

Procesamiento de Imágenes

Trabajo práctico 2

Manuel Spreutels

Augusto Rabbia

Noviembre, 2023

Introducción

El trabajo práctico consiste en dos consignas principales. La primera se trata de detectar y diferenciar dados y monedas, así como sus valores, en una imagen a color con fondo no uniforme. La segunda, detectar y segmentar patentes, así como segmentar los caracteres de las mismas, de autos en contextos urbanos.

Setting

El trabajo se desarrolló en Python 3.11, aprovechando las funciones proveídas por la librería de OpenCV, Matplotlib, y numpy.

Desarrollo

Problema de las monedas y los dados

La solución encontrada para este problema se dividía en 2 partes:

- Encontrar las monedas.
- Encontrar los dados, a partir de esta información.

Para encontrar las monedas, simplemente utilizamos HoughCircles, y buscamos todos los círculos en la imagen que tuvieran un radio suficiente, y no demasiado grande. Sin embargo, esto arrojaba más círculos de los deseados:



Para lidiar con esto, sencillamente nos quedamos con sólo uno de los círculos que se intersequen.

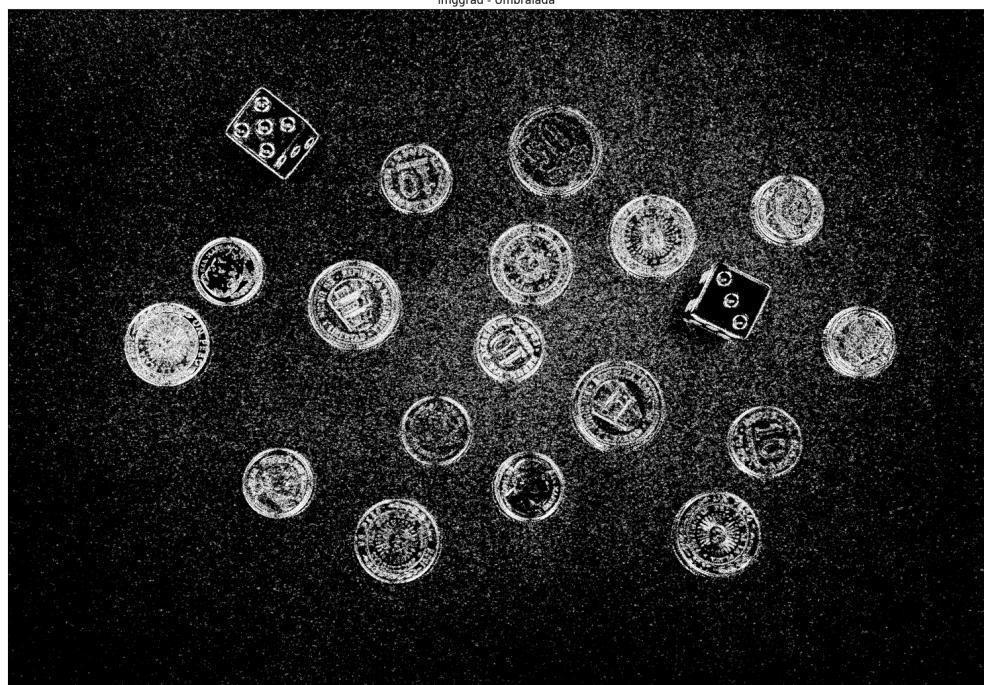
Posición de las monedas.



Una vez obtuvimos los círculos, pasamos a obtener los dados.

Para lograrlo, buscamos los bordes de la imagen usando Sobel, y luego umbralamos la imagen para obtener la siguiente imagen binaria:

imggrad - Umbralada

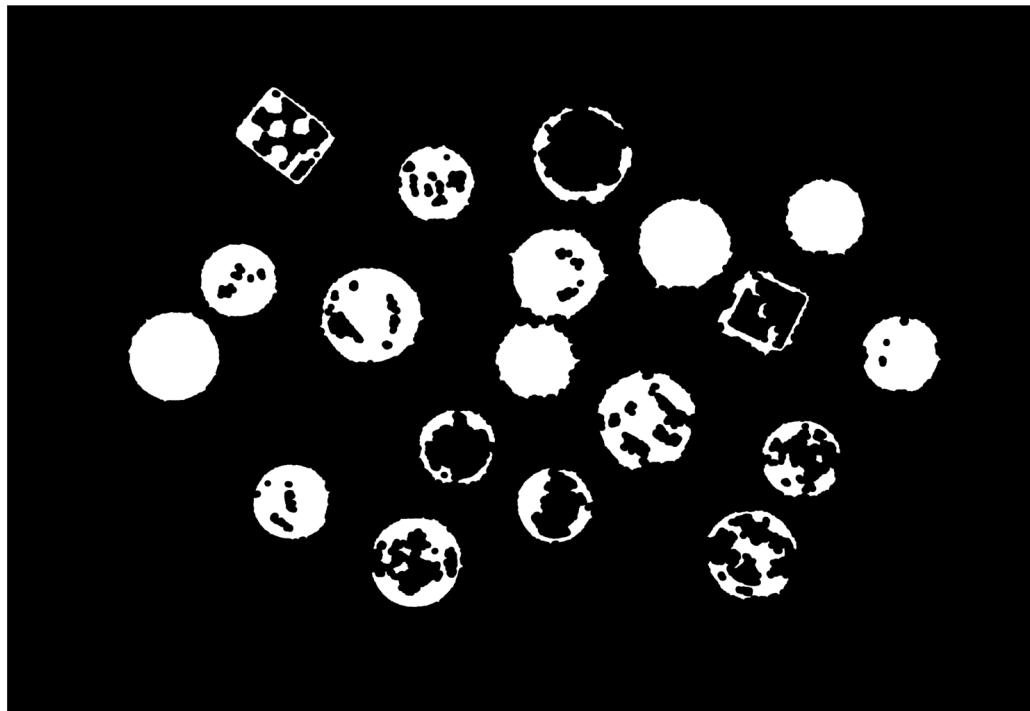


Ahora bien, esta imagen tiene una gran cantidad de ruido, lo cual hace imposible encontrar figuras en ella. Por lo tanto, eliminamos el ruido buscando contornos y coloreandolos de negro, y luego, sobre esto, hacemos clausura, lo cual finalmente resulta en lo siguiente:



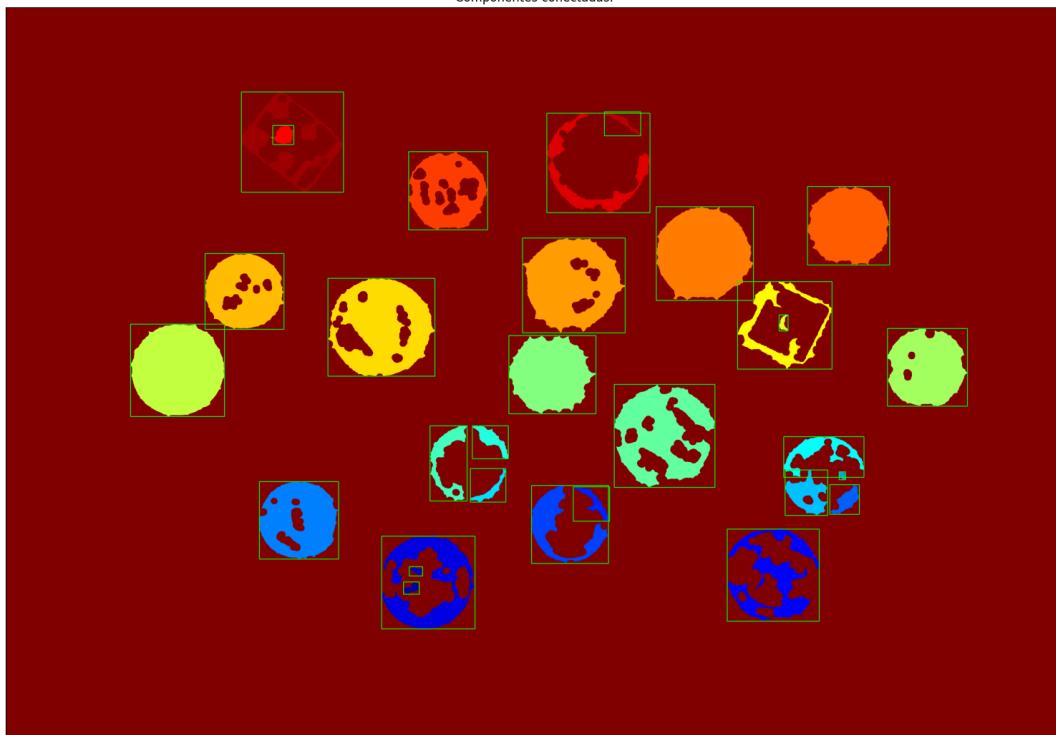
Ahora bien, queremos utilizar componentes conectadas para poder encontrar la posición de los dados. Esta imagen tiene un gran problema: El dado superior derecho está cortado, y resultarían 2 componentes conectadas de este. Por lo tanto, aplicamos clausura, obteniendo el siguiente resultado:

imClausura



Ahora sí, podemos obtener las componentes conectadas de la imagen:

Componentes conectadas.



Desde esta imagen, filtramos las componentes conectadas que no estén sobre uno de los círculos obtenidos previamente, que representaban las monedas, y que tengan un tamaño suficiente.

Nótese que no nos afecta que las monedas estén cortadas luego de la clausura por la manera en que filtramos.

De esta forma, obtenemos las posiciones de los dados, y con ellas, su subimagen, sobre la cual simplemente aplicamos HoughCircles nuevamente para obtener la cantidad de puntos que tenga el dado:

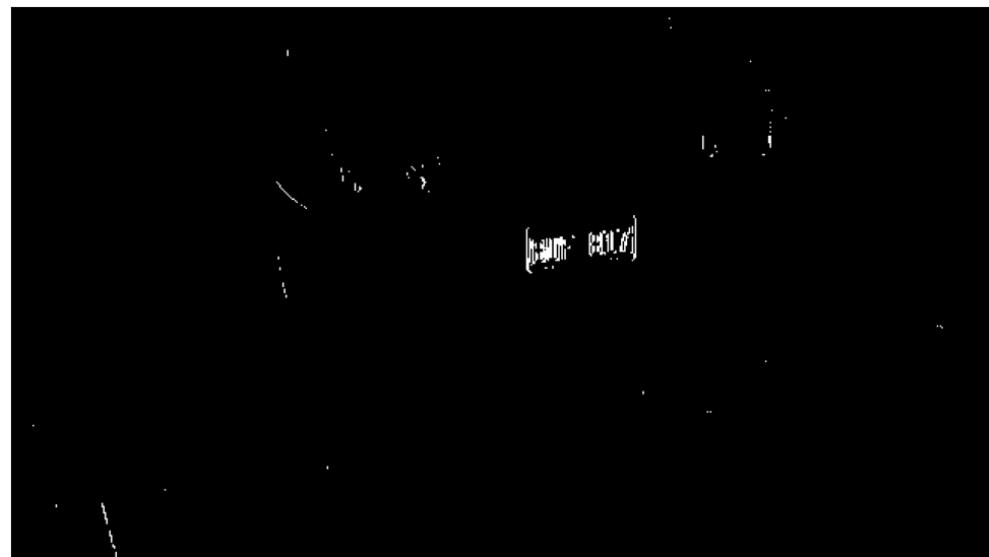


Finalmente, imprimimos los resultados obtenidos por pantalla.

Problema de las patentes

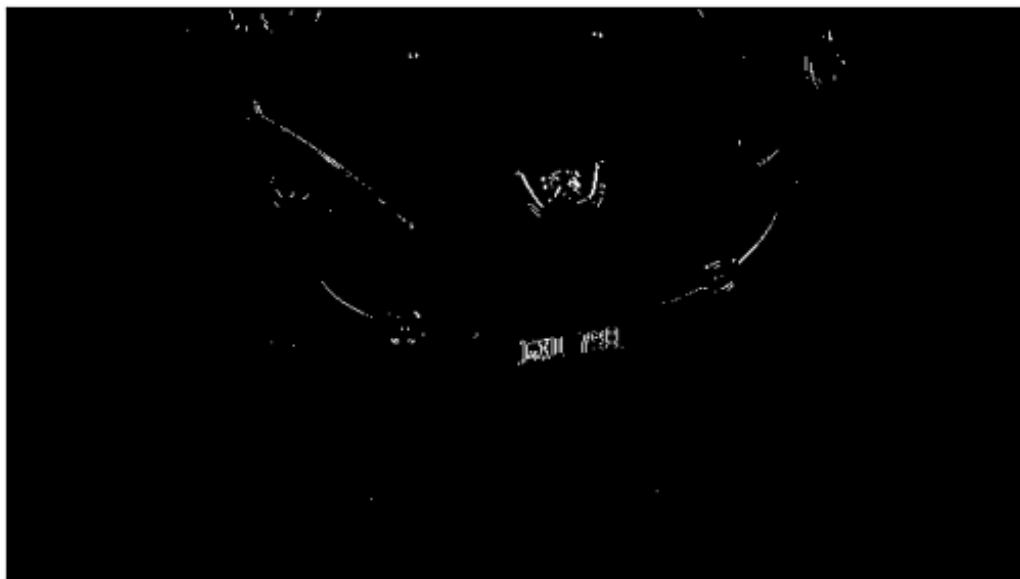
Para las patentes, comenzamos por pasar la imagen a escala de grises (directamente la leemos en cv2.IMREAD_GRAYSCALE) para luego hacer Sobel en la dirección horizontal, y umbralizar:

Ejemplo 1:



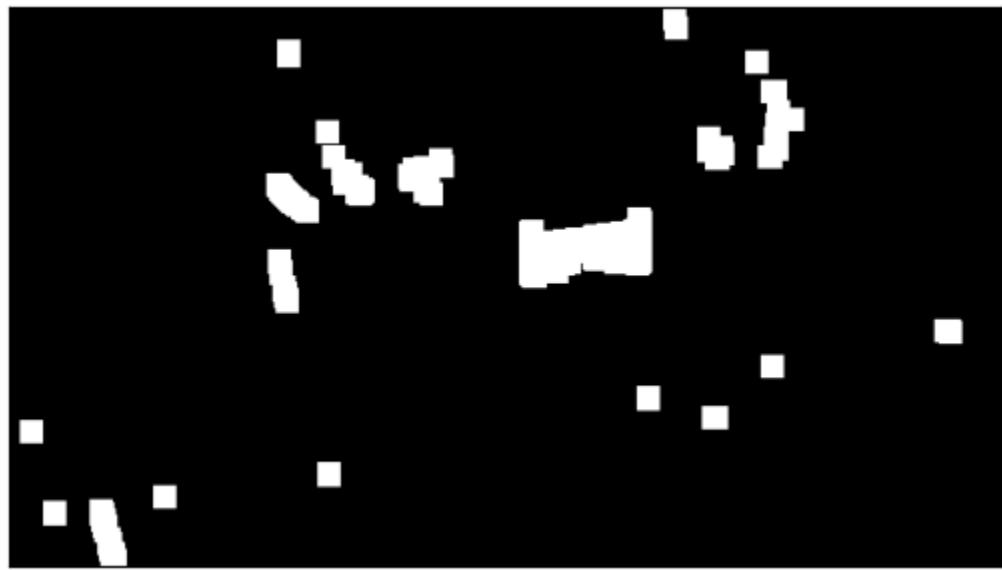
Ejemplo 2:





Esto nos da una idea de buenos candidatos a posibles patentes. Hacemos una clausura con un kernel significativamente amplio (13 píxeles), lo cual nos unifica los puntos blancos obtenidos en el umbralado en clústeres que representan componentes conectadas

Ejemplo 1:



Ejemplo 2:



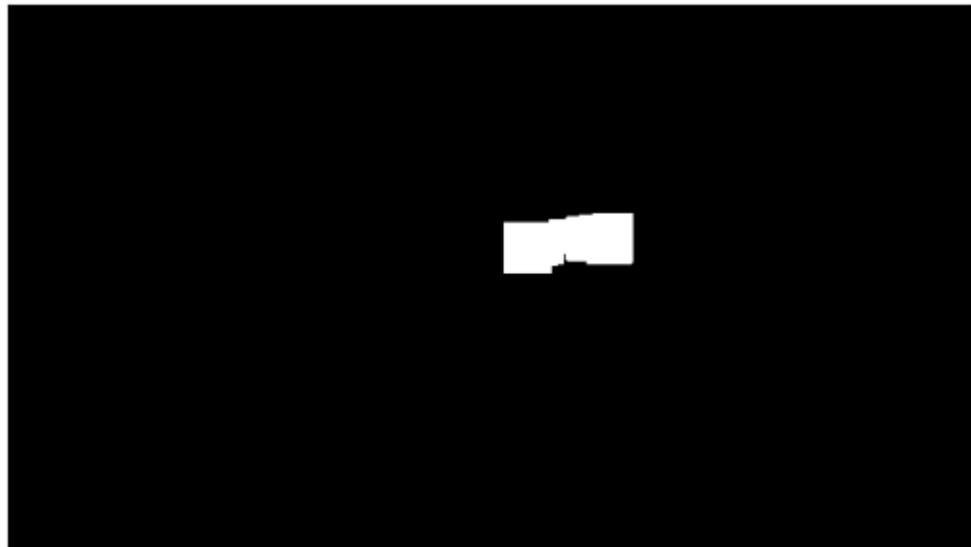
Como se observa, en todos los casos, la porción perteneciente a la patente de cada una de las imágenes se transforma en una sola componente conectada



Sin embargo, en casi todos los casos, no será la única. Por ello, tendremos que realizar otro procesamiento para distinguir la patente dentro de este conjunto de componentes conectadas. Para lograr esto, nos basamos en algunas de las características que se pueden obtener de las propias componentes. Por ejemplo, la relación entre el alto y el ancho, o cantidad de puntos en blanco/área de la componente), que identifican y distinguen de manera clara en la mayoría de los

casos a la patente del resto de componentes. De esta forma, filtramos las posibles patentes, eliminando puntos pequeños y ruido con apertura:

Ejemplo 1:



Ejemplo 2:



Con este resultado, ya sí podemos tener un indicio de cuál componente será la patente. Una vez obtenida la componente de la patente, utilizamos su bounding box para crear la subimagen de la región de interés en la imagen original.

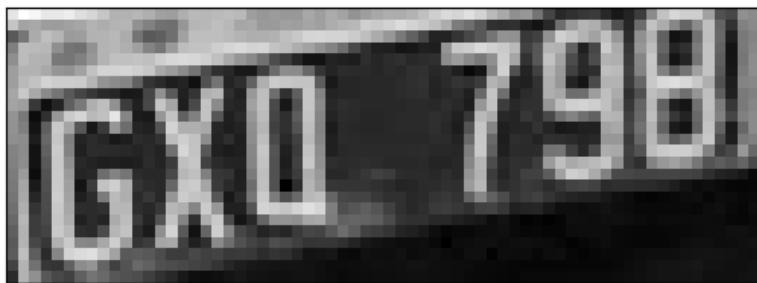
Ejemplo 1:

Patente



Ejemplo 2:

Patente



Luego, pasamos la subimagen original de la patente a binaria umbralando, lo que resulta en una imagen de estas características:

Ejemplo 1:



Ejemplo 2:



Planeamos utilizar componentes conectadas para segmentar los caracteres, pero, como se puede observar, con cualquier umbral que permita distinguir las letras de manera clara, algún carácter queda unido a los bordes exteriores de la patente en alguna de las patentes, lo cual distorsiona las componentes conectadas que buscamos.

A esta subimagen le aplicamos nuevamente Sobel, esta vez en el eje Y:

Ejemplo 2:



Ejemplo 2:

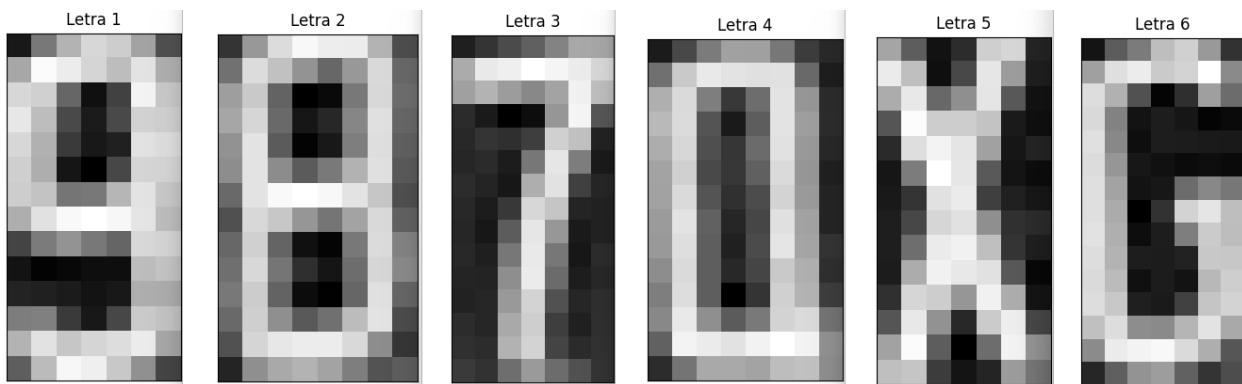


Esto nos permite distinguir claramente los cuadrantes de las patentes que rodean a los caracteres. Lo cual nos ayuda a solucionar el problema que teníamos los caracteres de la siguiente manera: obtenemos líneas suficientemente largas (con cv2.HoughLines) que demarcan

esos mismos bordes problemáticos. Luego, procedemos a pintarlas de negro, separando así a los caracteres de los bordes.



Finalmente, estamos en condiciones de obtener las componentes conectadas, y por tanto, (filtrándolas por alto y largo) los caracteres de la patente en la imagen original.



Es importante recalcar que este procesamiento NO funciona para todas las imágenes.

Resultados

Problema 1

Al finalizar la ejecución del programa, dada la siguiente imagen seleccionada como entrada:



Por pantalla se muestran los siguientes resultados:

Cantidad de monedas: 17

Monedas de 10 centavos: 9

Monedas de 50 centavos: 3

Monedas de 1 peso: 5

El valor de cada dados es:

Dado 1: 5

Dado 2: 3

Problema 2

Se muestra, en caso de haber sido encontradas, para cada ejemplo, la subimagen de la patente, así como las bounding-boxes de los caracteres.

- Imágen 1



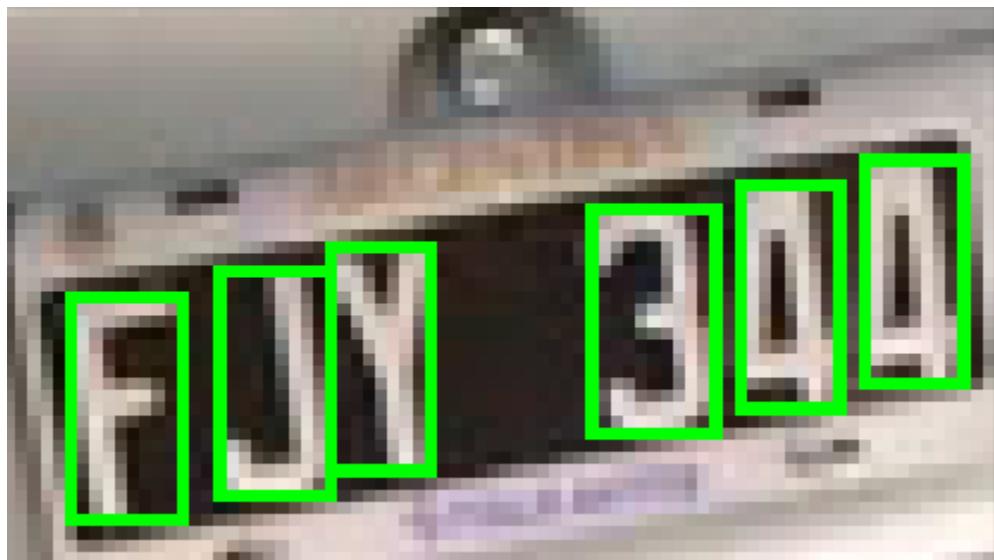
- Imagen 2



- Imagen 3



- Imagen 4
- Patente no encontrada
- Imagen 5



- Imagen 6



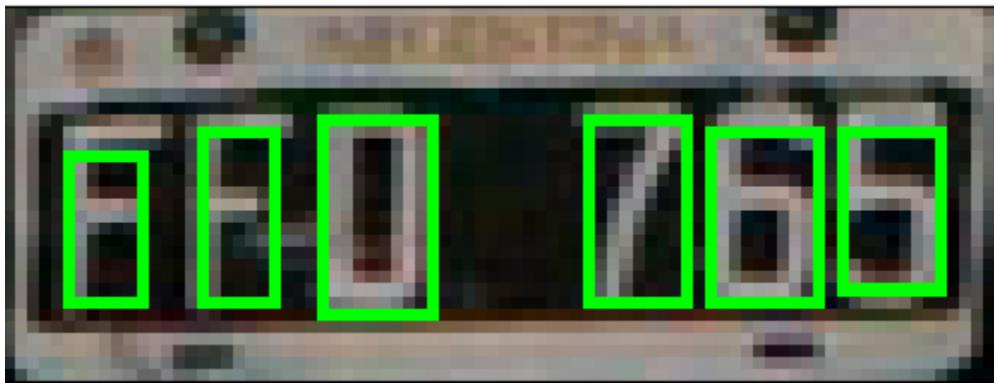
- Imagen 7



- Imagen 8



- Imagen 9



- Imagen 10



-
- Imagen 11



- Imagen 12

Patente no encontrada

Conclusion

En conclusión, el desarrollo de este trabajo nos da un pantallazo sobre técnicas para la detección, diferenciación y segmentación de objetos específicos imágenes con una complejidad mayor a las vistas hasta ahora en clase, que son más ejemplos de laboratorios a fines prácticos. Para encontrar datos y monedas con valores distintos en imágenes con fondos no uniformes, el mayor desafío fue aquel de encontrar una manera de lidiar con el ruido sin perder la información necesaria para encontrar la posición de los datos.

Por otro lado, para detectar y segmentar patentes y caracteres de automóviles en entornos urbanos, hubieron varios desafíos, pero principalmente, la necesidad de extrapolar un mismo fragmento de código a varios problemas, con diferentes tipos de autos, patentes, niveles de luminosidad, etc. incrementó la dificultad sustancialmente.