
SEGMENTACIÓN Y MEJORA DE DOCUMENTOS

MEMORIA DE LA PRÁCTICA

Héctor Padín Torrente

Visión Artificial - Grupo 3.3

Universidade da Coruña

December 2022

Índice

1. Introducción	3
2. Detección de Contornos	4
3. Sustracción de imperfecciones en el documento	5
4. Segmentación del texto	6
5. Resultados	6

1. Introducción

Este proyecto propone una solución basada en técnicas de visión por computador para la segmentación y mejora de documentos. El algoritmo tan solo necesita como entrada el nombre del documento o documentos a segmentar, opcionalmente también permite ver los pasos intermedios que va realizando para su objetivo.

Primeramente, se intentarán detectar los contornos del documento. Previamente se realiza un suavizado para aplicar un algoritmo de realce de bordes (*Unsharp Masking*), después utilizamos la salida del algoritmo de *Canny* sobre la imagen realizada para aplicar un algoritmo de detección de contornos, y sobre este aplicar unas técnicas de filtrado y mejora de contornos, que nos permitirá quedarnos con el que mejor represente los bordes del folio.

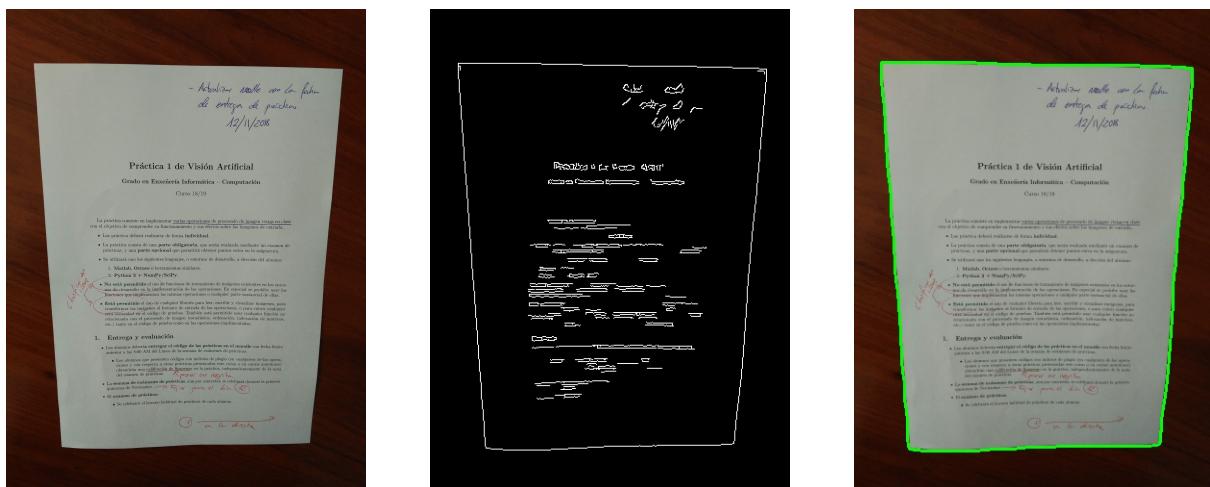


Figura 1: Pasos intermedios de la detección de contornos.

Una vez tenemos los contornos, realizamos una serie de transformaciones sobre la imagen para hacer que las esquinas del folio coincidan con las de esta. Después, creamos una máscara con el rango de colores a eliminar (en este caso azul y rojo) para después poder realizar una segmentación más correcta del texto. En la práctica no se eliminan los colores, si no que se computa un *K-Medias* sobre la imagen para detectar el color dominante de la imagen y así reemplazar estos colores eliminados.

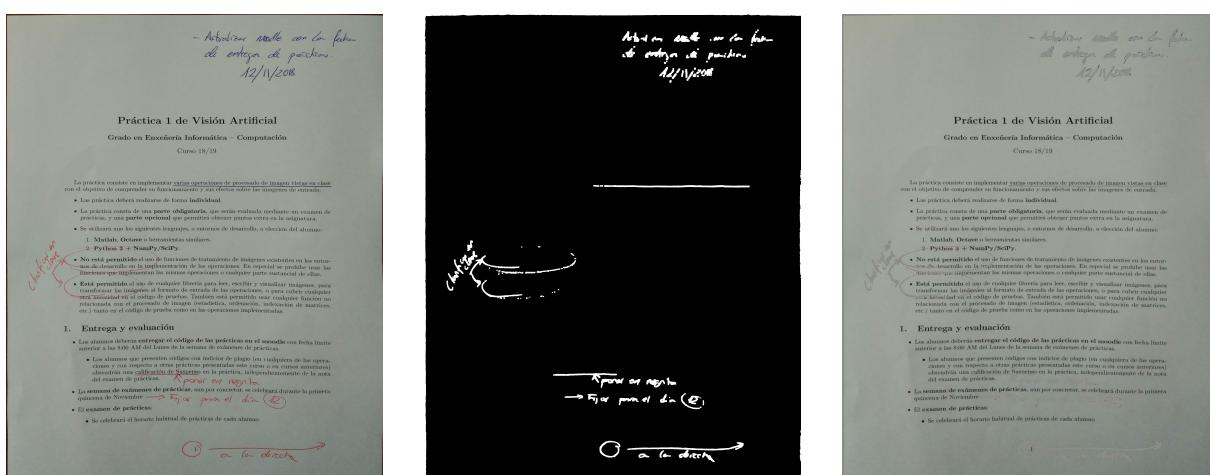


Figura 2: Pasos intermedios de la eliminación de marcas.

Una vez hemos eliminado las marcas de bolígrafo, realizamos un *Threshold* adaptativo de la imagen para realizar una mejor segmentación del texto. Sobre la imagen resultante, extraemos el rango de negros que representan las letras del texto y realizamos una erosión (eliminación de ruido). Una vez segmentado se reconstruye sobre un fondo blanco y se realiza un *threshold* binario.

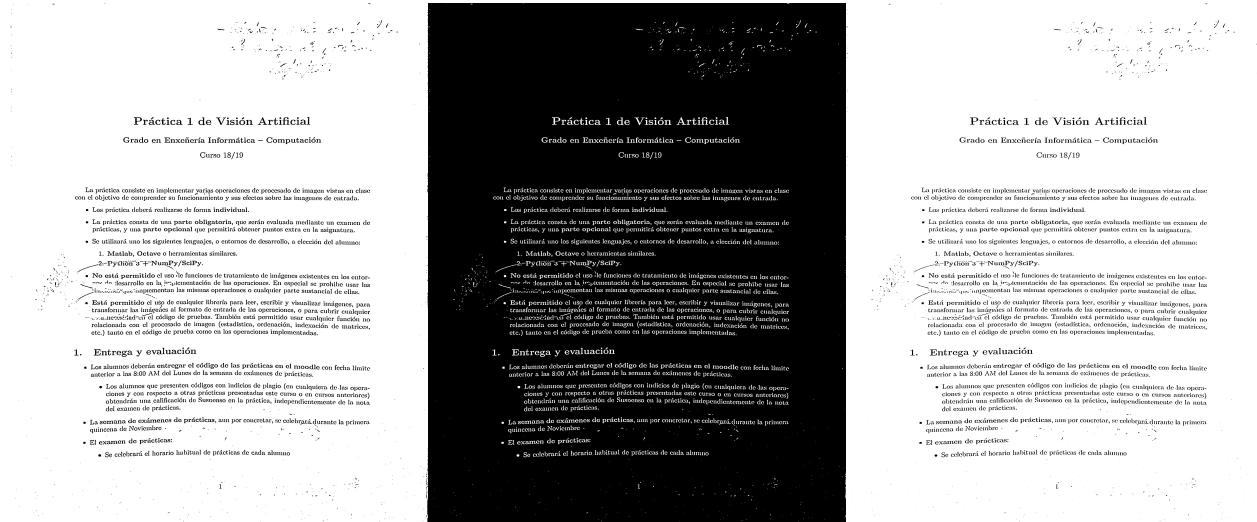


Figura 3: Pasos intermedios de la segmentación de texto.

2. Detección de Contornos

Para la detección de contornos nos encontramos con tres problemáticas principales:

- Documentos blancos sobre fondo blanco.
- Iluminación variante.
- Fondo con patrones con alto contraste o ruído.

Las dos primeras problemáticas se deben a un bajo contraste entre el folio y el fondo, esto se puede solucionar con un algoritmo de *Unsharp Masking*. El problema radica en que cuanto más se realzan los bordes, peor se segmentan documentos con ruido. Por lo tanto se ha optado por una técnica intermedia entre ambos, que sigue la siguiente fórmula:

$$\text{Blurred}(x, y) = I(x, y) * (I(i, j) - N(i, j)) \quad (1)$$

$$\text{Sharpen}(x, y) = \alpha I(x, y) - \beta \text{Blurred}(x, y) \quad (2)$$

$$\alpha = 1,3, \beta = 1,0 \quad (3)$$

Básicamente lo que realizamos es un suavizado de ruido gausiando, en donde calculamos una estimación del ruido en cada vecindario, y se lo restamos a la imagen original. Después, volvemos a realizar el mismo tipo de suavizado y se lo pasamos al algoritmo de *Canny*. Una vez se realiza la detección de bordes, se le pasa al algoritmo *Suzuki-85*, que nos devuelve todos los contornos detectados en la imagen.

A continuación aplicamos el algoritmo de *Ramer-Douglas-Peucker* para aproximar cada contorno detectado a una forma poligonal, de cuatro vértices con un área mínima determinada. El algoritmo previamente mencionado solo funciona bien con contornos bien definidos, o que apenas existe discontinuidad entre estos. En caso de no conseguir unos bordes bien definidos, debido a que el contraste no es lo suficientemente bueno o hay un cambio de iluminación muy fuerte o el fondo tiene mucho ruido, el algoritmo no segmentará correctamente los bordes del folio.

Si esto ocurre, los contornos detectados son muy discontinuos, se puede solventar mediante una técnica basada en la estimación entre la distancia entre el mínimo y máximo punto de cada contorno (esquina superior izquierda y esquina inferior derecha). Obviamente para que funcione correctamente deben detectarse estas dos esquinas, de una forma relativamente precisa. Lo que se realiza es una búsqueda entre los contornos detectados, en los que la unión de su máximo y su mínimo se pueda aproximar mediante un polígono que tenga un área lo suficientemente grande como para representar los bordes del documento.

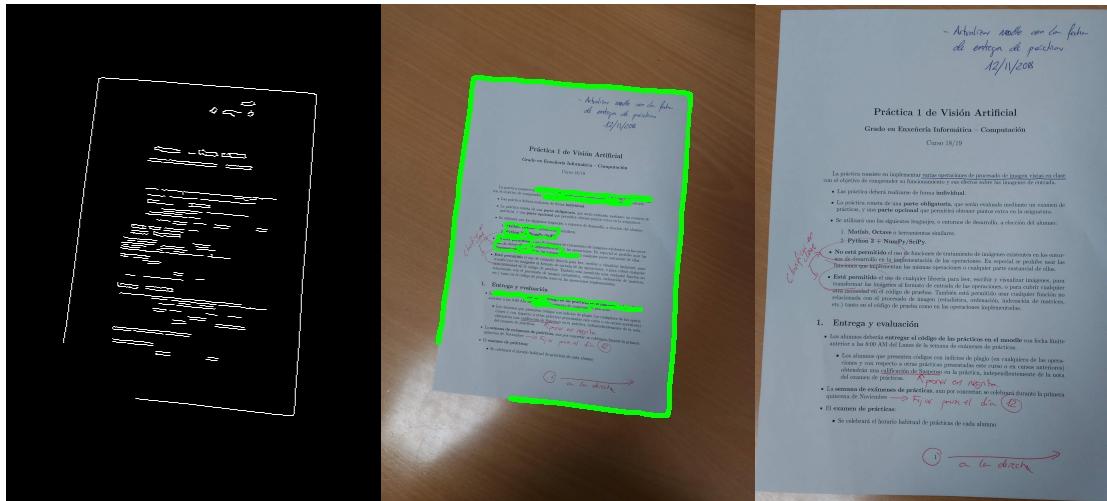


Figura 4: Seguimiento de contornos.

En el caso de no cumplir ninguno de los criterios, se 'recorta' la imagen un 15 % por cada eje para conseguir una especie de 'Zoom' y poder proseguir para la segmentación de texto.

Por último, se aplica una interpolación con las dimensiones de la imagen original para transformar los píxeles interiores de contornos trapezoidales en un rectángulo, para permitir una mejor visualización del documento.

Se han probado distintas alternativas a este método de detección de contornos, tal como la previa binarización de la imagen para despues extraer contornos, otras técnicas de realce de bordes, o algoritmos para el seguimiento de contornos. Todas ellas sin mucho éxito.

3. Sustracción de imperfecciones en el documento

Sobre los documentos del conjunto de test se detectaron ciertas imperfecciones en estos tales como:

- Manchas de café.
- Marcas de bolígrafo.
- Arrugas en el documento y pliegues.

Para solventar las problemáticas de manchas de café y pliegues se aplica un *Threshold* adaptativo de la imagen, despues de haber eliminado las marcas de bolígrafo, que nos permite eliminar todas estas imperfecciones sin nungún problema.

La dificultad radica en la previa extracción de marcas de bolígrafo ya que el bolígrafo azul es difícil de eliminar en ciertas imágenes debido a que tiene mucha similitud con el color negro, esto también ocurre con el rojo, ya que dependiendo de la iluminación se puede llegar a confundir con blanco. La selección de los rangos de colores a sustituir es compleja, ya que cada documento tiene diferentes exposiciones y el blanco nunca es blanco, es blanco azulado, blanco cálido, etc. Por lo tanto un rango que puede producir una muy buena segmentación para una imagen, en otra puede llegar a eliminar la mitad del texto, o cierta parte del documento.

Por lo tanto, se ha optado por una extracción más cautelosa, eliminando rangos en los que conocemos que se encuentran esos colores con certeza, con la desventaja de que difícilmente llegaremos a eliminar todas las marcas de bolígrafo, pero nunca llegaría a afectar al texto.

Teniendo esto en cuenta y dada la imagen interpolada del paso anterior, se procede a extraer los canales azul y rojo de esta imagen, creando una unión entre ambas máscaras. A estas máscaras se les aplica una dilatación, (con diferentes *kernels*, ya que se detecta muy poco azul) y se le sustrae esta máscara a la imagen. Sobre la imagen interpolada se computa un algoritmo de *K-Medias* para determinar los colores dominantes, y nos quedaremos con el primero de ellos, es decir, el blanco del documento. Por lo tanto nos quedará una imagen con las marcas de bolígrafo sustituidas por este color. Como efecto colateral de esta segmentación, a veces también conseguimos eliminar el fondo, en el caso de que la extracción de bordes nos haya dejado algo de fondo en el nuevo documento. Después se realiza el *Threshold* adaptativo que deja esta sustitución completamente uniforme, que consigue eliminar las manchas y los pliegues del documento.

4. Segmentación del texto

Después del paso anterior, en el que se realiza un *Threshold* adaptativo, la segmentación del texto es trivial, ya que prácticamente conseguimos nuestro objetivo a la salida del paso anterior. Primeramente extraemos los negros de la imagen y a esta máscara le aplicamos un operador de erosión para eliminar el ruido que se podría haber conseguido con el *Threshold*. El error acumulado de fases anteriores puede perjudicarnos en la correcta segmentación del texto, ya que si no se consigue filtrar alguna marca o borde este método no las eliminará.

Una vez hemos conseguido extraer el texto, simplemente creamos una matriz con las dimensiones iniciales de la imagen y la llenamos de blanco. Después a esa matriz se le resta la máscara extraída de la imagen anterior y sobre esta se realiza un threshold binario, dejando los componentes negros a negro puro. Consiguiendo así nuestro objetivo.

5. Resultados

Los resultados obtenidos en la mejora y segmentación de documentos, con el conjunto de test dado, no son los mejores. Se obtiene una precisión sobre 60 % en la extracción correcta de contornos y un 100 % en la segmentación de texto, aunque la metodología empleada nos elimine todas las manchas, tiene muchos problemas a la hora de eliminar las marcas de bolígrafo.

Estos resultados se deben principalmente a la gran variabilidad de imágenes en el conjunto de test. Con respecto a la detección de contornos, obtenemos dicho resultado ya que algunas de las imágenes apenas hay contraste entre el fondo y el documento, o debido a la presencia de un fondo con ruido, patrones, o iluminación variante. Con respecto a las marcas, el bolígrafo azul es difícil de eliminar en ciertas imágenes ya que lo puede llegar a confundir con negro, o el rojo con blanco, dependiendo de la iluminación.

En conclusión, podemos afirmar que la metodología obtiene resultados aceptables. En los documentos reconstruídos observamos una buena segmentación del texto, a veces con un poco de ruido, pero sería un buen preprocesador de imágenes para algún algoritmo de *Machine Learning* para el reconocimiento de lenguaje, por ejemplo. Obviamente en el trabajo propuesto hay diferentes problemáticas con las se puede lidiar, presencia de ruido, mala iluminación, etc., pero hasta cierto punto.