DSA0210 Computer Vision with Open CV LAB Experiments

Experiment- 21: Perform Sharpening of Image using Gradient masking.

PROGRAM:

```
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Read the input image
img = cv2.imread(r"D:\New Folder\input.jpeg")

# Check if image is loaded
if img is None:
    raise FileNotFoundError("Image not found. Check the file path.")

# Convert image to grayscale
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)

# Compute Sobel gradients
grad_x = cv2.Sobel(gray, cv2.CV_64F, 1, 0, ksize=3)
grad_y = cv2.Sobel(gray, cv2.CV_64F, 0, 1, ksize=3)

# Compute gradient magnitude
gradient_mag = np.sqrt(grad_x**2 + grad_y**2)

# Normalize gradient
gradient_mag = np.uint8(255 * gradient_mag / np.max(gradient_mag))

# Sharpening factor
```

```
k = 1.0

# Apply gradient masking
sharpened = gray + k * gradient_mag

# Clip values to valid range
sharpened = np.clip(sharpened, 0, 255).astype(np.uint8)

# Display images
plt.figure(figsize=(8, 4))

plt.subplot(1, 2, 1)
plt.imshow(gray, cmap="gray")
plt.title("Original Grayscale Image")
plt.axis("off")

plt.subplot(1, 2, 2)
plt.imshow(sharpened, cmap="gray")
plt.title("Sharpened Image (Gradient Masking)")
plt.axis("off")

plt.tight_layout()
plt.show()
```

OUTPUT:


Figure 1
Original Grayscale Image
Sharpened Image (Gradient Masking)