

Tarea 1 "Qué hora es?" - IEE2714

Alonso Rivera
adrivera1@uc.cl

Motivacion

La detección de imágenes es una tarea cotidiana en el mundo de hoy en día. Es por esto que resulta útil desarrollar y comprender cómo es posible extraer información de una imagen de la manera más básica, analizando únicamente sus píxeles de forma matemática.

Solucion propuesta

La solución propuesta se compone de dos pasos: la segmentación del reloj y la detección de la hora. Para ilustrar esto, observaremos la evolución del reloj en el apéndice (Figura 1).

Segmentacion

Como primer paso, reducimos las dimensiones de la imagen realizando un muestreo cada 12 píxeles (Figura 2). Luego, nos quedamos únicamente con el canal R. Para lograr un mayor contraste, filtramos para retener únicamente los píxeles con valores superiores a 165, y escalamos los valores de estos píxeles entre 0 y 255 (Figura 3), aplicamos una máscara de 5 x 5 píxeles. El objetivo de esto es crear gradientes y suavizar la imagen consigo misma, a fin de eliminar los píxeles aislados alrededor de otros con un contraste opuesto (Figura 4), aplicamos un umbral que asigna valores binarios, estableciendo el límite en 235 (Figura 5). Eliminamos las filas y columnas que contienen 15 o menos píxeles blancos, ya que esta información no aporta al análisis (Figura 6). Con esto podemos obtener los límites laterales del reloj para segmentarlo (Figura 7).

Deteccion de la hora

Primero reducimos la imagen original mediante un muestreo cada 2 píxeles. Luego, segmentamos la imagen utilizando los datos obtenidos en la primera parte y volvemos a aplicar un muestreo para obtener una imagen cuadrada de 160 por 160 píxeles. Dado que el minuterero y el horario son de un color distinto al segundero, dividimos el proceso en dos partes.

Deteccion de hora y minuto

Para esto, nos quedamos con el canal R y filtramos la imagen utilizando un umbral superior de 120 y otro inferior de 20 (Figura 8). Con la imagen filtrada, volvemos a centra la imagen para que el centro se encuentre en el centro de la imagen, realizamos un recorrido con 3 radios distintos desde el centro para detectar la presencia de píxeles blancos. Si encontramos píxeles blancos en algún radio, guardamos el valor del ángulo con respecto al centro en una lista. Posteriormente, agrupamos los valores cercanos y comparamos las listas entre sí. Si encontramos un ángulo similar en las tres listas, significa que ese es

el ángulo correspondiente al minuterero. Sin embargo, si solo encontramos coincidencias en las 2 primeras listas, entonces se trata del ángulo correspondiente al horario.

Deteccion de segundos

De manera similar, procedemos a realizar un umbral doble en los canales R y G de la imagen y realizamos una operación AND entre estos (Figura 9). Con la imagen filtrada, aplicamos el mismo proceso que para la hora y minuto pero con 4 radios. Si encontramos el mismo ángulo en las 4 listas generadas, determinamos que ese ángulo corresponde al segundero.

Experimentos realizados

Durante el proceso, se priorizó la eficiencia del algoritmo con el objetivo de minimizar el tiempo de ejecución. En la primera iteración, el tiempo total era de 40 segundos con una eficacia del 87.5%. En términos generales, se seguía una metodología similar, aplicando múltiples pasos de difuminación e iterando constantemente en diversos pasos del proceso.

Debido a la extensión del tiempo de ejecución en la iteración inicial, se optó por reemplazar las funciones de lenguaje explícitas de Python por operaciones matriciales implícitas de NumPy. Esta adaptación permitió realizar las operaciones de manera significativamente más eficiente. Como resultado, el tiempo promedio de ejecución se redujo a menos de 5 segundos para las 6 imágenes, lo que a su vez condujo a mejoras en la precisión del proceso. Gracias a esta modificación, se logró alcanzar una eficacia del 97%.

Conclusiones

A modo de resumen, reducir el tamaño de una imagen antes de ser procesada es útil para agilizar la velocidad de los algoritmos, en casos como este, un submuestreo es adecuado. Difuminar una imagen resulta ser bastante efectivo para reducir el contraste de pequeños detalles, y utilizar umbrales es una buena técnica de segmentación de objetos en imágenes con alto contraste.

También, es una buena técnica reducir la imagen a un cuadrado para estandarizar el análisis del reloj. En el análisis de este, resulta práctico analizarla con píxeles binarios debido al alto contraste que presenta.

Anexos



Figure 1: Imagen inicial



Figure 2: Imagen submuestreada

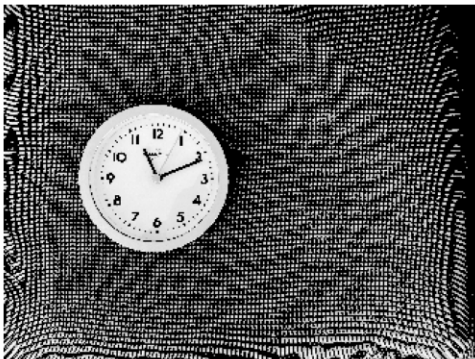


Figure 3: Rango escalado



Figure 4: Imagen difuminada

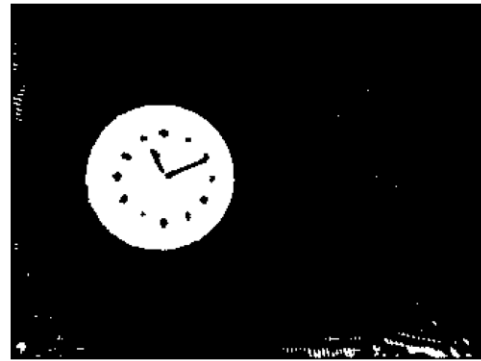


Figure 5: Imagen post umbral

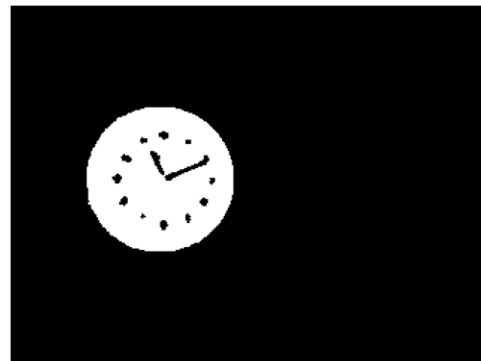


Figure 6: Imagen filtrada



Figure 7: Reloj segmentado



Figure 8: Reloj solo con minuterio y horario

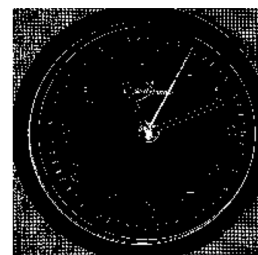


Figure 9: Reloj solo con segundero