

# Reconocimiento Óptico de caracteres Números Kanji-Entrega de proyecto

Logreira, José.Limas, Luis.  
{jalogreiraa,lmlimasd}@unal.edu.co  
Universidad Nacional de Colombia-Sede Bogotá

**Resumen**—El reconocimiento de caracteres Japoneses es un campo de acción con múltiples retos por delante. Este trabajo es un acercamiento al reconocimiento de caracteres numéricos Kanji usando las técnicas de procesamiento de imágenes vistas en el curso “Técnicas de Inteligencia Artificial”. La funcionalidad lograda consiste en la toma de una foto de suficiente contraste, de un carácter kanji numérico (se consideraron 14 distintos kanjis) sobre un fondo blanco y donde no existan más caracteres escritos. El programa logra filtrar y pre-procesar la imagen, extraer características, analizarla según los patrones de entrenamiento usados y dar un resultado sobre el tipo de carácter al cual se le tomó la foto.

**Índice de términos**—Adelgazamiento morfológico, enfoques dinámicos, Kanjis, extracción de características, segmentación.

## I. INTRODUCCIÓN

El lenguaje japonés posee una escritura complicada, existen 3 tipos de caracteres: el hiragana (para palabras propias del japonés), el katakana (para palabras extranjeras) y el Kanji (caracteres de origen chino que expresan unas ideas, conceptos y números) [1]. Entre estos tipos de escritura el reconocimiento de Kanjis es el que más se dificulta. Su escritura consiste en muchos trazos situados en posiciones específicas con relación al resto de trazos, además es una estructura recursiva (Kanjis formados por otros Kanjis) [2]. Esto genera, a diferencia de otros caracteres alfabéticos que los Kanjis sean complicados en forma, existan en gran número (en el japonés se aceptan 2136 como uso cotidiano) y las diferencias de unos con otros sean muy leves.

Como limitante del proyecto se consideró tener en cuenta únicamente el reconocimiento de símbolos numéricos, los cuales son mostrados en la tabla siguiente. Se ha desarrollado como proyecto para el curso de Técnicas de Inteligencia Artificial, una aplicación capaz traducir Kanjis usados para numeración (números Kanji-Kanji numéricos), siendo el computador capaz de interpretar 14 símbolos distintos que representan la numeración estándar kanji.

Número	Símbolo	Pronunciación
0	〇	Rei, zero
1	一	Ichi
2	二	Ni
3	三	San
4	四	Yon
5	五	Go
6	六	Roku
7	七	Nana
8	八	Hachi
9	九	Ku
Decimas	十	Yuu
Centésimas	百	Hyaku
Miles	千	Sen
10 Miles	万	Man

Las restricciones consideradas para las imágenes a analizar son básicamente un tamaño adecuado de las imágenes no menor a 32x32 pixeles, y una escritura en tinta negra sobre fondo blanco.

## II. REVISIÓN DEL ESTADO DEL ARTE

Algunos de los retos descritos en la literatura al respecto de la caracterización de kankis son su gran cantidad de categorías, complejidad de las formas a identificar, existencia de caracteres muy similares, gran variedad de deformaciones en la escritura, etc. Es claro que este trabajo no busca desarrollar algoritmos de reconocimiento que sean invariantes al tipo de letra o distorsión de la escritura; estos objetivos quedaron por fuera del alcance del proyecto.

Similarmente, diversos problemas de la escritura a mano alzada que le añaden complejidad a la identificación fueron dejados de lado; tales son los problemas de distorsión y transformaciones de la escritura, o incluso problemas de entrenamiento que requieren clasificadores con bases de datos inmensas no fueron considerados. Todos estos problemas son mencionados en la literatura permitiendo ver la complejidad del problema del reconocimiento.

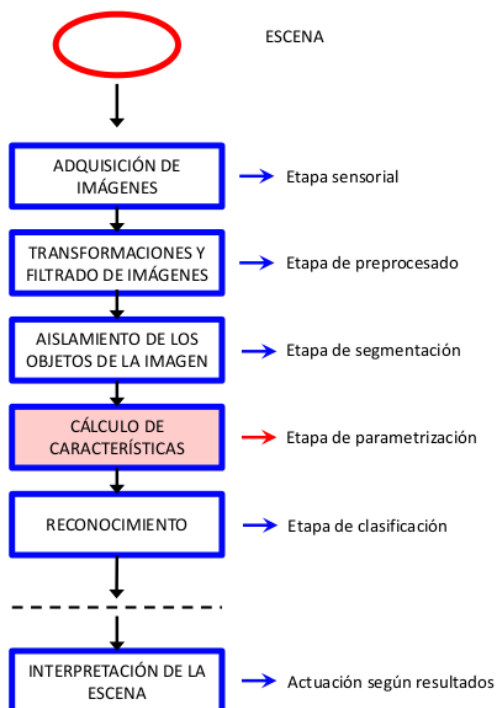
La literatura también menciona métodos de clasificación que usa métodos poco refinados o “toscos” para realizar la preclasificación de caracteres basado en medición y relación de distancias. Este enfoque es usado en este trabajo para la extracción de una buena cantidad de características.

Otro enfoque de clasificación está en el uso de redes neuronales, que funcionan bien aunque no en todos los casos, siendo especialmente problemáticas cuando la escritura posee muchas distorsiones, el ancho de línea es muy cambiante en los caracteres, o las líneas son difusas. No considerando estos problemas, las redes neuronales son una alternativa atractiva para los clasificadores. Estas son probadas en este trabajo, consiguiendo resultados bastante favorables.

Para evitar problemas del ancho de línea se usan algoritmos de adelgazamiento que permiten extraer la estructura del kanji de manera sencilla, y con base a ella se puede trabajar la extracción de características. Así, se implementaron algunos algoritmos de extracción del esqueleto para probar su desempeño, obteniendo resultados no muy favorables en cuanto a la extracción de estructuras que no representaban fielmente el kanji.

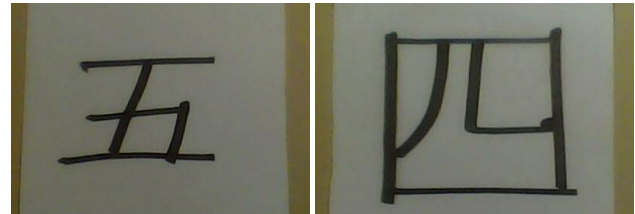
### III. DESARROLLO DEL PROYECTO

Los pasos para la construcción de un sistema de reconocimiento de imágenes se describen gráficamente en la figura siguiente. Se expone a continuación lo desarrollado en cada uno de ellos en este proyecto.



#### • Adquisición de Imágenes

Como ya se ha mencionado, varias restricciones se han puesto a la adquisición de la imagen, tales como un tamaño adecuado, mínimo de 32x32 píxeles, símbolos de color negro escritos sobre fondo blanco, e imágenes que contengan un único carácter a reconocer, no múltiples caracteres. A continuación se muestran algunas de las imágenes sin procesar usadas para este proyecto.

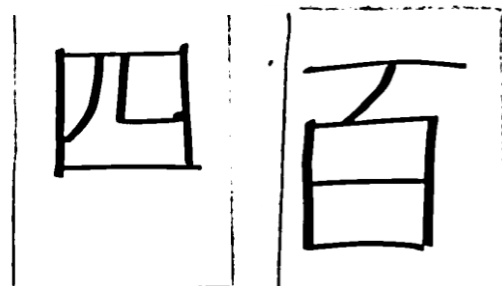


Las imágenes fueron tomadas con cámara digital, tamaño de imagen de 1 megapixel e iluminación día. Más adelante se verá que la iluminación es un factor que afecta fuertemente los procesos de filtrado, y que por tanto es necesario tener un control sobre la iluminación a usar, tal que no exista ningún trazo difuso en la escritura del carácter.

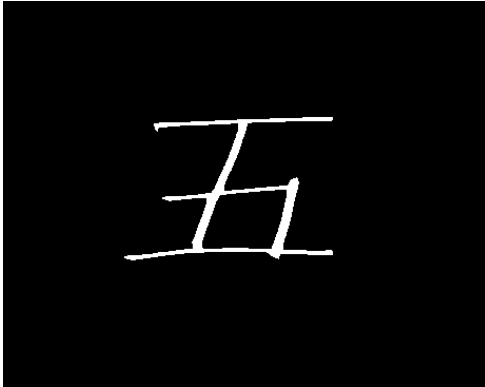
#### • Transformaciones y Filtrado

El filtrado aplicado a las imágenes fue un proceso de prueba y error, en el cual la secuencia de filtros de suavizado, convolución, conversión de espacios de color, etc. fue definida tras múltiples pruebas y usando diferentes imágenes. El objetivo de este paso es lograr que el carácter a extraer quede totalmente filtrado del resto de la imagen, es decir, que al aplicar la umbralización de la imagen para trabajar con imágenes binarias, la imagen solo contenga información del carácter a analizar y no de los bordes de la hoja o ruido de la imagen.

Tras evaluar el desempeño de la umbralización adaptativa, que inicialmente fue la opción preferida, se concluyó que esta no era óptima para nuestro propósito pues, en el intento de usar tal tipo de umbralización para discriminar bordes no pertenecientes al carácter pero con intensidad de grises similar, no solo resaltaba el carácter deseado sino también el ruido, bordes indeseados y demás cosas que se debían eliminar. Una imagen que ilustra los resultados obtenidos con el filtro adaptativo se muestran a continuación.



Por el contrario, cuando se usa un filtro de umbralización simple, donde los parámetros del umbral son escogidos una vez más por prueba y error, se logran mejores resultados logrando filtrar más adecuadamente los bordes indeseables. Esto se evidencia en las imágenes que se presentan a continuación.



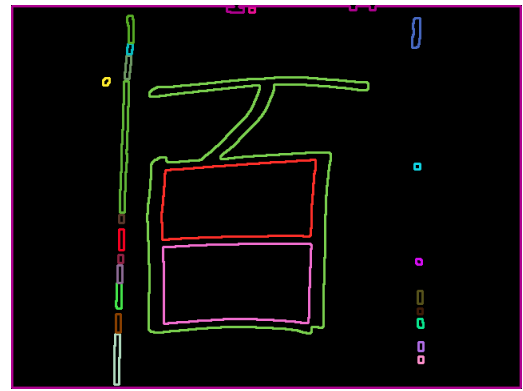
Los pasos que se siguieron para llegar hasta este resultado fueron los siguientes:

- Filtro bilateral a imagen en el dominio RGB con parámetros sigmaColor de 50, lo que quiere decir que considera una vecindad de 50 pixeles a la redonda del pixel que analiza (elemento estructurante), y parámetro sigmaSpace de 10 que quiere decir que 10 pixeles a la redonda afectan el nuevo valor del pixel analizado según su tonalidad.
- Conversión a espacio de color escala de grises: Cambiar la imagen del espacio de color RGB a escala de grises.
- Umbralización binaria invertida con un umbral establecido en 65/255 (para iluminación día).
- Abertura y Cierre del resultado con un elemento estructurante de tamaño 5x5.

Aunque otra posibilidad para realizar la umbralización es convertir la imagen RGB en el espacio de color HSI, en el cual el símbolo es claramente resaltado y fácilmente filtrado. Esto no se realizó debido a que los resultados obtenidos fueron satisfactorios con la conversión a escala de grises.

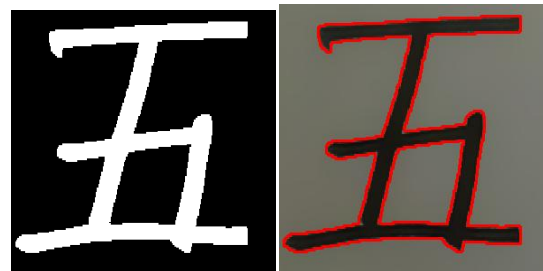
#### • Aislamiento de los objetos de la imagen

Para el aislamiento de los objetos, se procede a realizar la segmentación de la imagen usando la funcionalidad de OpenCV para este propósito. La importancia de haber obtenido únicamente la extracción del carácter en el paso anterior cobra importancia aquí en cuanto que no se desea segmentar imágenes que tengan algún tipo de ruido pues este sería considerado como un segmento de la imagen y no permitiría la lectura adecuada de los segmentos del carácter a identificar. Una imagen que muestra los segmentos herrados de bordes indeseados es la siguiente.



Como es sabido, la segmentación es útil para analizar los contornos que surgen de la imagen filtrada, y de ellos extraer la mayor cantidad de características posibles. Para ello es necesario asegurarse que los contornos analizados efectivamente pertenezcan a la imagen. Por tanto, es preferible en este punto del proceso, realizar un recorte de la imagen donde se descarte toda la porción de la imagen que no interesa, y se quede con la porción de la imagen que es útil. Esto se logra identificando el primer contorno (el más externo) del símbolo y construyendo una cuadrícula al rededor de éste para luego extraer únicamente tal cuadrícula.

La cuadrícula es construida al especificar las coordenadas de la imagen, usando las coordenadas provistas por la función cvBoundingRect al hallar la cuadrícula que rodea el contorno más externo. Así, se obtiene una imagen como la que se encuentra a continuación:



#### • Cálculo de características

Dentro de las características usadas para la identificación, hay varias que aportan una información valiosa a la identificación, mientras que otras no tanto. Esta clase de relaciones se determinan de acuerdo al análisis por componentes principales. En él se muestra de qué manera algunas características son más relevantes que otras en la clasificación, y por tanto algunas de ellas se pueden descartar.

A continuación se describen algunas de las características usadas para la identificación del carácter:

- Número de trayectorias cerradas halladas en la segmentación: Es una de las características más representativas para este tipo de proyectos. Permite descartar inmediatamente algunos caracteres siendo esta característica muy discriminativa.

- Área de contorno: Puesto que fue posible normalizar la imagen y compararla con un patrón previamente establecido, el área del contorno externo se convierte en uno de los parámetros más decisivos.
- Momentos del contorno: centralizados y normalizados. Estos momentos no son invariantes a escala o rotación, sin embargo la evaluación de influencia de características mostró que esta aportaba ligeramente a la discriminación para la clasificación.
- Momentos invariantes de Hu: El análisis de correlación entre los momentos de Hu de distintas muestras de una misma clase muestra que únicamente fue útil el primer momento de Hu, puesto que los otros 6 momentos daban valores bastante pequeños, del orden de  $10^{-9}$ . Estos valores eran particularmente problemáticos para lograr una comparación entre los mismos, por lo cual el único momento usado fue el primer momento de Hu.
- Compacidad: Permite relacionar fuertemente la forma del kanji con su área interna. Este parámetro es fuertemente discriminante entre los caracteres.

#### • Reconocimiento

El reconocimiento del carácter se realiza usando una red neuronal como clasificador. La red neuronal ha sido escogida según métodos heurísticos hallados en la literatura, y su entrenamiento se ha realizado usando el método de backpropagation.

Los heurísticos tenidos en cuenta son:

- Para Redes Neuronales con entrenamiento no supervisado es recomendable que la primera capa de neuronas sea del mismo tamaño que el número de nodos de entrada. Este no es el caso de nuestro proyecto, sin embargo, nos es provechoso para efectos de comparación.
- Para redes neuronales con entrenamiento no supervisado se elige la primera capa con un número de neuronas inferior al número de nodos de entrada, típicamente  $2/3$  del número de nodos de entrada.
- EL número de neuronas de salida se relaciona con la flexibilidad de entrenamiento de la red neuronal. Mayor número de neuronas de salida que de clases es innecesario y reduce el desempeño de la red neuronal.
- Para redes neuronales usadas en reconocimiento de caracteres, se usan típicamente 3 o 4 capas: la capa de nodos de entrada, la primera capa oculta, una segunda capa oculta opcional, y la capa de neuronas de salida que son típicamente el número de clases. La cuarta capa de neuronas es estimada típicamente como la mitad de la suma de las neuronas de la primera capa y de la última capa (de salida). La elección entre 3 o 4 capas es una decisión de diseño muy específica para cada aplicación, y que además repercute en el número de iteraciones para el entrenamiento y en la exactitud de los resultados esperados.

Basados en todo lo dicho, la arquitectura de la red neuronal se ha elegido de la siguiente manera:

- Se eligen 3 capas para la red neuronal: Los nodos de entrada de las características y 2 capas más, típicas para la resolución de problemas no lineales.
- Número de neuronas en la primera capa oculta igual al número de características de entrada.
- Número de neuronas en la segunda capa de entrada (la capa de salida) igual al número de características.
- Rata de convergencia del error: 0,1.
- Umbral de convergencia: 0,001
- No máximo de iteraciones: 600

La base de datos usada para el entrenamiento de la red neuronal fue consolidada usando tanto imágenes halladas en la web como fotos tomadas con muestras propias. Así, se construyó una base de datos de 20 muestras para cada clase.

#### IV. QUÉ FALLÓ

El proyecto se logró completar hasta el proceso de extracción de características, logrando identificar cuáles de estas son las más representativas para la discriminación entre caracteres. Sin embargo, en el proceso de entrenamiento de la red neuronal propuesta, se presentaron múltiples problemas referentes al tamaño máximo que pueden adquirir los vectores de entrenamiento y de prueba de la red. Esto se tradujo en la imposibilidad de construir una red neuronal con muchas muestras de entrenamiento o con muchas clases a entrenar.

También se presentaron problemas de incompatibilidad entre datos de funciones usadas. Pero por sobre todo, el principal inconveniente para finalizar satisfactoriamente el proyecto fue la falta de tiempo por parte de los integrantes del grupo.

Actualmente la red neuronal funciona correctamente con 3 clases (es decir, 3 tipos de kanji) teniendo un buen desempeño (los resultados no se muestran en este trabajo).

Como recomendaciones a futuro se decide trabajar enteramente en lenguaje C++ en vez del tradicional y ya no tan respaldado C. La clase de problemas que se presentaron se se pueden evitar fácilmente con el cambio de lenguaje de programación.

#### V. CONCLUSIONES

A pesar de no concluir satisfactoriamente el proyecto, la realización de éste contribuye a la consolidación de conceptos y habilidades aprendidas en clase. Los diversos resultados parciales obtenidos en este proyecto son de gran valor práctico por permitir un acercamiento más concreto a los procesos de construcción de sistemas de reconocimiento de patrones.

Se realizó un trabajo metódico para determinar cuáles eran las características más relevantes para ser tenidas en cuenta. En

este proceso se realizaron múltiples pruebas de comparación de características que permitió reducir el número de características inicialmente definido a un número reducido de 6, que pese a que no fue posible el entrenamiento del clasificador, se evidencia que son características fuertemente discriminantes y bastante útiles. Este tipo de conclusiones son las que permiten la implementación de clasificadores sencillos como redes neuronales pequeñas, que no requieren de gran costo de procesamiento en tiempo.

## VI. REFERENCIAS

- [1] T. Wakahara and Y. Kimura, "Toward robust handwritten Kanji character recognition," *Pattern Recognition Letters*, vol. 20, no. 10, pp. 979–990, Oct. 1999.
- [2] T. Wakahara, A. Suzuki, N. Nakajima, S. Miyahara, and K. Odaka, "On-line cursive Kanji character recognition as stroke correspondence problem," in *Proceedings of the Third International Conference on Document Analysis and Recognition, 1995*, 1995, vol. 2, pp. 1059–1064 vol.2.
- [3] F. Togawa, T. Ueda, T. Aramaki, and A. Tanaka, "Receptive field neural network with shift tolerant capability for Kanji character recognition," in *1991 IEEE International Joint Conference on Neural Networks, 1991*, 1991, pp. 1490–1499 vol.2.
- [4] K. Yamada, "Non-uniformly sampled feature extraction method for kanji character recognition," in *Proceedings of the Fourth International Conference on Document Analysis and Recognition, 1997*, 1997, vol. 1, pp. 200–205 vol.1.
- [5] B. Zhu and M. Nakagawa, "A robust method for coarse classifier construction from a large number of basic recognizers for on-line handwritten Chinese/Japanese character recognition," *Pattern Recognition*, vol. 47, no. 2, pp. 685–693, Feb. 2014.
- [6] B. Chen, B. Zhu, and M. Nakagawa, "Training of an on-line handwritten Japanese character recognizer by artificial patterns," *Pattern Recognition Letters*, vol. 35, pp. 178–185, Jan. 2014.
- [7] K. Nakayama and T. Kato, "A weighted competitive learning method extracting skeleton pattern from Japanese Kanji characters," in *1994 IEEE International Conference on Neural Networks, 1994. IEEE World Congress on Computational Intelligence, 1994*, vol. 7, pp. 4237–4242 vol.7.
- [8] S. Senda, M. Minoh, and I. Katsuo, "A fast algorithm for the minimum distance classifier and its application to Kanji character recognition," in *Proceedings of the Third International Conference on Document Analysis and Recognition, 1995*, 1995, vol. 1, pp. 283–286 vol.1.
- [9] E. Ishidera and A. Sato, "A candidate reduction method for handwritten Kanji character recognition," in *Sixth International Conference on Document Analysis and Recognition, 2001. Proceedings, 2001*, pp. 8–13.

## Autores

Luis miguel limas Duque

Estudiante de ingeniería Electrónica

Facultad de Ingeniería Universidad Nacional de Colombia  
Sede Bogotá

José Alejandro Logreira Ávila

Estudiante de ingeniería Electrónica

Facultad de Ingeniería Universidad Nacional de Colombia  
Sede Bogotá