1.

با اعمال فیلتر سوبل عمودی، می‌توانید خطوط عمودی در تصویر را شناسایی کنید. فیلتر سوبل عمودی یک فیلتر همگن است که برای تفاضل گرادیان در جهت عمودی استفاده می‌شود. این فیلتر به لبه‌ها و خطوط عمودی در تصویر حساس است و باعث می‌شود خطوط عمودی در تصویر بیشتر برجسته شوند و لبه‌های افقی به صورت کمتری مشخص شوند.

در کد ، ابتدا تصویر را به حالت خاکستری باز می‌کنیم. سپس فیلتر سوبل عمودی را با استفاده از یک آرایه NumPy تعریف می‌کنیم. در این حالت، فیلتر سوبل برای شناسایی خطوط عمودی قرار می‌گیرد.

سپس مقادیر فیلتر سوبل را نرمالیزه کرده و به تصویر اصلی اعمال می‌کنیم. برای این کار از تابع `cv2.filter2D` استفاده می‌کنیم. سپس نتیجه فیلتر را با استفاده از تابع `np.abs` به صورت مثبت در می‌آوریم. در نهایت، تصویر فیلتر شده را نرمالیزه کرده و با استفاده از `matplotlib.pyplot` نمایش می‌دهیم.

در نتیجه، تصویر فیلتر شده حاوی خطوط عمودی برجسته‌تر خواهد بود و لبه‌های افقی کمتری مشخص خواهند شد.

import cv2

import numpy as np

import matplotlib.pyplot as plt

# Load the image

image = cv2.imread("input\_image.jpg", cv2.IMREAD\_GRAYSCALE)

# Define the vertical Sobel filter

sobel\_filter = np.array([[-1, 0, 1],

[-2, 0, 2],

[-1, 0, 1]])

# Normalize the filter

sobel\_filter = sobel\_filter / 4

# Apply the vertical Sobel filter

filtered\_image = np.abs(cv2.filter2D(image.astype(float), -1, sobel\_filter))

# Normalize the filtered image

filtered\_image = cv2.normalize(filtered\_image, None, 0, 255, cv2.NORM\_MINMAX, dtype=cv2.CV\_8U)

# Display the original and filtered images

fig, axs = plt.subplots(1, 2, figsize=(10, 5))

axs[0].imshow(image, cmap='gray')

axs[0].set\_title("Original Image")

axs[0].axis("off")

axs[1].imshow(filtered\_image, cmap='gray')

axs[1].set\_title("Filtered Image (Vertical Lines)")

axs[1].axis("off")

# Show the plot

plt.show()

**نتیجه در نوت بوک ضمیمه شده**

2.

فیلتر گوسی یکی از روش‌های پرکاربرد در پردازش تصویر است که برای اهداف مختلف مورد استفاده قرار می‌گیرد. این فیلتر از یک تابع گوسی برای اعمال اثر خمیدگی به تصویر استفاده می‌کند. علت استفاده از فیلتر گوسی در پردازش تصویر عبارتند از:

1. کاهش نویز:

فیلتر گوسی قابلیت حذف نویز را دارد. با اعمال این فیلتر، نویزهای تصویر که اغلب به صورت نویزهای گوسی به تصویر اضافه می‌شوند، کاهش می‌یابند. این کاهش نویز باعث بهبود کیفیت تصویر و افزایش دقت در پردازش‌های بعدی می‌شود.

2. ایجاد اثر مات کردن:

فیلتر گوسی معمولاً برای ایجاد اثر مات کردن (blurring) در تصویر استفاده می‌شود. این اثر می‌تواند مفید باشد در مواردی مانند حذف جزئیات ناخواسته یا تلفیق صاف و آرام دو تصویر. برای مثال، در حوزه‌ی عکاسی، استفاده از فیلتر گوسی در عکس‌برداری حرکتی می‌تواند بهبود بخشیده و جلوه‌ی ظاهری نرم‌تری را ایجاد کند.

3. استفاده در الگوریتم‌های پردازش تصویر:

فیلتر گوسی در الگوریتم‌های پردازش تصویر مختلف، مانند تشخیص لبه‌ها و تشخیص الگوها، به عنوان یک مرحله‌ی پیش‌پردازش استفاده می‌شود. این فیلتر با کاهش نویز و حذف جزئیات ناهموار، می‌تواند بهبودی در عملکرد الگوریتم‌های بعدی مانند تشخیص الگو و تفکیک شی ها ایجاد کند.

با این حال، استفاده از فیلتر گوسی همیشه منجر به حذف جزئیات نمی‌شود و بسته به پارامترهای استفاده شده و نوع تصویر، می‌تواند تنظیم شود تا جزئیات مورد نظر حفظ شود.

3.

برای جلوگیری از تغییر ابعاد تصویر پس از اعمال فیلتر گوسی، می‌توان از روش‌های زیر استفاده کرد:

1. Padding:

یکی از روش‌های متداول برای حفظ ابعاد تصویر، استفاده از Padding است. در این روش، تصویر با اضافه کردن پیکسل‌های مقداردهی شده به لبه‌ها، به اندازه‌ای گسترش می‌یابد که ابعاد خروجی برابر با ابعاد تصویر اولیه باقی بماند. این روش معمولاً با استفاده از صفر (zero-padding) یا تکرار مقدارهای لبه (replication padding) انجام می‌شود. این کار باعث حفظ اطلاعات و جزئیات در لبه‌ها و نقاط مرزی تصویر می‌شود.

2. Cropping:

یک روش دیگر برای حفظ ابعاد تصویر پس از اعمال فیلتر گوسی، استفاده از Cropping است. در این روش، پس از اعمال فیلتر، تصویر خروجی به ابعاد کوچکتری نسبت به تصویر اولیه می‌رسد. برای بازگرداندن ابعاد به حالت اولیه، می‌توان قسمت‌های لبه یا بخش‌های مهم تصویر را از تصویر خروجی برش داد تا به ابعاد مورد نظر برسد. این روش ممکن است منجر به از دست دادن برخی اطلاعات در لبه‌ها یا نقاط مرزی تصویر شود، اما در برخی موارد ممکن است قابل قبول باشد، به خصوص اگر جزئیات مهم تصویر در مرکز آن قرار دارند.

همچنین، می‌توان از روش‌های دیگر مانند تغییر اندازه تصویر (resizing) یا استفاده از فیلتر گوسی با پارامترهای خاصی که ابعاد تصویر را حفظ کند، استفاده کرد. اما این روش‌ها ممکن است

منجر به از دست دادن جزئیات و کاهش کیفیت تصویر شوند. بنابراین، Padding و Cropping روش‌های مناسب‌تری برای حفظ ابعاد تصویر پس از اعمال فیلتر گوسی هستند.

4.

با عذرخواهی بابت ابهام در پاسخ قبلی، برای بهبود تصویر مات شده، می‌توان از فیلتر آنشارپ (Unsharp) استفاده کرد. فیلتر آنشارپ با افزایش کنتراست و تیزی تصویر، جزئیات را بهبود می‌بخشد. این فیلتر از تفاوت تصویر اصلی با نسخه مات شده استفاده می‌کند.

در زیر، روش استفاده از فیلتر آنشارپ را در پایتون با استفاده از کتابخانه OpenCV نشان می‌دهم:

import cv2

import numpy as np

# خواندن تصویر

image = cv2.imread("input\_image.jpg")

# تبدیل تصویر به فضای رنگی آبی-سبز-قرمز (BGR) به همراه نرمال‌سازی

image = cv2.cvtColor(image, cv2.COLOR\_BGR2RGB) / 255.0

# اعمال فیلتر آنشارپ

blurred = cv2.GaussianBlur(image, (0, 0), sigmaX=10)

sharpened = cv2.addWeighted(image, 1.5, blurred, -0.5, 0)

# نمایش تصویر بهبود یافته

cv2.imshow("Improved Image", sharpened)

cv2.waitKey(0)

cv2.destroyAllWindows()

در این کد، ابتدا تصویر مقداری که مات شده است را با استفاده از `cv2.imread` بارگیری می‌کنیم. سپس تصویر را به فضای رنگی RGB تبدیل کرده و آن را نرمال‌سازی می‌کنیم. سپس با استفاده از `cv2.GaussianBlur`، تصویر را با یک فیلتر گوسی مات می‌کنیم. در نهایت، با استفاده از `cv2.addWeighted`، تصویر اصلی و تصویر مات شده را با وزن‌های مناسب با هم ترکیب می‌کنیم تا تصویر بهبود یافته را بدست آوریم.

توجه داشته باشید که مقادیر وزن‌دهی alpha و beta در cv2.addWeighted می‌توانند تنظیم شوند تا نتیجه بهتری به دست آید. همچنین، مقدار sigmaX در cv2.GaussianBlur نیز می‌تواند تنظیم شود تا شدت فیلتر گوسی تغییر کند.

**نتیجه در نوت بوک ضمیمه شده موجود است.**