Experiment no: 2

1. **Aim:** TO study various pixel-wise and transformation.

2. **Software Tool Used:**

MATLAB is a high-performance language and interactive environment for numerical computing, algorithm development, and data analysis. It's designed for matrix and array operations, data visualization, and application creation. MATLAB is widely used in fields like engineering, science, and finance.

3. **Theory:**

In image processing, pixel-wise transformations are operations that modify the pixel intensity values of an image. These transformations are often used to enhance or adjust the image's appearance.

1. Linear transformation: Linear transformations are the simplest forms of pixel-wise transformations and are applied directly to the intensity values of the pixels.

Identity transformation: This transformation leaves the image unchanged. Each pixel intensity 'r' remains the same, so the output image 's' is identical to the input image.

Formula: 's = r'

Negative transformation: This transformation inverts the intensity of each pixel. Darker pixels become lighter, and lighter pixels become darker. It is often used for enhancing features in an image.

Formula: s = 'L - 1 – r' Where L = 2^N (N is no of bits required for each pixel)

2. Logarithmic Transform: Logarithmic transformations map a narrow range of low-intensity values in the input image to a wider range of output levels. This transformation is useful for enhancing details in dark regions of the image while compressing the dynamic range of brighter regions.

Formula: s = c * log (1 + r) where c is a constant**.**

3.Exponential Transformation: Exponential transformations emphasize higher intensity values while compressing lower intensity values. This can help in enhancing brighter areas of the image.

Formula s = c * (exp(r) - 1) where c is a constant.

4.Power-Law (Gamma) Transformation: Power-law transformations raise the pixel intensity values to a power of γ. Depending on the value of γ, this can either enhance or reduce the contrast in different intensity ranges.

- If $\gamma<1$, the transformation enhances the image's intensity, especially in the dark regions.
- If $\gamma>1$, the trans. compresses the intensity values, especially in the bright regions.

Formula s = c * r^γ. Where c is a constant**.**

4. Result:

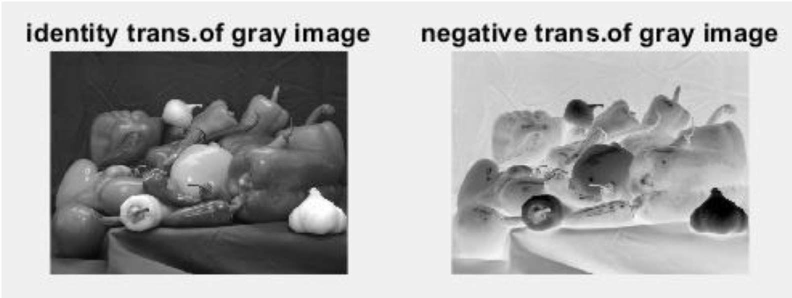


Fig (1): identity and negative transformation of Gray image "peppers.png".

Fig (2): identity and negative transformation of binary image "peppers.png".

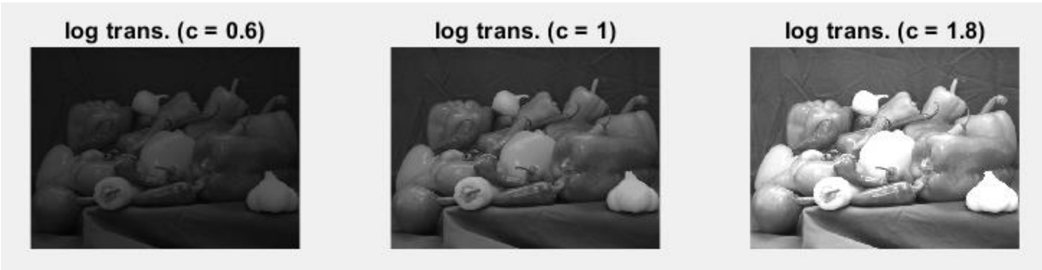


Fig (3): logarithmic transformation of Gray image "peppers.png".

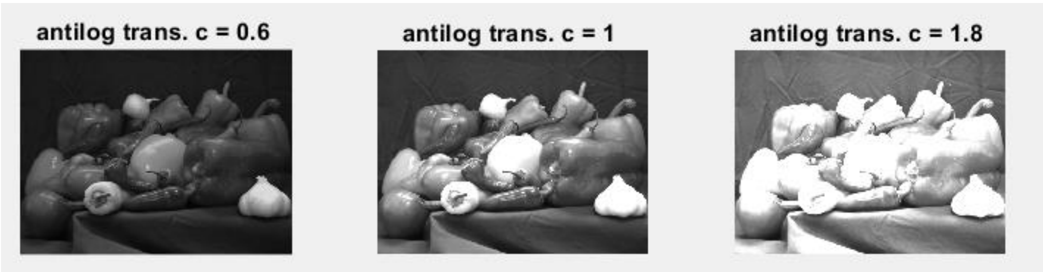


Fig (4): anti-logarithmic transformation of Gray image "peppers.png".

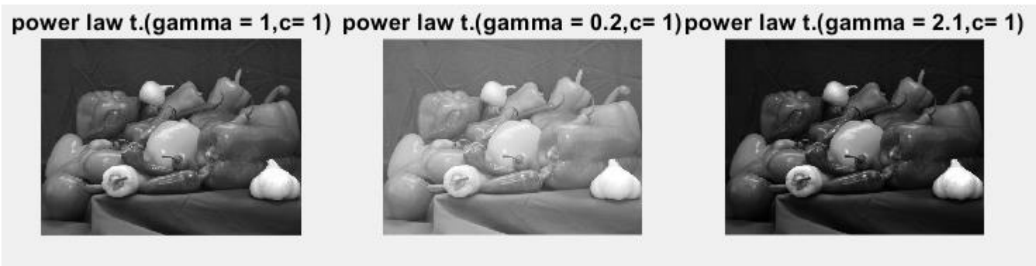


Fig (5): power law transformation of Gray image "peppers.png".

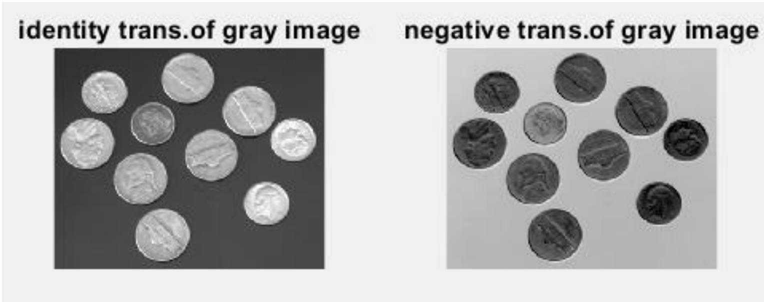


Fig (6): identity and negative transformation of Gray image “coins.png”.

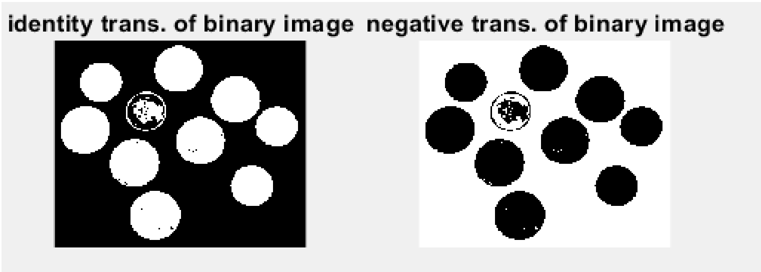


Fig (7): identity and negative transformation of binary image “coins.png”.

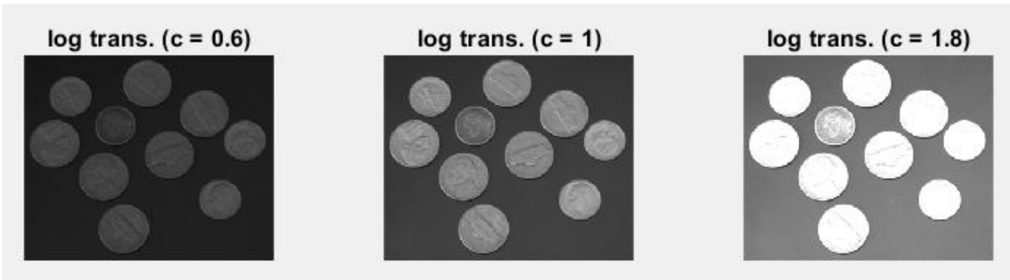


Fig (8): logarithmic transformation of Gray image “coins.png”

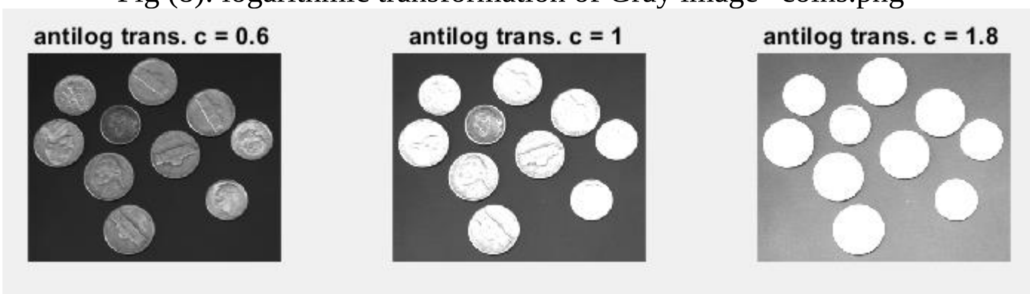
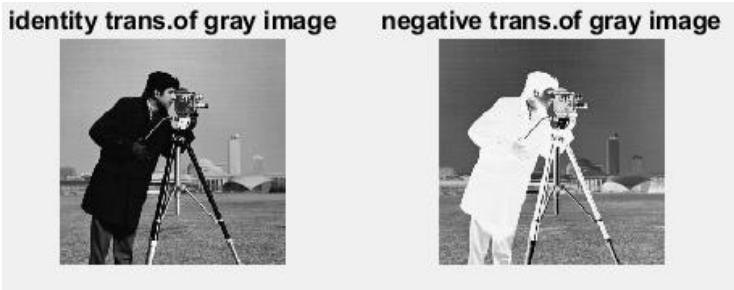


Fig (9):anti-logarithmic transformation of Gray image “coins.png”

Fig(10): power law transformation of Gray image “coins.png”.

Fig(11): identity and negative transformation of Gray image “cameraman.tif”.



Fig(12): identity and negative transformation of binary image “cameraman.tif”.

Fig(13): logarithmic transformation of Gray image “cameraman.tif”.

Fig(14): anti-logarithmic transformation of Gray image “cameraman.tif”.

Fig(15): power law transformation of Gray image “cameraman.tif”.

5. **Discussion:**

The experiment on pixel-wise transformations using different images (coins.png, cameraman.tif, and peppers.png) demonstrates the effects of various transformation techniques on image enhancement and manipulation.

Linear transformation: in the fig (1), fig (6), fig (11) the left picture shows the identity transformation because spit copy of the same but in the right picture is totally changed in the pixel intensity where the darker part made more brighter and the brighter part gone darker. It happened the same thing with the black and white picture in the fig (2), fig (7) and fig (12) where the left picture is directly opposed of the right picture where the lighter part changed into dark, and the darker part changed into lighter.

Logarithmic transformation: in the fig (3), fig (8) and fig (13) with the respectively change in the value of c it made change in the intensity of the picture. With a low value of c like the 0.6 or o.5 (the leftist picture), it reduces the effect of the log transformation, resulting in more subtle contrast enhancements and less impact on the image's brightness. If c=1The transformation results in moderate contrast enhancement, focusing primarily on dark regions. But with the higher value like c = 1.5, c =1.8 (the rightest picture) increases the brightness of the image, making the transformation more aggressive by stretching the lower intensity values more. The entire image appears brighter, with more pronounced detail in the dark areas.

Anti-Logarithmic transformation: in the fig (4), fig (9) and fig (14) the leftist picture with the low c value like 0.6 , 0.4 etc, the bright regions are slightly enhanced while retaining the original contrast but with the high value of c like 2 , 1.8 ,5 ,10 (like the rightest picture) The bright regions become intensely highlighted, and the image gains a more dramatic contrast, especially in lighter areas. The constant ccc controls the degree to which bright regions are emphasized. A higher ccc results in a brighter output image by expanding the intensity values, especially in the high-intensity regions. Lowering ccc reduces the enhancement effect, maintaining a closer resemblance to the original image but with a slight emphasis on bright areas.

Power-Law (Gamma) Transformation: in the fig (5), fig (10), fig (15), When c = 1, the transformation mainly focuses on changing the contrast. Where γ= 0.2, the darker parts of the image become much brighter, making details in shadows more visible. Where γ=2.1, the brighter areas are darkened, which helps reduce the intensity of overexposed regions. where c =2, the transformation not only changes the contrast but also makes the entire image brighter. For instance, with γ=0.5, the dark areas are brightened even more than with c = 1, and the whole image looks brighter. On the other hand, with γ=2.5, the bright parts of the image are toned down, but the overall image still appears brighter because of the higher ccc value.

In simple terms, c controls how bright the entire image is, while γ changes how light or dark certain areas appear. Lower γ values brighten dark areas, and higher values darken bright areas.

6. **Conclusion:**

In this experiment, explored several pixel-wise transformations, including linear (identity, negative), logarithmic, exponential, and power-law transformations, to understand their effects on image enhancement. Linear transformations allowed for basic intensity adjustments, with negative inverting the contrast of the image. Logarithmic and exponential transformations enhanced dark and bright regions respectively, proving useful for images with varying intensity distributions. Power-law transformation, with its gamma adjustment, enabled fine-tuning of brightness and contrast. These transformations demonstrated their importance in revealing hidden details and optimizing image quality for various applications.

7. **Reference:**

1.Gonzalez, R.C., & Woods, R.E. (2018). *Digital Image Processing* (4th ed.). Pearson.
2.Pratt, W.K. (2007). *Digital Image Processing: PIKS Scientific Inside* (4th ed.). Wiley-Interscalene .

8. Code:

Code 1: subplot(4,4,1); x = imread("peppers.png"); r = rgb2gray(x); s = r ; imshow(s); title("identity trans.of gray image");
subplot(4,4,2); n = (255- r); imshow(n); title("negative trans.of gray image");
subplot(4,4,3); s = im2bw(r) ; imshow(s); title("identity trans. of binary image");
subplot(4,4,4); n = (1- im2bw(r)); imshow(n); title("negative trans. of binary image");
d =(double(r)/255) ;
subplot(4,4,5); u = 0.5 * log(d+1); imshow(u); title("log trans. (c = 0.6)");
subplot(4,4,6); u = 1 * log(d+1); imshow(u); title("log trans. (c = 1)");
subplot(4,4,7); u = 2 * log(d+1); imshow(u); title("log trans. (c = 1.8)");
subplot(4,4,9); v = 0.5 * (exp(d)-1); imshow(v); title("antilog trans. c = 0.6 ");
subplot(4,4,10); v = 1 * (exp(d)-1); imshow(v); title("antilog trans. c = 1 ");
subplot(4,4,11); v = 2 * (exp(d)-1); imshow(v); title("antilog trans. c = 1.8 ");
subplot(4, 4,13); x = 1*(d.^1); imshow(x); title("power law t.(gamma = 1,c= 1)");
subplot(4,4,14); x = 1 * (d.^0.5); imshow(x); title("power law t.(gamma = 0.2,c= 1)");
subplot(4,4,15); x = 1*(d.^1.5); imshow(x); title("power law t.(gamma = 2.1,c= 1)");

code 2: subplot(4,4,1); r = imread("coins.png"); s = r ; imshow(s); title("identity trans.of gray image");
subplot(4,4,2); n = (255- r); imshow(n); title("negative trans.of gray image");
subplot(4,4,3); s = im2bw(r) ; imshow(s); title("identity trans. of binary image");
subplot(4,4,4); n = (1- im2bw(r)); imshow(n); title("negative trans. of binary image");
d =(double(r)/255) ;
subplot(4,4,5); u = 0.5 * log(d+1); imshow(u); title("log trans. (c = 0.6)");
subplot(4,4,6); u = 1 * log(d+1); imshow(u); title("log trans. (c = 1)");
subplot(4,4,7); u = 2 * log(d+1); imshow(u); title("log trans. (c = 1.8)");
subplot(4,4,9); v = 0.5 * (exp(d)-1); imshow(v); title("antilog trans. c = 0.6 ");
subplot(4,4,10);v = 1 * (exp(d)-1); imshow(v); title("antilog trans. c = 1 ");
subplot(4,4,11); v = 2 * (exp(d)-1); imshow(v); title("antilog trans. c = 1.8 ");
subplot(4, 4,13); x = 1*(d.^1); imshow(x); title("power law t.(gamma = 1,c= 1)");
subplot(4,4,14); x = 1 * (d.^0.5); imshow(x); title("power law t.(gamma = 0.2,c= 1)");
subplot(4,4,15); x = 1*(d.^1.5); imshow(x); title("power law t.(gamma = 2.1,c= 1)");

code 3: subplot(4,4,1); r = imread("cameraman.tif"); s = r ; imshow(s); title("identity trans.of gray image");
subplot(4,4,2); n = (255- r); imshow(n); title("negative trans.of gray image");
subplot(4,4,3); s = im2bw(r); imshow(s); title("identity trans. of binary image");
subplot(4,4,4); n = (1- im2bw(r)); imshow(n); title("negative trans. of binary image");
d =(double(r)/255) ;
subplot(4,4,5); u = 0.5 * log(d+1); imshow(u); title("log trans. (c = 0.6)");
subplot(4,4,6); u = 1 * log(d+1); imshow(u); title("log trans. (c = 1)");
subplot(4,4,7); u = 2 * log(d+1); imshow(u); title("log trans. (c = 1.5)");
subplot(4,4,9); v = 0.5 * (exp(d)-1); imshow(v); title("antilog trans.(c = 0.8) ");
subplot(4,4,10); v = 1 * (exp(d)-1); imshow(v); title("antilog trans. (c = 1) ");
subplot(4,4,11); v = 2 * (exp(d)-1); imshow(v); title("antilog trans. (c = 1.4) ");
subplot(4, 4,13); x = 1*(d.^1); imshow(x); title("power law t.(gamma = 1,c= 2)");
subplot(4,4,14); x = 1 * (d.^0.5); imshow(x); title("power law t.(gamma = 0.5,c= 2)");subplot(4,4,15); x = 1*(d.^1.5); imshow(x); itle("power law t.(gamma = 2.5,c= 2)");