

INFORME TP2

PROCESAMIENTO DE IMÁGENES, 2025



FACULTAD DE CIENCIAS EXACTAS, INGENIERÍA Y AGRIMENSURA

Tecnicatura Universitaria en Inteligencia Artificial

Bollini Lorenzo - B-6729/6

Speranza Emanuel - S-5882/3

Introducción.

En este proyecto nos tocó resolver dos problemas distintos:

El primero, desarrollar un algoritmo capaz de procesar una imagen con dados y monedas, identificar si se trata de uno u otro, y además, clasificar en el caso de los dados, de que cara se trataba (d1,d2,etc.), y en el de las monedas, el valor de la mismas (10c, 50c, 1peso).

El segundo, un algoritmo que procese un total de 12 imágenes de automóviles estacionados, para luego segmentar las patentes y los caracteres dentro de ellas.

Problema 1 – Detección y clasificación de monedas y dados

¿El primer obstáculo de este planteo? El fondo irregular. Una vez cargada la imagen, la escalamos y pasamos a escala de grises. Aplicamos un filtrado Gaussiano para el ruido. La iluminación irregular de la imagen hizo que en algunos sectores el fondo este mas oscuro que en otros. Para destacar tanto los dados como las monedas, decidimos utilizar Top-Hat, basado en cierre y apertura para destacar objetos brillantes.

Una vez “corregida” (tal vez, “tratada”, sea una mejor palabra para este caso) la irregularidad de la iluminación, encontramos que algunas de las monedas mas oscuras/desgastadas y pertenecientes al sector mas oscuro del fondo apenas se notaban, o se mezclaban con el fondo. Esto, claramente iba a representar un desafío a la hora de detectar los bordes de los objetos que queremos clasificar.

El proceso de detección de bordes comenzó con Canny. ¿Se podían identificar todos los objetos? Si, pero el tamaño de las monedas que mencionamos antes claramente se veía afectado. Para solucionar esto usamos dilatación, y con la dilatación, aparecieron objetos conectados por ruido. Estos objetos adyacentes los desconectamos aplicando un cierre y por último una erosión.

Detectando cada objeto ahora sí, el algoritmo debía diferenciar entre dados y monedas. Para esto utilizamos el factor de forma provisto ($\text{área} / \text{perímetro}^2$), apoyado de un análisis de la cantidad de vértices.

A simple vista notamos que las monedas, que tienen distintos valores, 10 centavos, 50 centavos y 1 peso, también tienen distintos tamaños. Con esto en

cuenta, la clasificación de cada una depende del tamaño. Estimamos su denominación de acuerdo al área que ocupaban.

Para los contornos clasificados como dados, recortamos el interior del objeto y aplica umbralización adaptativa para detectar los puntos. Cada uno de los dos dados tenía su dificultad. El d5 en la esquina superior izquierda de la imagen, tenía 3 puntos pertenecientes a una cara al costado, por lo tanto lo detectaba como un d8. El segundo dado (d3), se encontraba en una zona donde todavía existía un poco de ruido, debido a la iluminación irregular. Los 3 puntos de la nada eran 5. Para lidiar con esto, nos servimos de la circularidad de un punto, es decir, definimos un umbral (que tan cercano está el punto a ser un círculo perfecto) el cual el punto debía superar para ser considerado un punto. Con esto, ya terminamos la clasificación de las monedas y de los dados

Al final se dibujan las etiquetas sobre la imagen original y se muestra un resumen, indicando la cantidad de monedas clasificadas por denominación y los valores encontrados en los dados.



Imagen binarizada después de aplicar el filtrado Gaussiano y Top Hat.

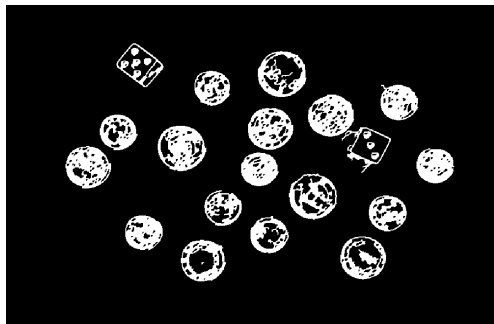
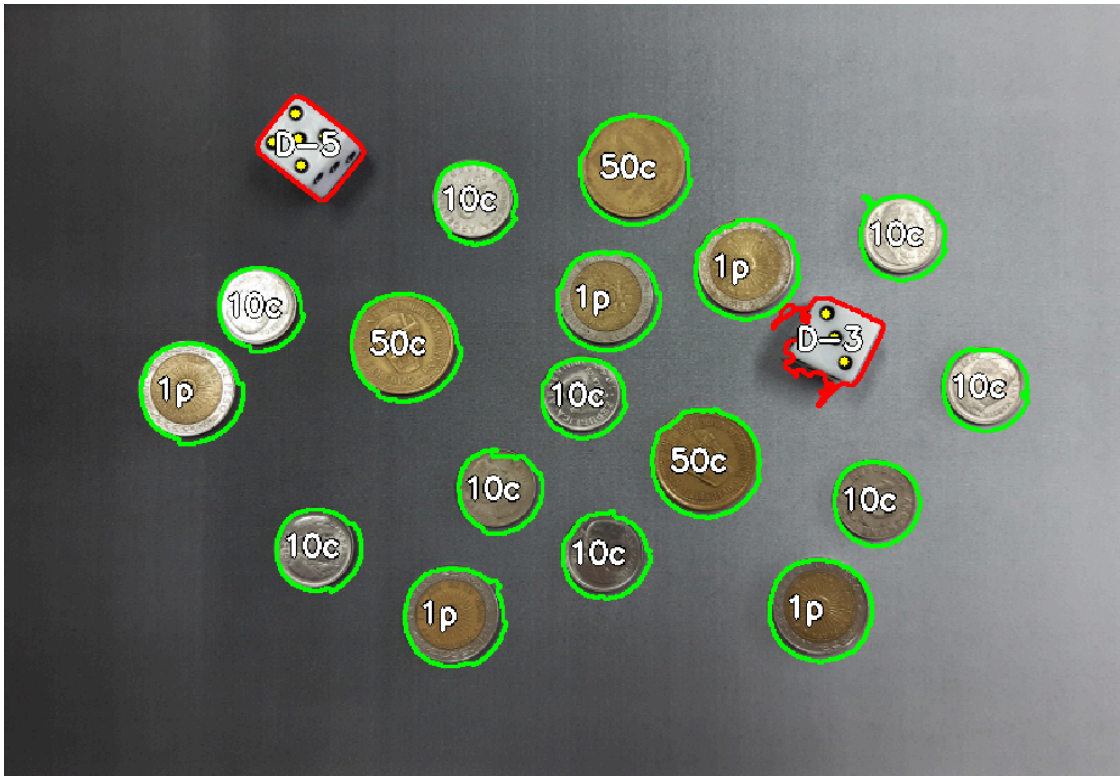


Imagen del resultado del Canny (Erosión ya aplicada)



Resultado Final.

Problema 2 – Detección de patentes y segmentación de caracteres.

El segundo problema aborda la detección automática de patentes vehiculares y la segmentación de sus caracteres. El algoritmo desarrollado divide el proceso en cuatro etapas secuenciales: preprocesamiento, definición morfológica, clasificación geométrica y una segmentación robusta con reintentos automáticos.

Este problema presentó distintas dificultades, para la detección de las patentes y también para la segmentación de los caracteres. Las fotos, estaban sacadas de distintos ángulos.

En la etapa de **Preprocesado**, la imagen original se convierte a escala de grises y se suaviza mediante un filtrado Gaussiano para reducir el ruido. A continuación, se aplica un operador Sobel en X con el fin de resaltar los gradientes blancos verticales, ideal para detectar entre tantos objetos posibles uno que contenga manchas blancas verticales (como el caso de las patentes que tienen un fondo negro con caracteres blancos). Luego, normalizamos para obtener un buen contraste. Para esto, utilizamos otsu. En esta primer etapa,

obtenemos una imagen binarizada donde se destacan las texturas verticales blancas .

Ahora tenemos que encontrar potenciales patentes. Esta etapa la denominamos la **Definición del objeto**. Utilizamos morfología matemática con el fin de consolidar la región de la placa. Primero, se aplica una erosión vertical con un kernel rectangular estrecho (1x3) para eliminar ruido o líneas finas horizontales (cosa que nos sirvió para ignorar por ejemplo, la parrilla del frente del vehículo de una de las imágenes). Por último se realiza una operación de cierre con un kernel horizontal alargado (14x1), lo que permite fusionar los caracteres individuales en un único bloque sólido rectangular, facilitando su detección como un solo contorno.

En la **Clasificación de la patente**, la dificultad que vamos a encontrar está en definir que rectángulos son y cuales no son patentes. Para esto, encontramos que la mejor solución era evaluar la relación de aspecto para encontrar rectángulos mas anchos que altos. No solo eso, si no que también, conociendo que las patentes estan en la parte inferior del vehículo, descartamos los rectángulos que no se encuentren en dicho sector.

Una vez ya obtenidos estos candidatos ordenados por puntaje, la función detectar_placa, valida del contenido el rectángulo, llamando a la segmentación de caracteres.

Finalmente, la **Segmentación de caracteres** implementa una estrategia de fuerza bruta controlada, debido a que es extremadamente difícil y poco probable encontrar una solución al tener fotos de patentes que, si bien todas tienen el mismo formato, varia el ángulo de la imagen, el desgaste de la patente o inclusive el color de los bordes. Entonces, el algoritmo intenta segmentar el recorte de la patente mediante una función núcleo que realiza los siguientes pasos:

El primer intento consiste en umbralizar con otsu y luego erosionar para separar los caracteres del clásico borde (inferior y superior) blanco. Luego se refina el recorte utilizando componentes conectados para aislar la zona del texto. Para los casos en los que la patente esta doblada, o tiene una sombra irregular, se utiliza un umbral fijo para la binarización. Había casos donde los caracteres de algunas patentes, seguian pegados a estos bordes, en esa situación se aplico una erosión vertical extra para “despegarlos”. Si la

configuración inicial no detecta al menos 6 caracteres, el sistema reintenta automáticamente el proceso variando los parámetros (cambiando Otsu por umbral fijo o aumentando la intensidad de la erosión) hasta obtener el mejor resultado posible. Por último, si se detectaba mas de 6 caracteres, se filtraban los mismos basándonos en el ancho y alto que esperamos de los caracteres.

Finalmente, el programa muestra el recorte de la patente detectada, la segmentación obtenida y la cantidad total de caracteres hallados, cumpliendo así con los objetivos planteados para el problema.



Imagen binarizada después de aplicar el filtro Gaussiano, detección de bordes verticales con Sobel y umbralización Otsu.



Imagen del resultado de la erosión vertical seguida de una clausura horizontal.



Región de interés extraída de la imagen original correspondiente a la patente detectada.



Resultado Final.

Conclusión:

El primer problema nos invitó a poner en práctica los conceptos de descriptores y clasificación de formas. Analizamos qué características definen a un dado (forma cuadrada, puntos en su superficie) y qué características definen a una moneda (circularidad), para luego clasificarlos según estos valores. Si bien logramos resolver el planteo para esta imagen específica, reconocemos un margen de mejora en términos de robustez: nuestra clasificación de monedas se basó en valores absolutos de área, lo cual fallaría con imágenes de diferentes dimensiones o con objetos adicionales. Una mejora sería utilizar proporciones relativas entre objetos, por ejemplo, determinar qué porcentaje de la imagen representa cada moneda en relación a las demás, permitiendo que el algoritmo funcione independientemente de la escala.

En el segundo problema, el objetivo fue detectar patentes y segmentar sus caracteres. Nuestro enfoque utilizó la detección de caracteres como método de validación de las placas candidatas, lo cual resultó efectivo pero rebuscado. Una alternativa “superior” habría sido filtrar candidatos primero por geometría (relación de aspecto típica de patentes), para luego validar solo los mejores candidatos basandonos por ejemplo ahora si, en la posición con respecto a la imagen, o también, si contiene dos grupos de 3 manchas blancas cada uno, separados por un espacio en el medio.