Segmentación del plano de texto en imágenes deterioradas

Luis Gerardo de la Fraga §, Jorge Martínez Carballido ‡

§ Centro de Investigación en Ingeniería y Ciencias Aplicadas,
Universidad Autónoma de Morelos. Ave. Universidad 1001, 62210 Cuernavaca Mor.
E-mail: fraga@inaoep.mx

‡Grupo de Ingeniería en Sistemas de Computadoras. Instituto Nacional de Astrofísica, Óptica y Electrónica
Luis Enrique Erro No. 1. Tonantzintla, Puebla, 72840
E-mail: jmc@inaoep.mx

Resumen

La extracción de caracteres en imágenes de documentos que tienen mezcladas letras con pinturas, sombras, manchas y zonas borrosas, es una operación frecuente. Resulta crítica en al análisis de imagen, el reconocimiento óptico de caracteres, en transmisión de imágenes y sistemas de videoconferencia. En este trabajo presentamos un esquema diferente de un algoritmo ya propuesto y se presenta un algoritmo nuevo y apropiado para automatizarse. También se presentan resultados comparativos con los algoritmos existentes [1, 2, 3, 4].

Palabra clave: Procesado de imagen, Segmentación, Reconocimiento Óptico de Caracteres.

1 Introducción

Los documentos han sido el medio de información primario en nuestra sociedad. La disponibilidad de computadoras, terminales de video, discos ópticos, máquinas de fax, impresoras láser y scaners hace que los documentos puedan ser generados, transmitidos, accesados, almacenados, desplegados, impresos y reabsorbidos automáticamente. Como sustituto al tratamiento manual, existen sistemas automáticos de análisis de imágenes de documentos que permiten reconocer los símbolos sobre los documentos y organizarlos en otro tipo de archivos. Estos sistemas automáticos van desde programas de OCR (Optical Character Recognition) [5] hasta sistemas especiales que incluyen el reconocimiento automático de patrones [6] y sistemas gráficos de vectorización para dibujos en ingeniería [7].

La aplicación de tales sistemas está grandemente limitada por sus requerimientos de documentos de buena calidad. Algunos problemas que contribuyen a tener una calidad pobre son: 1) caracteres brillantes de varios colores, 2) caracteres manchados o borrosos, 3) cambios no uniformes en los colores del documento original dado su gran tiempo de almacenamiento, 4) se presentan sombras generadas por las condiciones de iluminación, 5) el documento presenta una calidad pobre de escritura o impresión, 6) existe un solo documento original deteriorado imposible de reproducir, y 7) Se tiene texto combinado con un fondo uniformes de patrones geométricos o con un fondo no uniforme de pinturas, tales como formas oficiales, cheques bancarios, pagarés, etc.

Dadas las grandes variaciones que se tienen en los casos arriba expuestos, aunándose a la gran variedad de formas de los caracteres de texto: distintos tipos, tamaños y colores; todavía se requiere mucho esfuerzo para encontrar soluciones generales a estos problemas.

El objetivo de este trabajo fue tener un método adecuado para extraer la forma de los caracteres en imágenes gráficas en tonos de gris, de manera que sea más fácil el reconocimiento óptico de caracteres que el logrado con la imagen original deteriorada.

De esta forma se desarrollaron dos nuevos algoritmos que cumplen nuestro objetivo: 1) Un algoritmo modificado por nivel lógico [1], y 2) El algoritmo por filtro de formas. Ambos algoritmos están basados en la aplicación de máscaras, o en otras palabras, en la aplicación de filtros espaciales.

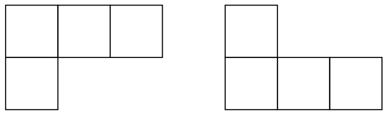


Figura 1: Primitivas para el filtro de formas

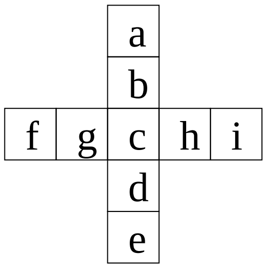


Figura 2: Máscara del filtro de formas

2 El algoritmo por nivel lógico

Este algoritmo consta de dos etapas: 1) Una binarización por promedios de la imagen de entrada en 256 tonos de gris y 2) un filtro lógico. El algoritmo toma ventaja de la suposición de que los caracteres, en la mayoría de los escritos, son más obscuros que el fondo.

El paso 1) de binarización consta de lo siguiente. Sea $f(x, y)$ la imagen de entrada. Una imagen de salida, $b_1(x, y)$, se genera con un operador de máscara de tamaño K×K que calcula el promedio alrededor de cada punto (x, y) sobre la imagen f. El pixel (x, y) en la imagen de salida b_1 tomara el valor de 1 si la diferencia entre el promedio calculado alrededor del punto (x, y) y el mismo valor del pixel es mayor que un predeterminado umbral T, en caso contrario b(x,y) tomará el valor de 0.

El paso 2) se aplica debido a que la binarización, en imágenes donde los caracteres están pobremente definidos o se tiene ruido, produce unos trazos discontinuos. Se tiene una máscara de tamaño 3×3, cada pixel, dentro de la máscara, queda representado por las letras abc, def y fgh. Una nueva imagen b_2 tomará el valor de 1 si se cumple H —una nueva H distinta a la definida en [1]— en la ecuación Booleana [8] (1).

$$\begin{aligned} H &= abd + bcf + fhi + dgh \\ &= b(ad + c) + h(fi + dg) \end{aligned} \tag{1}$$

3 El algoritmo por filtro de formas

Este algoritmo sólo requiere un parámetro, el tamaño de la ventana, K, para una primera etapa de binarización por promedios, como se explicó en la sección anterior, pero en este caso el valor del parámetro T se hace igual a cero.

La segunda etapa de este algoritmo es un *filtro de formas*, que consiste en lo siguiente: empíricamente se hace que los trazos de los caracteres se compongan se dos “primitivas”, ilustradas en la Fig. 1. Entonces, los trazos se suponen compuestos por la agrupación de todas las ocho formas rotadas de estas primitivas. La aplicación del filtro de formas se hace usando la máscara mostrada en la Fig. 2. Tomando como valores lógicos el valor de los pixels resultantes de la binarización, el punto central “c” se considera parte del trazo si H, en la Ec. (2), es verdadera.

$$\begin{aligned} H &= abcg + abch + fgcb + fgcd + \\ &\quad + edcg + edch + edch + ihcb + ihcd \\ &= c[(g + h)(ad + ed) + (b + d)(fg + hi)] \end{aligned} \tag{2}$$

4 Resultados

4.1 Prueba cualitativa

Se aplicó a la imagen de la Fig. 3(a) 1) el algoritmo por detección de bordes, originalmente descrito en [2] (ver Fig. 3(b)); 2) El algoritmo por nivel lógico descrito en [1]; (ver fig 3(c)); 3) El algoritmo nuevo descrito en la sec. 1; (ver fig 3(d)); 4) El algoritmo nuevo por filtro de formas descrito en la sec. 2; (ver fig 3(e)); y el algoritmo YDH [3].

4.2 Prueba cuantitativa

Usando el programa de OCR [5] se obtuvieron los resultados de la Tabla siguiente:

Figura	Número de caracteres reconocidos visualmente n_b	Número de caracteres reconocidos por el OCR n_r	Razón de caracteres extraídos $C_E = n_b/n$	Razón de caracteres reconocidos $C_R = n_r/n$
3(b) Por detección de bordes	107	56	0.465	0.243
3(c) Por nivel lógico	229	184	0.996	0.800
3(d) Nuevo nivel lógico	229	199	0.996	0.865
3(e) Por filtro de formas	229	192	0.996	0.834
3(f) YDH	227	184	0.987	0.800

Tabla 1: El número de caracteres legibles es n=230

5 Conclusiones

En este trabajo se han revisado tres algoritmos, y se proponen otros dos nuevos, para la segmentación de imágenes documentos, en 256 tonos de gris, que presentan mala calidad. Los algoritmos separan las posibles letras del fondo y dan como resultado imágenes binarias. Este proceso es un paso previo para el reconocimiento óptico de caracteres. También se pueden utilizar estos algoritmos para la comprensión de imágenes de documentos para bases de datos y/o transmisión de los mismos.

De las pruebas efectuadas con el OCR, los dos algoritmos nuevos propuestos, el nuevo nivel lógico y el algoritmo por filtro de formas, son los mejores. El algoritmo YDH [3] y el algoritmo por nivel lógico propuesto en [1] se consideran buenos. El algoritmo por detección de bordes se considera malo.

Los dos algoritmos nuevos requieren menos parámetros para su aplicación: el algoritmo nuevo por nivel lógico requiere solo dos parámetros; el algoritmo por filtro de formas requiere solo uno. En comparación, el algoritmo por nivel lógico propuesto en [1] requiere tres parámetros, y el algoritmo YDH necesita cinco. Además, los dos algoritmos nuevos aquí propuestos comparten un parámetro que no resulta crítico: el ancho de la ventana donde se toman los promedios en la vecindad de un pixel.

Referencias

[1] M. Kamel and A. Zhao. Extraction of binary character/graphics images from grayscale document images. *Graph. Models and Image Proc.*, 55(3):203–217, 1993.

[2] R.C. Gonzalez and R.E. Wood. *Digital Image Processing.* Addison-Wesley, 1993.

[3] Y. Yasuda, M. Dubois, and T.S. Huang. Data compression for check processing machine. *Proc. IEEE*, 68(7):874–885, 1980.

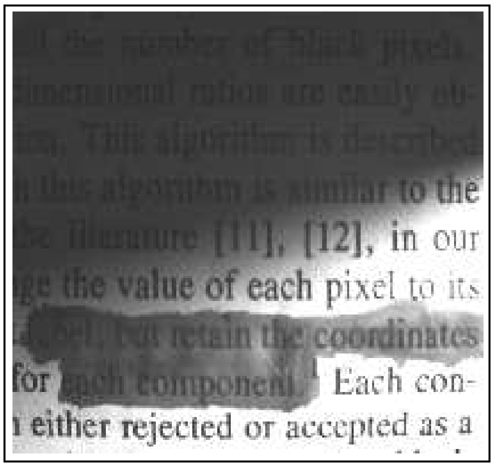
[4] F.M. Wahl, K.Y. Wong, and R.G. Casey. Block segmentation and text extraction in mixed text/image documents. *Comp. Vis. Graph. & Image Proc.*, 20:375–390, 1982.

[5] SZXI Recognita Corp. *Recognita Plus. The American/English OCR User's Guide. Version 2.1.*

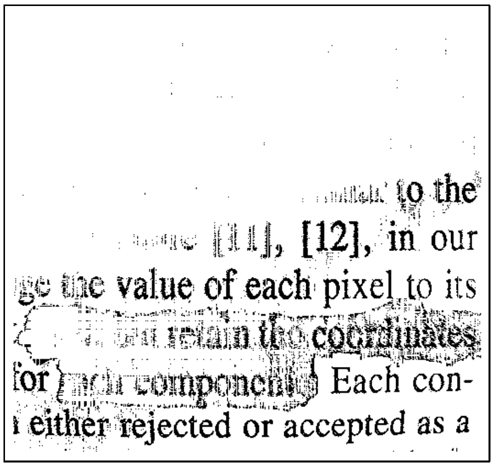
[6] S. Suzuki and T. Yamada. Maris: Map Recognition Input System. *Pattern Recognition*, 23:919–933, 1990.

[7] R. Kastuni and J. Alemany. Information extraction from image of paper-based maps. *IEEE Trans. Software Eng.*, 14:671–675, 1988.

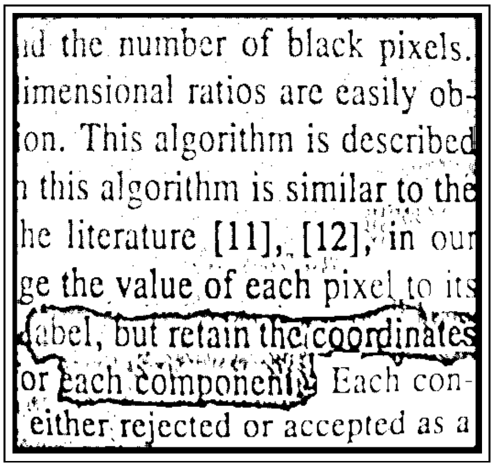
[8] M. M. Mano. *Diseño Digital.* Prentice-Hall, 1987.



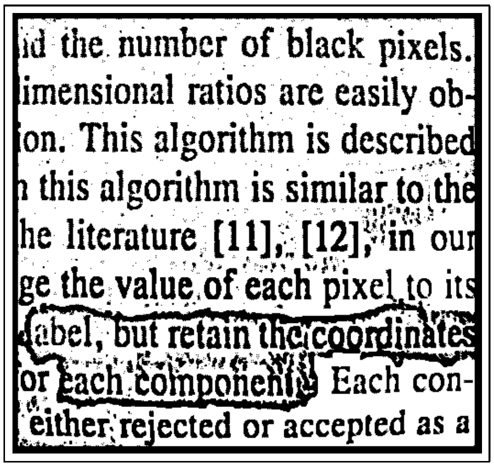
(a) Imagen original



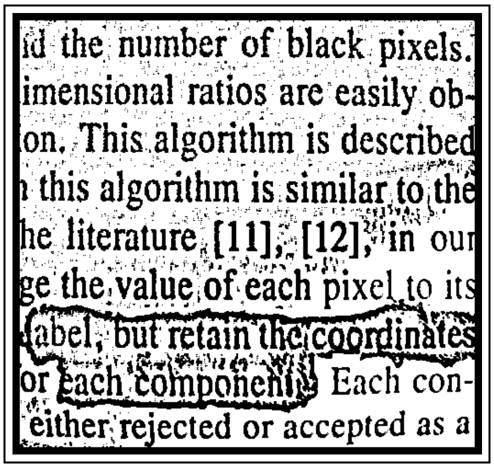
(b) Algoritmo por detección de bordes



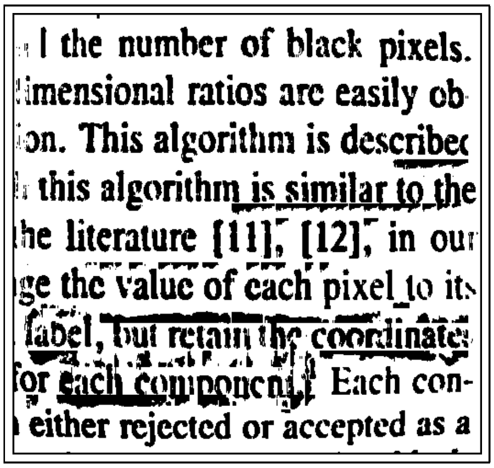
(c) Algoritmo por nivel lógico original



(d) Nuevo nivel lógico



(e) Por filtro de formas



(f) Algo. Yasuda-Dubois-Huang (YDH)

Figura 3: a) Imagen original deteriorada. b-f) Imágenes binarias resultantes de aplicar varios algoritmos para obtener el plano de caracteres. d) y e) son el resultado de los nuevos algoritmos propuestos es este trabajo.