Nagar Yuwak Shikshan Sanstha's

Yeshwantrao Chavan College of Engineering

(An Autonomous Institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University)

Hingna Road, Wanadongri, Nagpur - 441 110

NAAC A++

Ph.: 07104-237919, 234623, 329249, 329250 Fax: 07104-232376, Website: www.ycce.edu

Department of Computer Technology

Vision of the Department

To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration.

Mission of the Department

To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies.

Session 2025-2026

Vision: Dream of where you want.	**Mission:** Means to achieve Vision

Program Educational Objectives of the program (PEO): (broad statements that describe the professional and career accomplishments)

PEO1	**Preparation**	**P: Preparation**	**Pep-CL abbreviation pronounce as Pep-si-lL easy to recall**
PEO2	**Core Competence**	**E: Environment (Learning Environment)**	
PEO3	**Breadth**	**P: Professionalism**	
PEO4	**Professionalism**	**C: Core Competence**	
PEO5	**Learning Environment**	**L: Breadth (Learning in diverse areas)**	

Program Outcomes (PO): (statements that describe what a student should be able to do and know by the end of a program)

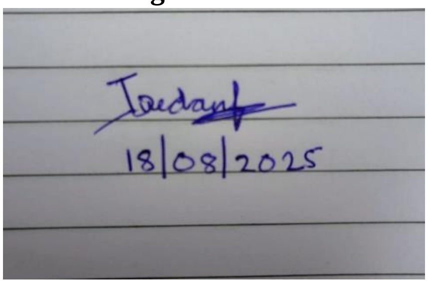
Keywords of POs:
Engineering knowledge, Problem analysis, Design/development of solutions, Conduct Investigations of Complex Problems, Engineering Tool Usage, The Engineer and The World, Ethics, Individual and Collaborative Team work, Communication, Project Management and Finance, Life-Long Learning

PSO Keywords: Cutting edge technologies, Research

"I am an engineer, and I know how to apply engineering knowledge to investigate, analyse and design solutions to complex problems using tools for entire world following all ethics in a collaborative way with proper management skills throughout my life." <u>*to contribute to the development of cutting-edge technologies and Research*</u>.

Integrity: I will adhere to the Laboratory Code of Conduct and ethics in its entirety.

Name and Signature of Student and Date



Vision of the Department

To be a well-known centre for pursuing computer education through innovative pedagogy, value-based education and industry collaboration.

Mission of the Department

To establish learning ambience for ushering in computer engineering professionals in core and multidisciplinary area by developing Problem-solving skills through emerging technologies.

Session	**2025-26 (ODD)**	**Course Name**	**Computer vision Lab**
Semester	**5**	**Course Code**	**CT**
Roll No	**81**	**Name of Student**	**Vedant Jiwanapurkar**

Practical Number	**1**
Course Outcome	**Upon successful completion of the course the students will be able to** 1. Apply image enhancement and smoothing techniques to improve image quality for further analysis. 2. Extract meaningful features from images using descriptors such as HOG and SIFT. 3. Implement and evaluate modern object detection methods including YOLO and R-CNN. 4. Analyze and develop solutions for motion estimation, object recognition, and facial expression recognition using classical and learning-based methods.
Aim	**Implement various gray level transformations for image enhancement.**
Problem Definition	**Conversion of original image into:** 1. Negative Transformation 2. Logarithmic Transformation 3. Power Law Transformation
Theory (100 words)	**Image enhancement** is the most fundamental and simple process of digital image processing. In this process, the intensity level of an image is manipulated to get a better output image. For this purpose, we will use the mathematical operation of grey level transformation, also known as **intensity transformation.** Image enhancement in the spatial domain is based on the following expression: **g(x. y) = T[f(x, y)]** where, f (x, y) is the input image, g(x, y) is the output image and T is an operator on f defined over the neighborhood of point (x, y). **1. Negative Image** The negative of a image having intensity range [0, L - 1] can be found using the following transformation: $s = T(r) = L - r - 1$ **2. Log Transformation** The log transformation is of the form: $s = T(r) = c \log (1 + r)$ where, c is constant and $r \geq 0$. **3. Power Law (Gamma) Transformation** The power law (or gamma) transformation is of the form: $s = T(r) = c * r^{\gamma}$ where, c, γ are constants; c, γ > 0

Procedure and Execution

(100 Words)

Algorithm:

In **Negative Transformation**, first we read the input image. Then we check the maximum intensity value, which is 256 for an 8-bit image. For every pixel, we apply the formula **$s=(L-1)-r$** This changes dark areas into bright and bright areas into dark. After replacing all pixel values, the negative image is ready.

In **Logarithmic Transformation**, we start with the input image and select a scaling constant ccc. For each pixel, we calculate the new value using **$s=c \cdot \log(1+r)$** This increases the details in dark regions and reduces the effect of very bright pixels. Finally, the values are normalized and the output image is obtained.

In **Power-Law (Gamma) Transformation**, we take the image and choose two values, constant ccc and gamma γ. Then for every pixel. If gamma is less than 1, the image becomes brighter, and if gamma is greater than 1, the image becomes darker. After normalization, the gamma corrected image is displayed.

Code:

```
clc;
clear;
close all;

img = imread('C:\Users\mithi\OneDrive\ドキュメント\MATLAB\macos-big-sur-apple-layers-fluidic-colorful-wwdc-stock-4096x2304-1455.jpg');
img = im2double(img);
```

DISPLAY ORIGINAL IMAGE

```
figure('Name', 'Original Image');
imshow(img);
title('Original Image');
gray_img = rgb2gray(img);
```

NEGATIVE TRANSFORMATION

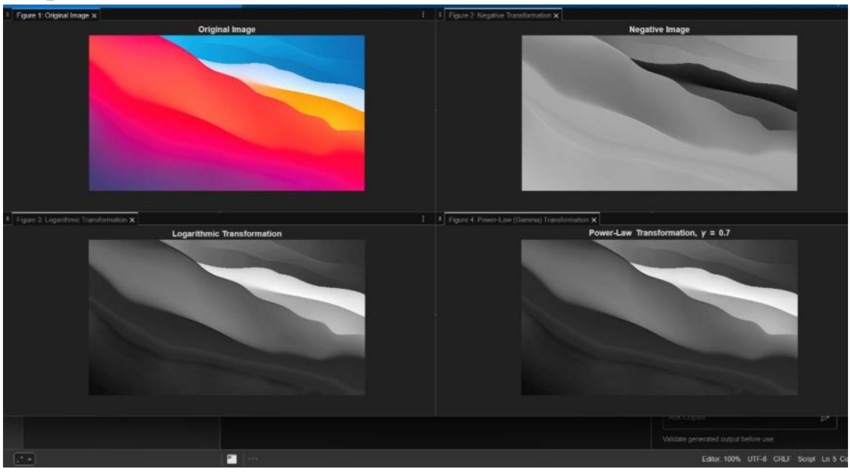
```
negative_img = 1 - gray_img;
figure('Name', 'Negative Transformation');
imshow(negative_img);
title('Negative Image');
```

LOGARITHMIC TRANSFORMATION

```
c = 1;
log_img = c * log(1 + gray_img);
log_img = (log_img - min(log_img(:))) / (max(log_img(:)) - min(log_img(:)));
figure('Name', 'Logarithmic Transformation');
imshow(log_img);
title('Logarithmic Transformation');
```

POWER_LAW TRANSFORMATION

```
gamma = 0.7;
c = 1;
power_law_img = c * (gray_img .^ gamma);
power_law_img = (power_law_img - min(power_law_img(:))) / (max(power_law_img(:)) - min(power_law_img(:)));
figure('Name', 'Power-Law (Gamma) Transformation');
imshow(power_law_img);
title(['Power-Law Transformation, \gamma = ', num2str(gamma)]);
```

	Output: 
Output Analysis	In the **Negative Transformation**, the picture looks opposite, like a photo negative. Dark places become bright and bright places become dark. In the **Logarithmic Transformation**, the dark parts of the picture become more clear. Very bright parts become less shiny. It helps to see small details in the dark. In the **Power-Law (Gamma) Transformation**, the picture changes based on gamma. If gamma is small, the picture looks brighter. If gamma is big, the picture looks darker. This is useful to adjust brightness and contrast on screens.
Link of student Github profile where lab assignment has been uploaded.	
Conclusion	From these transformations, we learn that by changing pixel values in simple ways, we can improve images for better understanding. Each method has its own role: one shows hidden details, another balances brightness, and another controls contrast. Together, they make image processing more powerful and useful in real life.
Plag Report (Similarity index < 12%)	
Date	**18/08/2025**