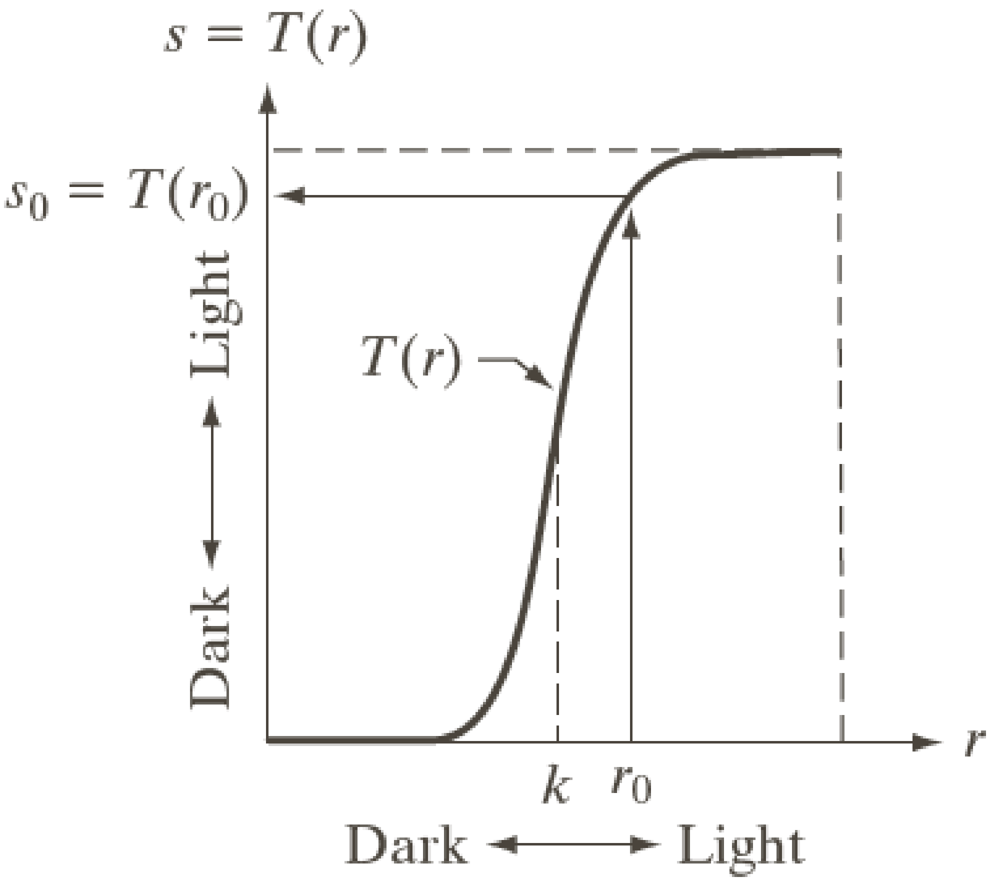
Intensity Transformation

Sung Soo Hwang

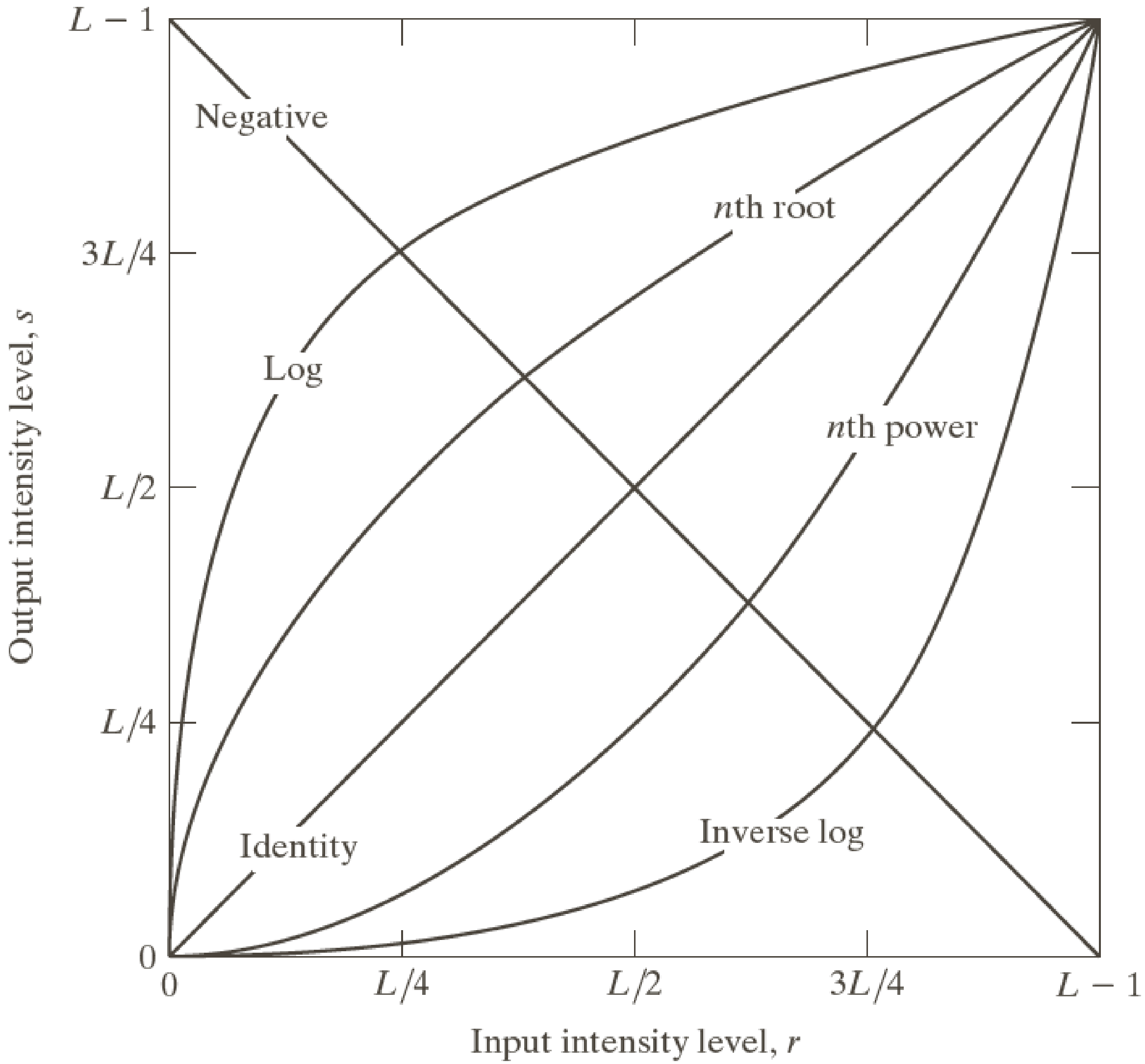
Intensity transformation

- Definition
  - Process of mapping each intensity value of an input image into the corresponding output intensity value through mathematical expression



Intensity transformation

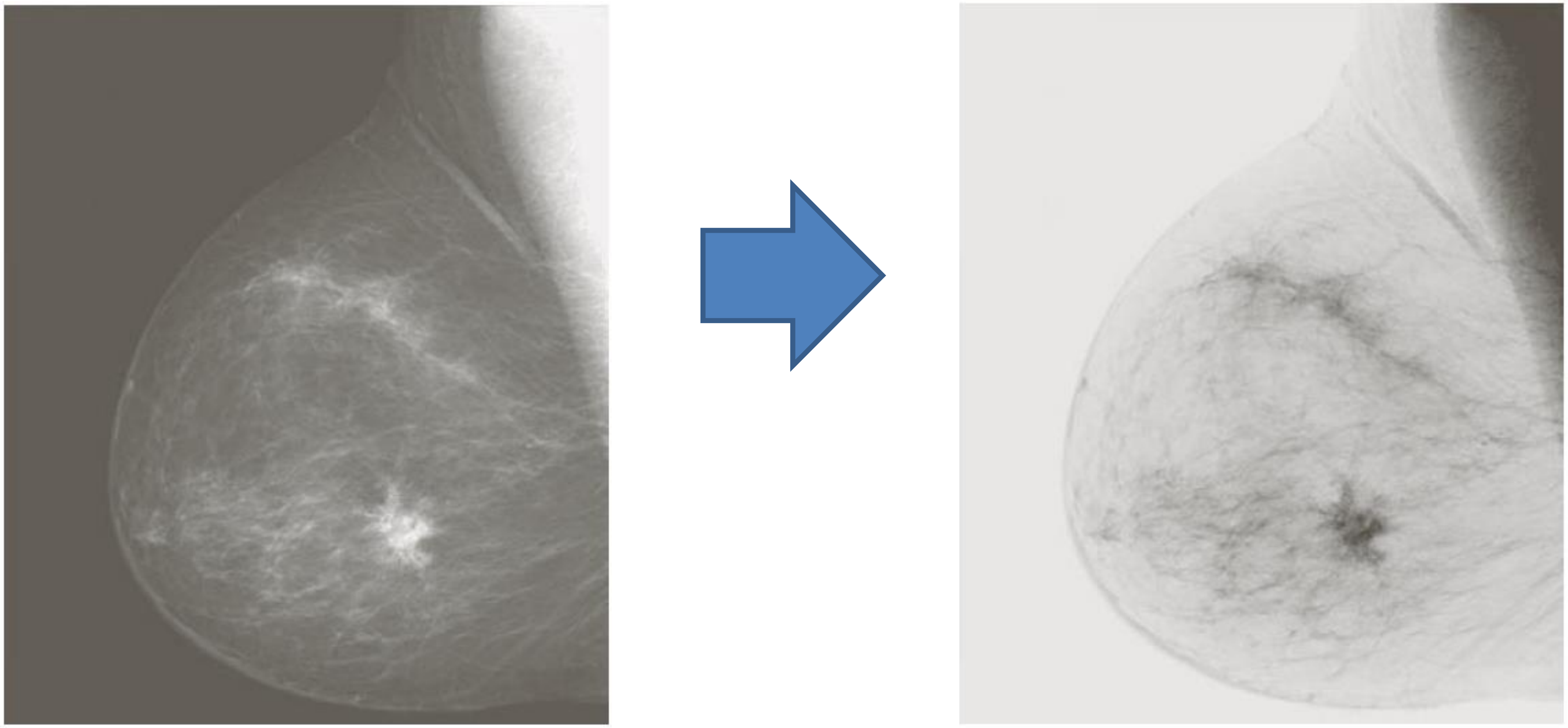
- Example of intensity transformation



Intensity transformation

- Image negatives
  - When the intensity level of an image is [0,L-1], the negative of the image is

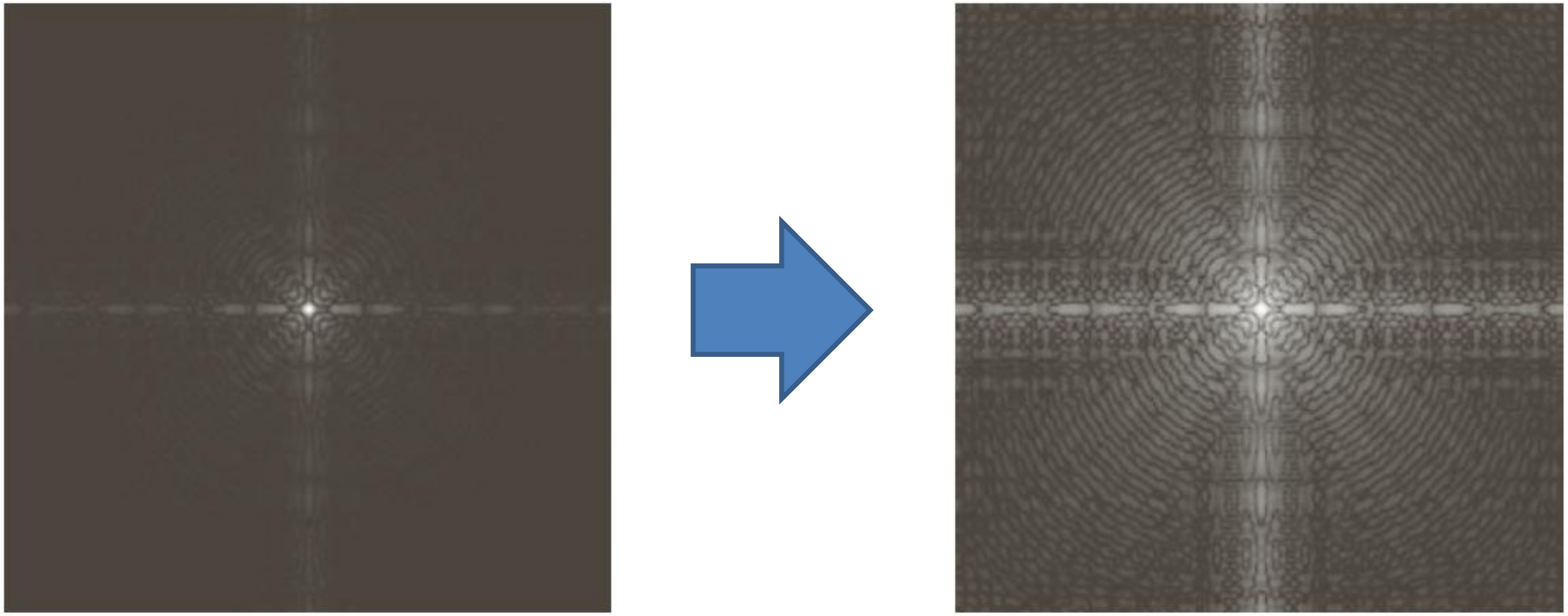
    $s = L-1-r$ s: output r: input
  - It is suited for enhancing white or gray detail embedded in dark regions of an image



Intensity transformation

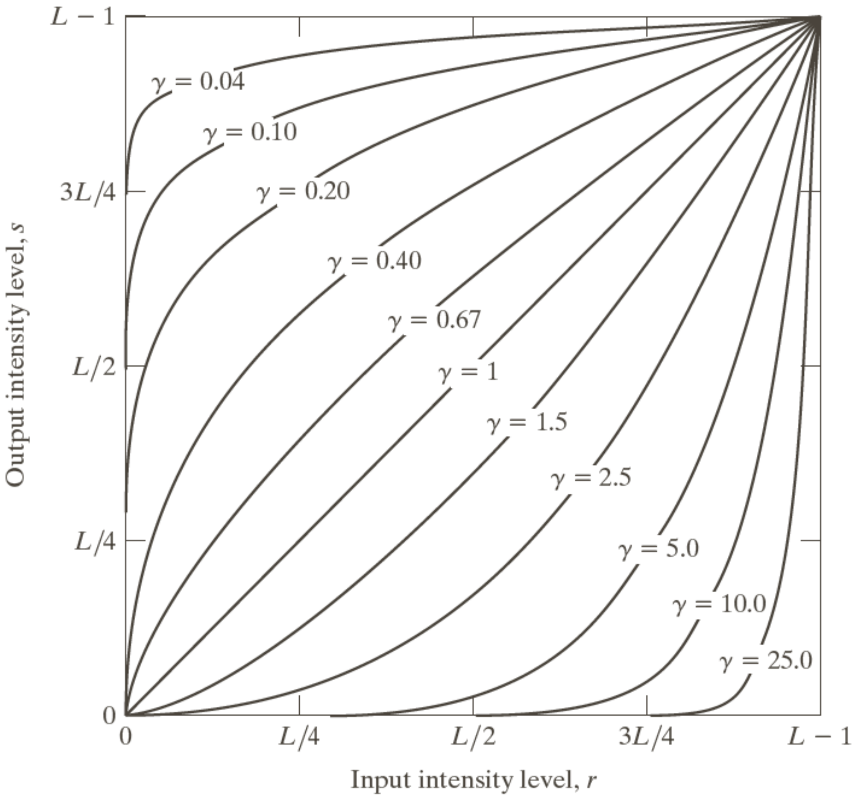
- Log transformation
  - s=c log(1+r) , c: constant, r: input, s: output
  - It maps a narrow range of low intensity values into a wider range of output levels

    ➔ It enhances contrast of dark region
  - The opposite is true of higher values of input levels



Intensity transformation

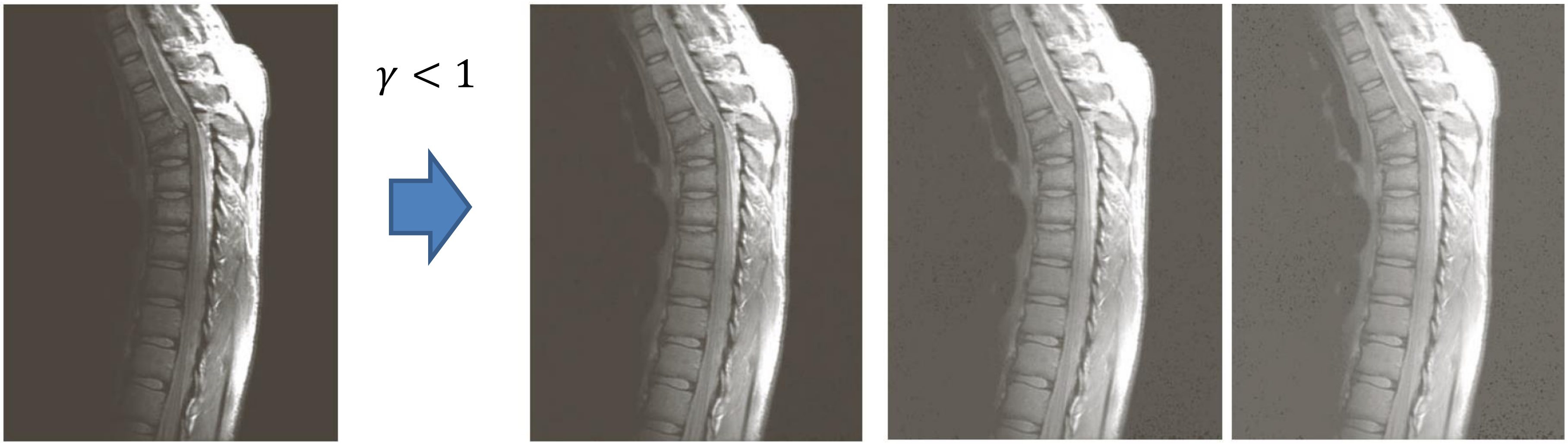
- Power-Law (Gamma) transformation
  - $s = cr^{\gamma}$ c:constant, s:output, r:input

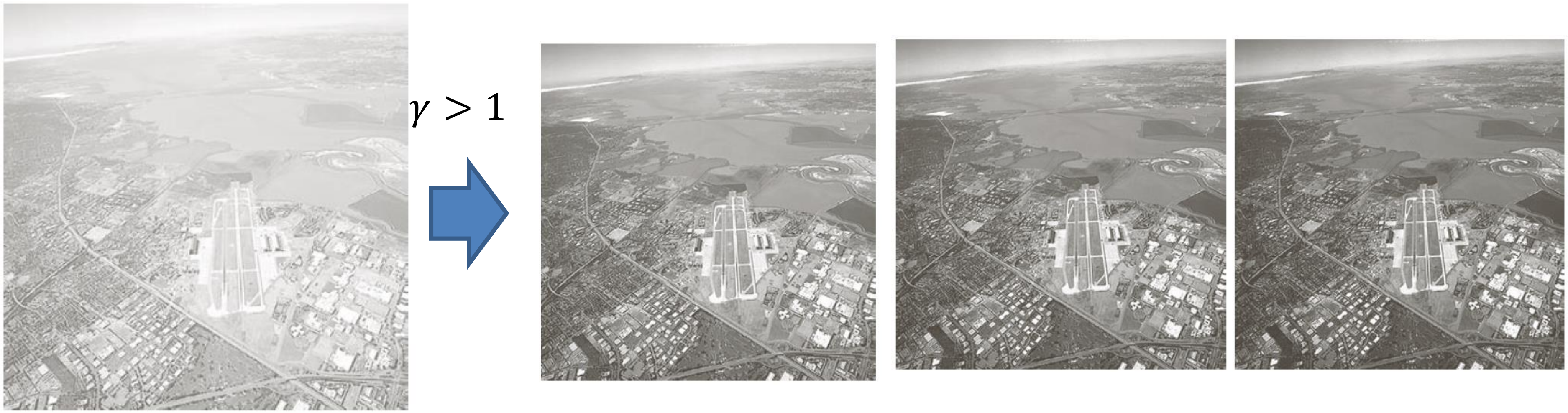


  - Depending on the value of gamma, the range of darks pixels in an image would be expanded (highlights the dark pixel details) or vice versa

Intensity transformation

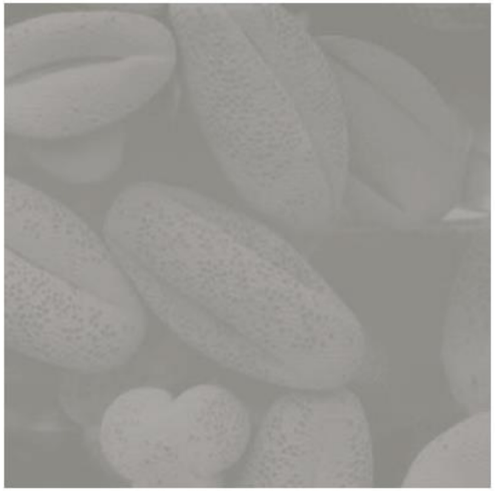
- Example

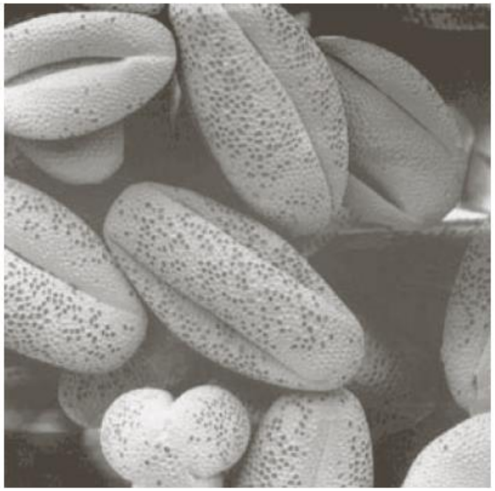


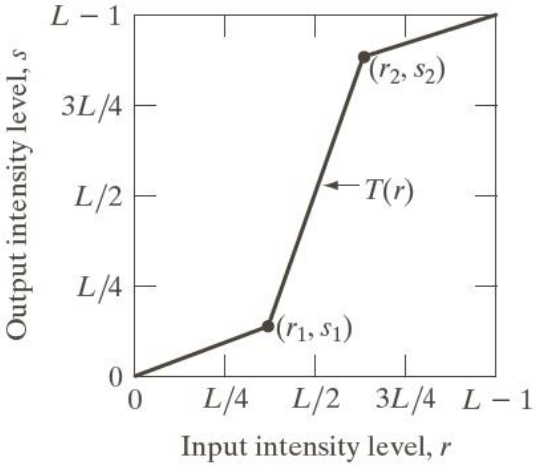


Intensity transformation

- Piecewise-linear transformation functions
  - We can form transformation function more complex

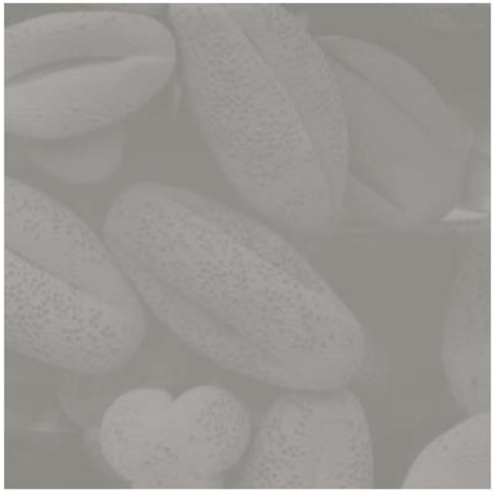


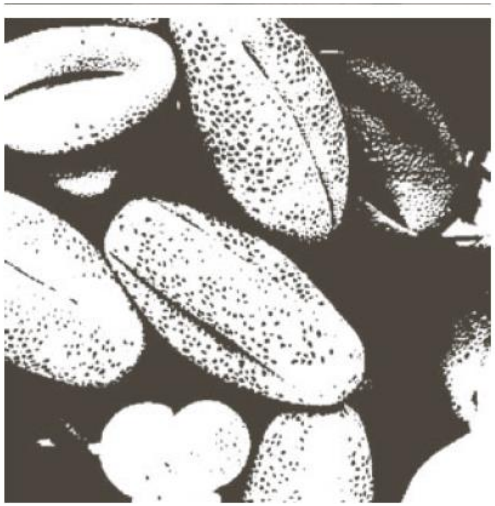


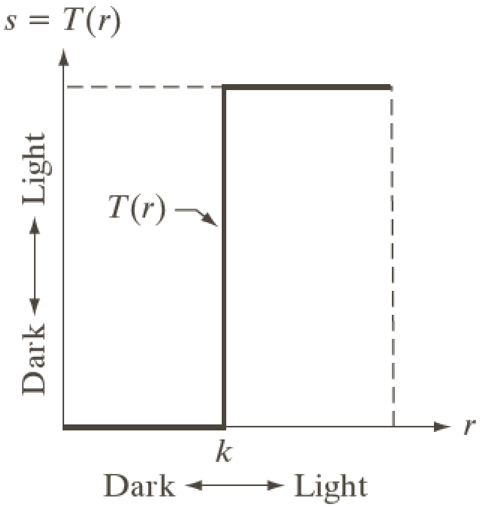


Intensity transformation

- Piecewise-linear transformation functions





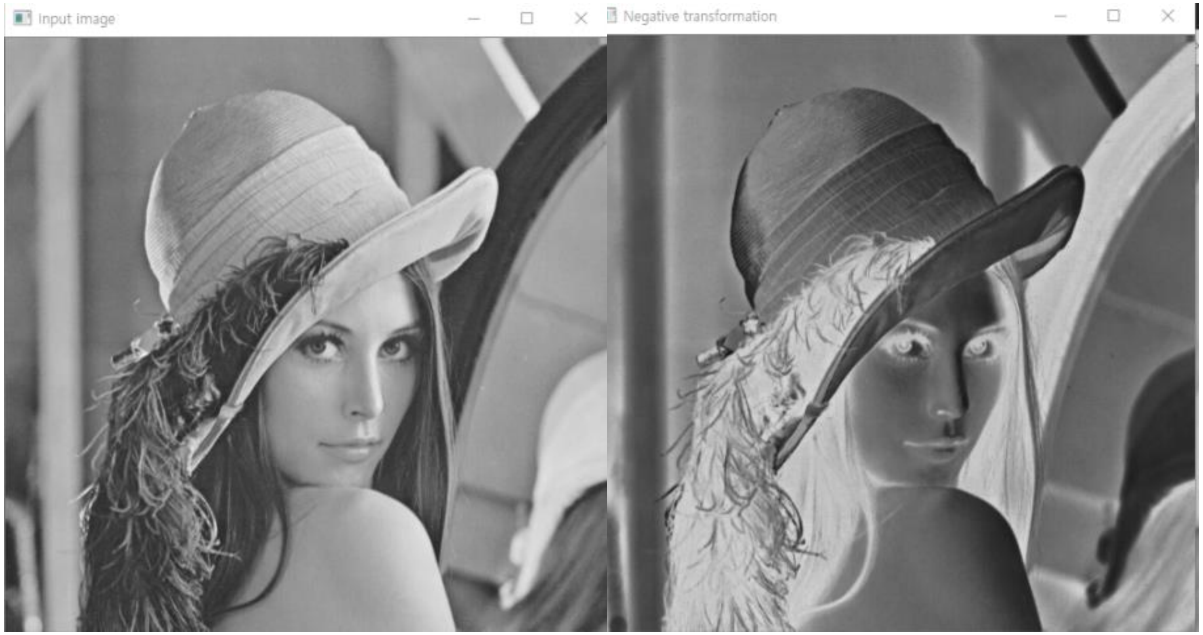


Thresholding

Intensity transformation

- Example code(Image negative)

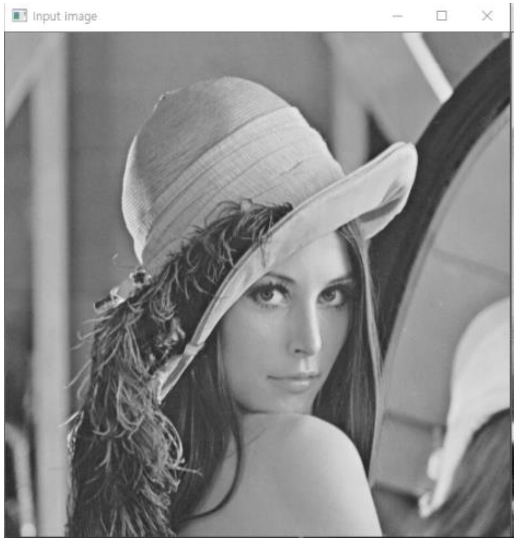
```
int main() {
        Mat image = imread("lena.png", 0);
        Mat negative_img = image.clone();
        for (int i = 0; i < image.rows; i++)
                for (int j = 0; j < image.cols; j++)
                        negative_img.at<uchar>(i, j) = 255 - image.at<uchar>(i, j);
        imshow("Input image", image);
        imshow("Negative transformation", negative_img);

        waitKey(0);
        return 0;
}
```

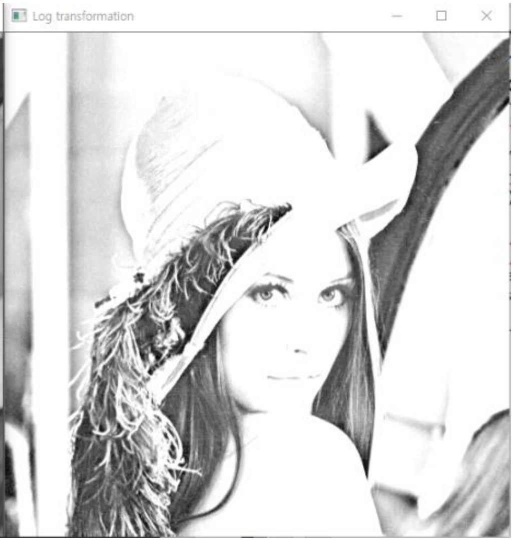


Intensity transformation

- Example code(Log transformation)

```
int main() {
        Mat image = imread("lena.png", 0);
        Mat f_img, log_img;
        double c = 1.5f; // scale constant
        image.convertTo(f_img, CV_32F);
        f_img = abs(f_img) + 1;
        log(f_img, f_img);
        normalize(f_img, f_img, 0, 255, NORM_MINMAX); // normalize image to (0~255)
        convertScaleAbs(f_img, log_img, c); // scaling by c, conversion to an unsigned 8-bit type
        imshow("Input image", image);
        imshow("Log transformation", log_img);

        waitKey(0);
}
```





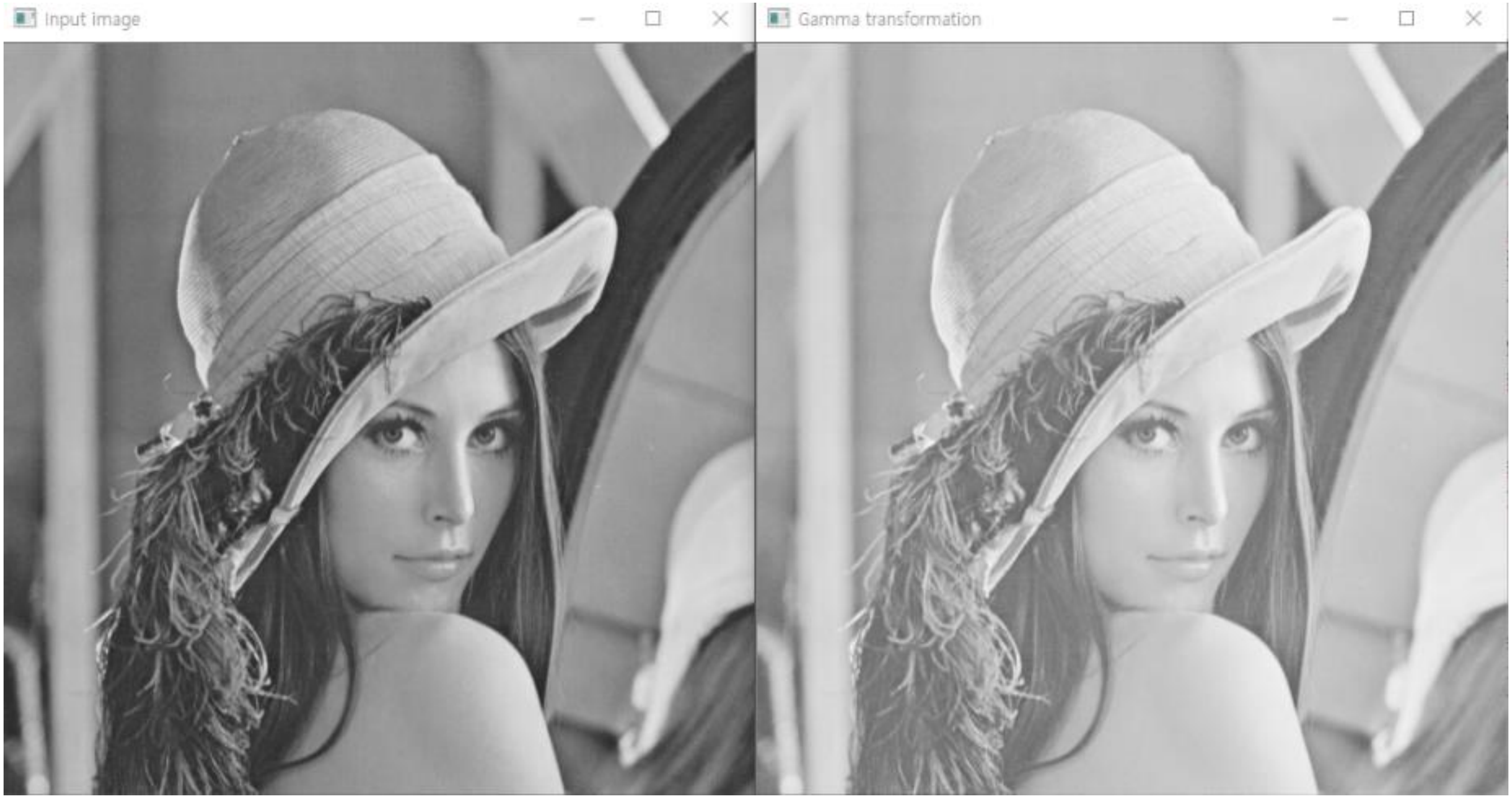
Intensity transformation

- Example code(Gamma correction)

```
int main() {
        Mat image = imread("lena.png", 0);
        Mat gamma_img;
        MatIterator_<uchar> it, end;
        float gamma = 0.5;
        unsigned char pix[256];

        for (int i = 0; i < 256; i++) {
                pix[i] = saturate_cast<uchar>(pow((float)(i / 255.0), gamma) * 255.0f);
        }
        gamma_img = image.clone();

        for (it = gamma_img.begin<uchar>(), end = gamma_img.end<uchar>(); it != end; it++) {
                *it = pix[(*it)];
        }

        imshow("Input image", image);
        imshow("Gamma transformation", gamma_img);
        waitKey(0);
}
```

Intensity transformation

- Example code(Gamma correction)


Input image
Gamma transformation