Actividad autocorregible: Uso de filtros y contornos para la localización de placas en imágenes de coches

Objetivos

A lo largo de esta actividad conocerás y aplicaras los conceptos básicos para el procesamiento de imágenes.

* Aplicar operaciones de suavizado.
* Utilizar operadores espaciales.
* Utilizar operadores morfológicos.

Descripción de la actividad

El problema de reconocimiento de placas (también conocido como reconocimiento de matrículas) consiste en identificar y leer las placas de matrícula de los vehículos en imágenes o vídeos. Este problema es importante en una variedad de aplicaciones, como la seguridad de la carretera, la vigilancia y el seguimiento de vehículos.

El proceso de reconocimiento de placas generalmente incluye varios pasos, como la detección de la placa en la imagen, la segmentación de los caracteres individuales en la placa y la identificación y extracción del texto en los caracteres. A menudo, se utilizan técnicas de procesamiento de imágenes, como la detección de bordes y la transformada de Hough, para detectar la placa. Luego, se pueden utilizar técnicas de aprendizaje automático, como el reconocimiento de caracteres con OCR, para identificar el texto en la placa.

En esta actividad utilizaremos funciones básicas para llevar a cabo este proceso, el cual nos arroja como resultado la ubicación de regiones candidatas a ser placas. El código aquí presentado puede ser optimizado con otras técnicas; solamente se muestra una aproximación para conocer las diferentes funciones de procesamiento de imágenes que nos ayudarán a realizar esta tarea.

Preguntas

1. En el siguiente código vamos a leer la imagen de un coche tanto a color como en niveles de gris, responde correctamente cuáles son los valores correctos que arroja alto 1 y ancho 1 y cuántos canales tiene:

Una captura de pantalla de un celular con letras

Descripción generada automáticamente con confianza media

A. 248x400, 1 canal.

B. 248 x 400, 3 canales.

C. 320 x 460, 3 canales.

D. 248 x 420, 1 canal.

1. El kernel aplicado a la imagen cars3.png es un filtro espacial\_\_\_\_, y el resultado es la imagen:

Texto

Descripción generada automáticamente

A. Pasa bajo.

Imagen en blanco y negro de un carro

Descripción generada automáticamente con confianza media

B. Pasa alto.



C. Pasa baja.

Imagen en blanco y negro de un carro

Descripción generada automáticamente

D. Pasa alto.



1. El kernel aplicado a la imagen es un filtro \_\_\_\_\_\_\_\_ y se aplica principalmente para \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_. La imagen resultante es:

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

A. Es un filtro pasa baja y se aplica para quitar ruido y suavizar la imagen. La imagen resultante es:

****

B. Es un filtro pasa alta y se aplica para suavizar la imagen. La imagen resultante es:



C. Es un filtro pasa baja y se aplica para resaltar los contornos. La imagen resultante es:

Diagrama

Descripción generada automáticamente

D. Es un filtro pasa alta y se aplica para resaltar los contornos. La imagen resultante es:

Imagen que contiene Texto

Descripción generada automáticamente

1. El objetivo de la ecualización de imágenes es \_\_\_\_\_\_\_\_. El resultado al aplicar ecualización a la imagen coche11.jpg es:

Texto

Descripción generada automáticamente

A. El objetivo de la ecualización es cambiar una imagen a niveles de gris enfatizando sus contornos. La imagen resultante es:



B. El objetivo de la ecualización es mejorar el contraste en una imagen, la imagen resultante es:

**Una camioneta negra estacionada en la calle

Descripción generada automáticamente**

C. El objetivo de la ecualización es cambiar una imagen a niveles de gris, haciendo énfasis en los tonos obscuros.

Imagen en blanco y negro de un carro

Descripción generada automáticamente

D. El objetivo de la ecualización es mejorar el contraste, la imagen resultante es:

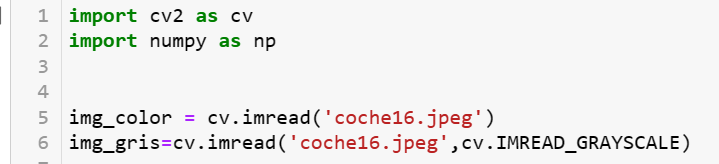
Un coche antiguo estacionado en un área abierta

Descripción generada automáticamente

**Localización de placas con técnicas tradicionales de visión por computadora**

1. Para llevar a cabo el reconocimiento de placas es necesario antes localizarlas, existen una gran cantidad de condiciones que afectan una imagen como puede ser la iluminación, distancia y rotaciones, entre otros. Revisa las diferentes imágenes de coches que acompañan la actividad para que identifiques estos cambios. Para lograr el objetivo, hemos definido una metodología para encontrar placas candidatas en una imagen mediante técnicas tradicionales de procesamiento de imágenes, la metodología sugerida es la siguiente:

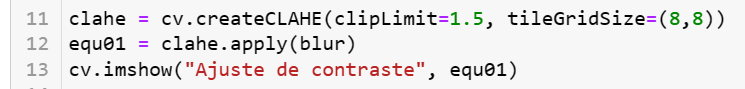
**Paso 1.** Leer imagen a color y transformar la imagen de color a gris.



**Paso 2.** Suavizar la imagen (filtro gaussiano con kernel de 11x11).



**Paso 3.** Mejora de contraste (ajuste local de contraste adaptativo).



**Paso 4.** Binarizar la imagen (Otsu).



**Paso 5**. Obtener bordes (CANNY).

Imagen que contiene Texto

Descripción generada automáticamente

**Paso 6.** Guardar bordes.



**Paso 7.** Dibujar placas candidatas.

Imagen que contiene Texto

Descripción generada automáticamente

**Paso 8.** Cerrar ventanas y terminar.

Texto

Descripción generada automáticamente

Al ejecutar el código para la imagen coche16.jpeg, ¿cuál es la salida para los pasos 2, 3 , 4 y 5? Relaciona las columnas.

1. Imagen 1.

Dibujo en blanco y negro

Descripción generada automáticamente con confianza media

2. Imagen 2.

Imagen que contiene Diagrama

Descripción generada automáticamente

3. Imagen 3.

Imagen en blanco y negro de un carro

Descripción generada automáticamente

4. Imagen 4.

Imagen en blanco y negro de un carro

Descripción generada automáticamente

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Imagen 1. | 1 |  | A | Paso 2. |
| Imagen 2. | 2 |  | B | Paso 3. |
| Imagen 3. | 3 |  | C | Paso 4. |
| Imagen 4. | 4 |  | D | Paso 5. |

1. Utilizando el siguiente algoritmo:

**Paso 1.** Leer imagen a color y transformar la imagen de color a gris.

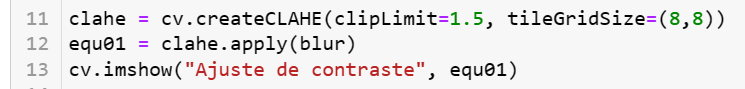
Texto

Descripción generada automáticamente con confianza media

**Paso 2.** Suavizar la imagen (filtro gaussiano con kernel de 11x11).



**Paso 3.** Mejora de contraste (ajuste local de contraste adaptativo).



**Paso 4.** Binarizar la imagen (Otsu).



**Paso 5.** Obtener bordes (CANNY).

Imagen que contiene Texto

Descripción generada automáticamente

**Paso 6.** Guardar bordes.



**Paso 7.** Dibujar placas candidatas.

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

**Paso 8.** Cerrar ventanas y terminar.

Texto

Descripción generada automáticamente

Si eliminamos el paso 2 de filtro gaussiano y lo aplicamos a la imagen coche14.jpeg, el resultado es:

A. No detecta la placa.

B. Detecta más de cinco candidatos a placa.

C. Detecta exclusivamente la placa.

D. Detecta tres candidatos de placa.

1. Utilizando el siguiente algoritmo:

**Paso 1.** Leer imagen a color y transformar la imagen de color a gris.

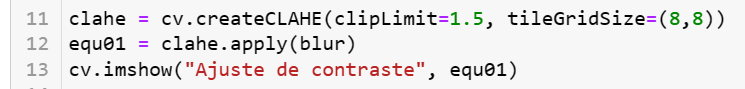
Texto

Descripción generada automáticamente con confianza media

**Paso 2.** Suavizar la imagen (filtro gaussiano con kernel de 11x11).



**Paso 3.** Mejora de contraste (ajuste local de contraste adaptativo).



**Paso 4.** Binarizar la imagen (Otsu).



**Paso 5.** Obtener bordes (CANNY).

Imagen que contiene Texto

Descripción generada automáticamente

**Paso 6.** Guardar bordes.



**Paso 7.** Dibujar placas candidatas.

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

**Paso 8.** Cerrar ventanas y terminar.

Texto

Descripción generada automáticamente

Si incluimos todos los pasos y lo aplicamos a la imagen coche14.jpeg, el resultado es:

A. Solo detecta la placa.

B. Detecta tres candidatos a placa.

C. Detecta dos candidatos a placa.

D. No detecta ninguna placa.

1. Utilizando el siguiente algoritmo, vamos a modificar unas líneas:

**Paso 1.** Leer imagen a color y transformar la imagen de color a gris.

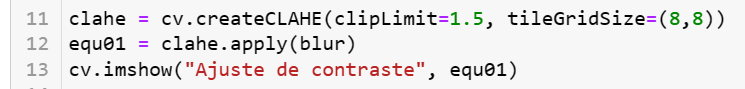
Texto

Descripción generada automáticamente con confianza media

**Paso 2.** Suavizar la imagen (filtro gaussiano con kernel de 11x11).



**Paso 3.** Mejora de contraste (ajuste local de contraste adaptativo).



**Paso 4.** Binarizar la imagen (Otsu).



**Paso 5.** Obtener bordes (CANNY).

Imagen que contiene Texto

Descripción generada automáticamente

**Paso 6.** Guardar bordes.



**Paso 7.** Dibujar placas candidatas.

Imagen que contiene Interfaz de usuario gráfica

Descripción generada automáticamente

**Paso 8.** Cerrar ventanas y terminar.

Texto

Descripción generada automáticamente

Vamos a aplicar un filtro gaussiano con un kernel de 3x3 en vez del de 11x11 (paso 2). Al aplicar el procedimiento de detección de placas, ¿cuántas placas detecta al aplicar el proceso a la imagen cars34.png? ¿Cómo se ve la imagen después de aplicar Canny?

A. No detecta ninguna placa.

B. Detecta exclusivamente la placa.

C. Detecta dos candidatos a placa.

D. Detecta más de tres placas.