

 <b>Marwadi University</b>	<b>Marwadi University</b> <b>Faculty of Engineering and Technology</b> <b>Department of Information and Communication Technology</b>	
<b>Subject: DSIP (01CT0513)</b>	<b>AIM: Perform gray level operations images.</b>	
<b>Image processing _ 7</b>	<b>Date:</b>	<b>Enrollment No: 92410133034</b>

**AIM: Perform gray level operations images.**

**Theory :-**

### **What is a Grayscale Image?**

A **grayscale image** is a type of image where each pixel represents only the **intensity of light (brightness)** — not color.

- Pixel values range from **0 to 255** in 8-bit images:
  - **0 = Black**
  - **255 = White**
  - Values in between represent varying shades of gray.
- Grayscale simplifies computation and is widely used in:
  - **Image analysis**
  - **Object detection**
  - **Medical imaging**
  - **Machine learning preprocessing**

### **1. Grayscale Image**

#### **Theory:**

- A **grayscale image** is composed of shades of gray, with **no color information**.
- Each pixel has a single intensity value ranging from **0 (black)** to **255 (white)**.
- It simplifies image processing tasks and reduces computational load.

### **2. Image Negation**

#### **Theory:**

- Image negation transforms each pixel to its **inverse intensity**.
- It is used to **highlight details** in dark regions and create photographic negatives.

### **3. Image Thresholding**

#### **Theory:**

- Thresholding is a method of **segmenting** an image by converting it into a **binary image**.
- All pixels **above** a certain threshold become **white**, and **below** become **black**.
- Useful in separating **objects from the background**.

 <b>Marwadi University</b>	<b>Marwadi University</b> <b>Faculty of Engineering and Technology</b> <b>Department of Information and Communication Technology</b>	
<b>Subject: DSIP (01CT0513)</b>	<b>AIM: Perform gray level operations images.</b>	
<b>Image processing _ 7</b>	<b>Date:</b>	<b>Enrollment No: 92410133034</b>

#### 4. Gamma Correction

 **Theory:**

- Gamma correction adjusts the **brightness** of an image **non-linearly**.
- It is used to **enhance details** in dark or bright regions without affecting contrast.

**Program :-**

```

import cv2
import numpy as np
from google.colab.patches import cv2_imshow

# Load the input grayscale image
image = cv2.imread('/content/3fa6bc6d-ae9c-4f37-af02-777c9211267a.jpg',
cv2.IMREAD_GRAYSCALE)

# Check if the image is loaded successfully
if image is None:
    print("Error: Could not open or find the image.")
    exit()

# Function to perform image negation
def image_negation(input_image):
    negated_image = 255 - input_image
    return negated_image

# Function to perform image thresholding
def image_thresholding(input_image, threshold_value):
    _, thresholded_image = cv2.threshold(input_image, threshold_value, 255, cv2.THRESH_BINARY)
    return thresholded_image

# Function to perform image gamma correction
def image_gamma_correction(input_image, gamma):
    gamma_corrected_image = np.power(input_image / 255.0, gamma) * 255.0
    gamma_corrected_image = np.uint8(gamma_corrected_image)
    return gamma_corrected_image

# Perform gray level operations
negated_image = image_negation(image)
thresholded_image = image_thresholding(image, 128)
gamma_corrected_image = image_gamma_correction(image, 1.5)

# Display the original and processed images
cv2_imshow(image)
cv2_imshow(negated_image)

```



**Subject: DSIP (01CT0513)**

**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:**

**Enrollment No: 92410133034**

```
cv2_imshow(thresholded_image)
cv2_imshow(gamma_corrected_image)
```

```
# Wait for a key press and then close the windows
cv2.waitKey(0)
cv2.destroyAllWindows()
```

program :- 2

```
import cv2
import numpy as np
from google.colab.patches import cv2_imshow

# Load the input grayscale image
image = cv2.imread('/content/apple.jpg', cv2.IMREAD_GRAYSCALE)

# Check if the image is loaded successfully
if image is None:
    print("Error: Could not open or find the image.")
    exit()

# Function to perform image negation
def image_negation(input_image):
    return 255 - input_image

# Function to perform image thresholding with a given type
def image_thresholding(input_image, threshold_value, threshold_type):
    _, thresholded_image = cv2.threshold(input_image, threshold_value, 255, threshold_type)
    return thresholded_image

# Function to perform image gamma correction
def image_gamma_correction(input_image, gamma):
    gamma_corrected_image = np.power(input_image / 255.0, gamma) * 255.0
    gamma_corrected_image = np.uint8(gamma_corrected_image)
    return gamma_corrected_image

# Perform gray level operations
negated_image = image_negation(image)

# Apply different threshold types
thresh_binary = image_thresholding(image, 128, cv2.THRESH_BINARY)
thresh_binary_inv = image_thresholding(image, 128, cv2.THRESH_BINARY_INV)
thresh_trunc = image_thresholding(image, 128, cv2.THRESH_TRUNC)
thresh_tozero = image_thresholding(image, 128, cv2.THRESH_TOZERO)
thresh_tozero_inv = image_thresholding(image, 128, cv2.THRESH_TOZERO_INV)

# Gamma correction
```



**Subject: DSIP (01CT0513)**

**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:** Enrollment No: 92410133034

```
gamma_corrected_image = image_gamma_correction(image, 1.5)
```

```
# Display all results
print("Original Grayscale Image:")
cv2_imshow(image)
```

```
print("Negated Image:")
cv2_imshow(negated_image)
```

```
print("Thresholding - Binary (THRESH_BINARY):")
cv2_imshow(thresh_binary)
```

```
print("Thresholding - Binary Inverted (THRESH_BINARY_INV):")
cv2_imshow(thresh_binary_inv)
```

```
print("Thresholding - Trunc (THRESH_TRUNC):")
cv2_imshow(thresh_trunc)
```

```
print("Thresholding - To Zero (THRESH_TOZERO):")
cv2_imshow(thresh_tozero)
```

```
print("Thresholding - To Zero Inverted (THRESH_TOZERO_INV):")
cv2_imshow(thresh_tozero_inv)
```

```
print("Gamma Corrected Image ( $\gamma$  = 1.5):")
cv2_imshow(gamma_corrected_image)
```

```
# cv2.waitKey(0)
# cv2.destroyAllWindows()
```

Output :-



**Subject: DSIP (01CT0513)**

**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:**

**Enrollment No: 92410133034**





**Subject: DSIP (01CT0513)**

**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:**

**Enrollment No: 92410133034**

Negated Image:





**Subject: DSIP (01CT0513)**

**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:**

**Enrollment No: 92410133034**

Thresholding - Binary (THRESH\_BINARY):





**Subject: DSIP (01ICT0513)**

**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:**

**Enrollment No: 92410133034**

Thresholding - Binary Inverted (THRESH\_BINARY\_INV):





**Subject: DSIP (01CT0513)**

**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:**

**Enrollment No: 92410133034**

Thresholding - Trunc (THRESH\_TRUNC):





**Subject: DSIP (01CT0513)**

**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:**

**Enrollment No: 92410133034**

Thresholding - To Zero (THRESH\_TOZERO):





**Subject: DSIP (01CT0513)**

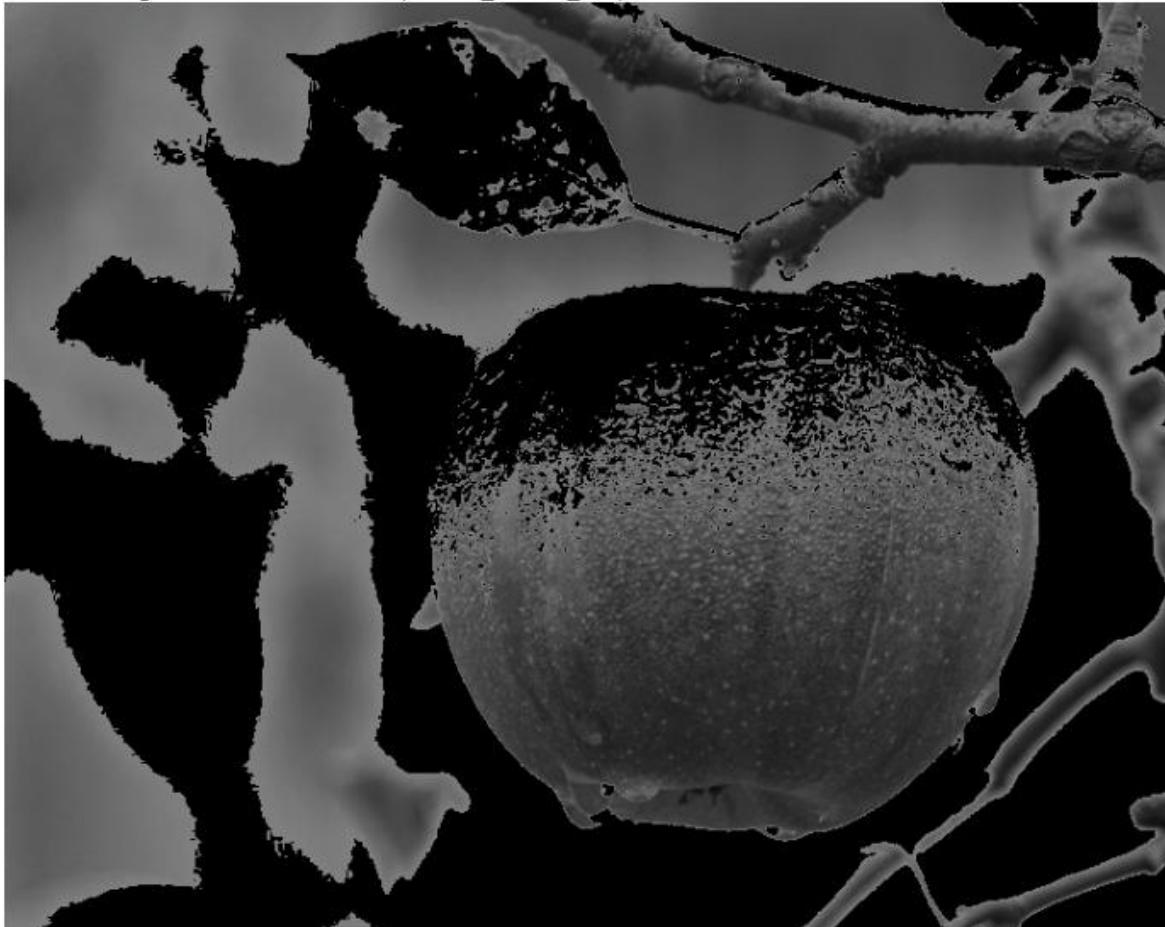
**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:**

**Enrollment No: 92410133034**

Thresholding - To Zero Inverted (THRESH\_TOZERO\_INV):





**Subject: DSIP (01CT0513)**

**AIM: Perform gray level operations images.**

**Image processing \_ 7**

**Date:**

**Enrollment No: 92410133034**

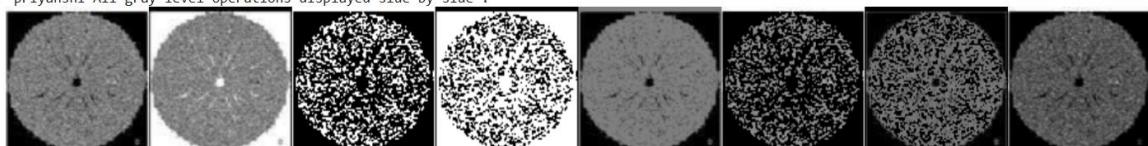
Gamma Corrected Image ( $\gamma = 1.5$ ):



priyanshi All gray level operations displayed side by side :



priyanshi All gray level operations displayed side by side :



 <b>Marwadi University</b>	<b>Marwadi University</b> <b>Faculty of Engineering and Technology</b> <b>Department of Information and Communication Technology</b>	
<b>Subject: DSIP (01CT0513)</b>	<b>AIM: Perform gray level operations images.</b>	
<b>Image processing _ 7</b>	<b>Date:</b>	<b>Enrollment No: 92410133034</b>

Output observation :-

**1. Original Grayscale Image**

- Shows the base image in various shades of gray (0–255).
- Used as the reference for all transformations.

**2. Negated Image**

- All pixel intensities are inverted:
  - Dark regions become light.
  - Light regions become dark.
- Helps to enhance **hidden details** in dark areas.

**3. Thresholding – cv2.THRESH\_BINARY**

- Pixels  $> 128 \rightarrow$  White (255),  $\leq 128 \rightarrow$  Black (0).
- Effectively segments bright areas from the background.
- Converts the image into **pure black and white**.

**4. Thresholding – cv2.THRESH\_BINARY\_INV**

- Inverse of THRESH\_BINARY.
- Bright areas become black; dark areas become white.
- Useful for **highlighting dark objects** on a bright background.

**5. Thresholding – cv2.THRESH\_TRUNC**

- Pixels  $> 128$  are clipped to 128.
- Pixels  $\leq 128$  stay unchanged.
- Flattens the bright areas, preserving darker details.

**6. Thresholding – cv2.THRESH\_TOZERO**

- Keeps pixels  $> 128$  unchanged.
- Sets pixels  $\leq 128$  to 0 (black).
- Retains **bright structures**, removes shadows/dark noise.

**7. Thresholding – cv2.THRESH\_TOZERO\_INV**

- Opposite of TOZERO.
- Keeps pixels  $\leq 128$  unchanged.
- Sets pixels  $> 128$  to 0.
- Isolates **darker structures** in the image.

**8. Gamma Correction ( $\gamma = 1.5$ )**

- Enhances **contrast in darker regions**, compresses brightness.
- Makes the image appear slightly darker and details in bright areas softer.
- Useful in lighting correction and visual enhancement.

 **Overall Understanding:**

- Each operation highlights **different aspects** of the image:
  - **Negation** for contrast reversal.
  - **Thresholding** for binary segmentation and isolation.

 <b>Marwadi University</b>	<b>Marwadi University</b> <b>Faculty of Engineering and Technology</b> <b>Department of Information and Communication Technology</b>
<b>Subject: DSIP (01CT0513)</b>	<b>AIM: Perform gray level operations images.</b>
<b>Image processing _ 7</b>	<b>Date:</b> <b>Enrollment No: 92410133034</b>

- **Gamma** for brightness and contrast adjustment.
- These are fundamental tools in **image preprocessing, object detection, and enhancement** in computer vision tasks.