



PROCESAMIENTO DIGITAL DE IMÁGENES

Nombre: Homero Meneses Vazquez, Enrique Perez Ramirez, Roberto Misael

Boleta:

Especialidad:

Fecha de la práctica: 4 abril 2023

Nombre de la práctica: Bordes y perfilado

Resultados de aprendizaje Propuestos (RAP's)

Analiza y aplica los algoritmos de detección de bordes: magnitud del gradiente, laplaciano de la gaussiana LoG.

Analiza y aplica uno de los algoritmos de perfilado: laplaciana + original.

Compara de forma experimental la diferencia entre detección de bordes a través de la magnitud del gradiente y LoG.

Objetivo

Un programa capaz de dar como resultado tres imágenes diferentes, a partir de una. El filtro de sobel, el de laplaciano segundo -los de bordes respectivamente- y el de perfilado de 3x3.

El objetivo de la práctica de bordes y perfilado en procesamiento de imágenes digitales es mejorar la calidad visual de una imagen mediante la identificación y realce de sus bordes y contornos, con el fin de destacar las características relevantes de la misma y facilitar su análisis y comprensión por parte del observador humano o de otros sistemas automatizados que requieran una imagen de alta calidad.

Introducción

La detección de bordes en el procesamiento de imágenes se refiere a la identificación y distinción de las regiones de una imagen donde hay una transición abrupta de los valores de intensidad de los píxeles. Estas transiciones pueden ser causadas por cambios en el color, la textura, la iluminación u otros factores.

El filtro de Sobel es un operador diferencial utilizado en procesamiento de imágenes para detectar los bordes de una imagen. Fue desarrollado por Irwin Sobel en 1968 y es uno de los operadores de detección de bordes más populares en el procesamiento de imágenes debido a su simplicidad y eficacia en la detección de bordes.

El filtro de Sobel se utiliza para calcular el gradiente de la imagen en una dirección específica (horizontal o vertical) utilizando una matriz de convolución que aproxima la derivada de la imagen en esa dirección.

El gradiente obtenido mediante el filtro Sobel resalta las regiones de la imagen donde hay una variación significativa en la intensidad de los píxeles, lo que indica un posible borde.

El filtro Laplaciano es especialmente bueno para detectar bordes en regiones de la imagen donde la intensidad de los píxeles cambia rápidamente.

El filtro Laplaciano es un ejemplo de filtro de convolución en el procesamiento de imágenes y se aplica a una imagen convolucionando los valores de intensidad de los píxeles con una matriz predefinida llamada "kernel". La aplicación del kernel

En este caso de forma específica para el Laplaciano, resalta las regiones de la imagen donde hay una gran variación de los valores de intensidad.

El filtro Laplaciano se utiliza a menudo en combinación con otros filtros, como el filtro Gaussiano, para reducir el ruido y mejorar aún más la calidad de la imagen y su detección de bordes.

El perfilado de imágenes se refiere al proceso de ajustar los parámetros de una imagen para mejorar su calidad y destacar ciertos aspectos específicos de la imagen. Esto se puede lograr mediante la modificación de los niveles de contraste, brillo, saturación, nitidez y otros atributos de la imagen. El objetivo del perfilado de imágenes es mejorar la visualización o análisis de la imagen resaltando características clave o eliminando detalles no deseados. El perfilado de imágenes se utiliza en una variedad de aplicaciones

Desarrollo

1. Selecciona una imagen a la que desees aplicar la detección de bordes.
2. Reutiliza el programa que produce la convolución de una máscara en una imagen.
3. Mediante un programa:
 - a. *Carga la imagen.*
 - b. *Construye las máscaras de sobelX, sobelY y LoG de 3x3.*
 - c. *Aplica el algoritmo de convolución discreta con las 3 máscaras de forma individual.*
 - d. *Calcula la magnitud del gradiente.*
 - e. *Muestra la imagen original y la imagen resultado después calcular la magnitud del gradiente y de LoG.*
 - f. *Construye la máscara de perfilado de 3x3 al 60%.*
 - g. *Aplica el algoritmo de convolución discreta con la máscara.*
 - h. *Muestra la imagen resultado.*

A continuación se mostrara el código y los comentarios que corresponden a que es lo que realiza el Código

```

import cv2
import numpy as np
from google.colab.patches import cv2_imshow
# Cargar la imagen
imagen = cv2.imread("/content/sample_data/hulk.jpeg", cv2.IMREAD_GRAYSCALE)
cv2_imshow(imagen)
# Máscaras de SobelX, SobelY y LoG
sobelx = np.array([[ -1,  0,  1], [ -2,  0,  2], [ -1,  0,  1]])
sobely = np.array([[ -1, -2, -1], [ 0,  0,  0], [ 1,  2,  1]])
laplacian = np.array([[ 0, -1,  0], [ -1,  4, -1], [ 0, -1,  0]])*1

|

# Algoritmo de convolución discreta
def convolucion(imagen, mascara):
    altura, ancho = imagen.shape
    m_alto, m_ancho = mascara.shape
    bordes = int((m_alto - 1) / 2)
    imagen_bordes = np.zeros((altura - bordes*2, ancho - bordes*2))
    for i in range(bordes, altura-bordes):
        for j in range(bordes, ancho-bordes):
            trozo = imagen[i-bordes:i+bordes+1, j-bordes:j+bordes+1]
            imagen_bordes[i-bordes, j-bordes] = np.sum(trozo * mascara)
    return imagen_bordes

# Calcular el gradiente
gradiente_x = convolucion(imagen, sobelx)
gradiente_y = convolucion(imagen, sobely)
magnitud = np.sqrt(gradiente_x**2 + gradiente_y**2)

#Calcular el laplaciano
laplaciano = convolucion(imagen, laplacian)
cv2_imshow( laplaciano)

# Mostrar las imágenes original y resultado

cv2_imshow( magnitud)
cv2.waitKey(0)
cv2.destroyAllWindows()

# Máscara de perfilado
perfilado = np.array([[ 0, -1,  0], [ -1,  5, -1], [ 0, -1,  0]])*0.6

# Aplicar el perfilado
perfilado_imagen = convolucion(imagen, perfilado)

# Mostrar la imagen resultado
cv2_imshow(perfilado_imagen)
cv2.waitKey(0)
cv2.destroyAllWindows()

```



Imagen Original



Imagen Laplaciano



Imagen de Gradiente



Imágenes Perfilado

Conclusiones

En la práctica de detección de bordes y perfilado de una imagen usando los filtros de Sobel y Laplaciano de segundo orden en Python, se ha podido observar que estos métodos son altamente efectivos para realzar los bordes y contornos de una imagen.

El filtro de Sobel permite detectar los bordes horizontales y verticales de una imagen, lo que permite resaltar las líneas y bordes en esas direcciones. Por otro lado, el filtro de Laplaciano de segundo orden detecta cambios en la intensidad de los píxeles en todas las direcciones, lo que permite identificar los bordes y contornos más complejos y curvos.

Además, se sabe que la combinación de ambos filtros puede mejorar aún más la detección de bordes y contornos en una imagen, lo que puede ser de gran utilidad en diversas aplicaciones como el reconocimiento de objetos, la segmentación de imágenes, entre otros.

En conclusión, la detección de bordes y perfilado de una imagen mediante los filtros de Sobel y Laplaciano de segundo orden es una técnica poderosa y efectiva en el procesamiento de imágenes.

digitales, que permite realzar las características más importantes de una imagen y facilitar su análisis y comprensión.

La detección de bordes se utiliza en una variedad de aplicaciones de procesamiento de imágenes, incluyendo la segmentación de imágenes, el reconocimiento de objetos y patrones, la corrección de la distorsión de la imagen y la compresión de imágenes. Existen diferentes técnicas de detección de bordes disponibles, siendo una de las más conocidas el filtro de Sobel, que se utiliza para calcular el gradiente de la imagen en una dirección específica. En general, la detección de bordes es un paso importante en el preprocesamiento de las imágenes antes de realizar otras operaciones , como el análisis y la manipulación de la imagen.

Bibliografía

CorelDraw. (2018). Efectos especiales de perfilado. Recuperado de:
<http://product.corel.com/help/CorelDRAW/540227992/Main/ES/Documentation/wwhelp/wwhimpl/common/html/wwhelp.htm#href=specfx-referece-drpdes.50.12.html&single=true>