.

Las técnicas más utilizadas y que podemos aplicar a nuestro problema y que aplicaremos en la fase dos son:

- **Binarización:** Consiste en convertir todos los píxeles de una imagen originalmente tengan varios valores a solamente dos tonos: blanco y negro. Para ello se establece un umbral por encima del cual convertiremos en negro y por debajo del cual será blanco. Esta operación es muy frecuente puesto que algunos algoritmos trabajan en esta escala. Las imágenes ocupan un tamaño en memoria muy pequeño puesto que cada píxel se puede representar con un píxel.

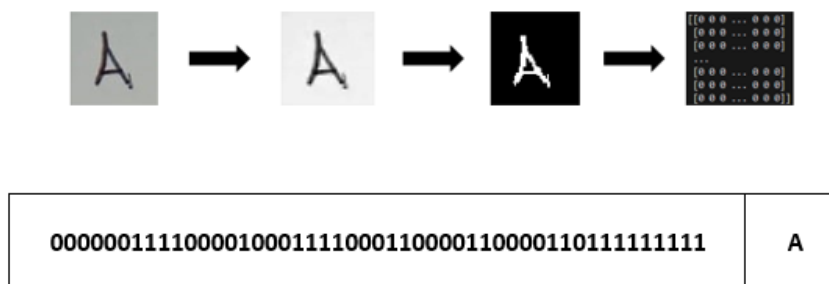


Fig. 2. Binarización de una imagen.

- **Fragmentación o segmentación:** Esta técnica consiste en seleccionar una parte de la imagen original. Para ello se desarrollan algoritmos que se basan en la detección de bordes que son pequeños cambios en la intensidad de los colores y otras propiedades. Existen muchos métodos genéricos de fragmentación, aunque para cada tipo de imagen se suele aplicar un algoritmo más específico. Por ejemplo, para detectar un cáncer de mama primeramente se aplica un filtro sobre las regiones sospechosas y posteriormente se aplica un algoritmo que evalúa ese conjunto de píxeles.

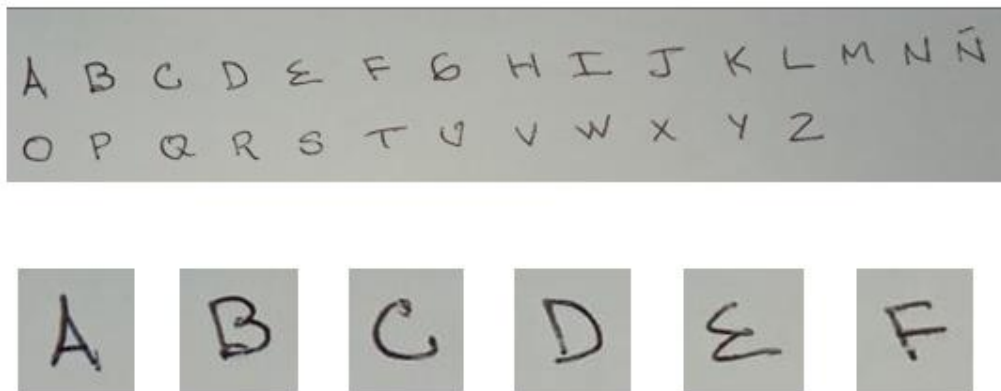


Fig. 3. Segmentación de una imagen.

#### Fase 4: Construcción del modelo

En este proyecto, nos proponemos desarrollar una red neuronal de Hopfield para la identificación de caracteres escritos a mano. La red aprenderá a reconocer y distinguir caracteres individuales, lo que es esencial en aplicaciones como el procesamiento de formularios, digitalización de texto y sistemas de reconocimiento de escritura a mano en tabletas y dispositivos móviles. La red neuronal de Hopfield, propuesta por John Hopfield en 1982, es un tipo de red neuronal recurrente que se destaca por su capacidad de almacenar y recuperar patrones. Su aplicación en el reconocimiento de caracteres escritos a mano ha demostrado ser prometedora debido a su capacidad para aprender y generalizar a partir de ejemplos de escritura manual. Esta red se basa en principios biológicos y utiliza conexiones ponderadas para representar y recuperar patrones de entrada. En esta fase, se debe definir la arquitectura de la red neuronal. Esto incluye la determinación del número de neuronas en la capa, la función de activación, y otros parámetros clave. Para nuestro caso hemos seleccionado una red neuronal Perceptrón Multicapa, que hemos diseñado a través de la librería MLPClassifier que nos provee Python, la cual hemos configurado de la siguiente manera:

Número de capas ocultas	3
Número de neuronas de cada capa	850
Función de activación de las capas ocultas	Logistic
Error	0.0001
Número máximo de iteraciones	1000

## Fase 5: Evaluación del modelo y muestra de resultados

La evaluación consiste en medir el grado de precisión de cada modelo generado. Esta fase informa si la dimensionalidad elegida es adecuada para la construcción del modelo. Para medir la calidad de los modelos utilizamos el accuracy de cada método.

La precisión (accuracy) es una métrica comúnmente utilizada en el contexto de problemas de clasificación en aprendizaje automático e inteligencia artificial. Representa la proporción de predicciones correctas que un modelo hace en relación con el total de predicciones realizadas.

En otras palabras, la precisión mide la fracción de ejemplos de un conjunto de datos que un modelo clasifica correctamente. Cuanto más alta sea la precisión, mejor es el rendimiento del modelo en términos de clasificación. La fase de prueba o evaluación consiste en medir el grado de precisión de nuestro modelo generado. Esta fase nos informa si la dimensionalidad elegida es adecuada para la construcción del modelo.

**Selección de Métricas de Rendimiento:** Identificar las métricas que se utilizarán para evaluar el rendimiento de la red. En el caso del reconocimiento de caracteres, las métricas comunes incluyen la precisión, la tasa de error y la matriz de confusión. Para nuestro caso hemos escogido la precisión, cuyo resultado fue:

<b>Precisión</b>	77%
------------------	-----

Esta cifra significa que la red ha logrado clasificar correctamente el 77% de las muestras en el conjunto de datos de prueba, un logro que destaca la capacidad del modelo para comprender y generalizar patrones en los datos. El hecho de que la red haya alcanzado un nivel de acierto del 77% sugiere una eficacia en su capacidad para aprender y adaptarse a las complejidades de los datos de entrada. Sin embargo, para una evaluación más completa, es esencial contextualizar este resultado dentro de la naturaleza específica del problema que la red está abordando.

En términos de éxitos y desafíos, la precisión del 77% puede ser un indicador prometedor, pero es crucial considerar factores como el equilibrio de clases en el conjunto de datos y las posibles implicaciones prácticas de errores específicos, como falsos positivos o falsos negativos.

También destacamos la oportunidad de mejorar aún más el rendimiento del modelo. La experimentación con diferentes configuraciones de red, funciones de activación y tasas de aprendizaje podría ser el siguiente paso para aumentar la precisión. Además, estrategias como la ingeniería de características y la implementación de técnicas de regularización podrían contribuir significativamente a la robustez del modelo.

En conclusión, una precisión del 77% es un logro prometedor que refleja la capacidad de la red para aprender y generalizar patrones en datos complejos. Sin embargo, la mejora continua y la atención a los detalles específicos del problema son esenciales para llevar el rendimiento del modelo al siguiente nivel.

## Conclusiones

Después de trabajar con Python y las librerías ya mencionadas, evaluamos este entorno como altamente adecuado para abordar problemas relacionados con la construcción y evaluación de modelos predictivos. Las ventajas clave de esta herramienta radican en su capacidad para crear modelos con diversas configuraciones, realizar una selección automática de las combinaciones más efectivas y presentar resultados en varias métricas.

Hemos observado que el redimensionamiento de la matriz para trabajar con imágenes más pequeñas ha conducido a ahorros significativos en tiempo y complejidad en la creación de modelos.

La modificación futura para el sistema OCR podría ser la unión de dos paradigmas que normalmente trabajan separados. Al agregar una red Hopfield para la reconstrucción de los datos de entrada, en este caso el carácter a reconocer, aumentaría el porcentaje de la eficacia del sistema en la etapa de reconocimiento.

El sistema que hemos creado parte de una fase en la que se localizan los caracteres y se centran en una imagen. De forma que partimos de imágenes ya recortadas que facilitan en gran medida el reconocimiento. Pero en la vida real los caracteres no están aislados, sino que se encuentran dentro de textos. Una manera de segmentar este problema es descomponer los textos en pequeñas imágenes que reconozcan solamente un carácter.

Teniendo en cuenta los resultados obtenidos con los modelos de redes Perceptron Multicapa nos parece interesante seguir profundizando en este tipo de algoritmos. Pensamos que puede ser de interés general un estudio comparativo de estos modelos.

## Referencias bibliográficas

- [1] Miralles Pechuán, L., Rosso Pelayo, D., & Brieva, J. (2015). Reconocimiento de dígitos escritos a mano mediante métodos de tratamiento de imagen y modelos de clasificación. *Research in Computing Science*, 93(1), 83–94. <https://doi.org/10.13053/rscs-93-1-7>
- [2] Al Shalabi, H. M., Hasan, M. F., & Ali, A. M. (2005). A New Data Base Scheme Arabic Handwriting Recognition by Hopfield Neural Networks Algorithm. *Journal of Computer Science*, 1(2), 204–206. <https://doi.org/10.3844/jcssp.2005.204.206>
- [3] M. Butaev, M., Yu. Babich, M., Salnikovq, I. I., Martyshkin, A. I., Pashchenko, D. V., & Trokoz, D. A. (2020). Red neuronal para el reconocimiento de escritura a mano. *Nexo Revista Científica*, 33(02), 623–637. <https://doi.org/10.5377/nexo.v33i02.10798>
- [4] Martínez Cortés, Eliana Araujo Mosquera, Fredy Andres. (2020). Algoritmo para la Lectura por medio de Reconocimiento Óptico de Caracteres (OCR) de Etiquetas Nutricionales y la Generación de un Tipo de Sellos Frontales. <http://repositorio.uan.edu.co/handle/123456789/3154>