

A Note About Grey Levels

So far when we have spoken about image grey level values we have said they are in the range $[0, 255]$

- Where 0 is black and 255 is white

There is no reason why we have to use this range

- The range $[0,255]$ stems from display technologies

For many of the image processing operations in this lecture grey levels are assumed to be given in the range $[0.0, 1.0]$

Улучшение изображений

Улучшение – такая обработка изображения, при которой результат оказывается более подходящий с точки зрения *конкретного* применения

Причины улучшения:

- Выделение интересующих деталей на изображении
- Удаление шума
- Визуальная привлекательность

Пространственные и частотные методы улучшения

Два подхода к улучшению

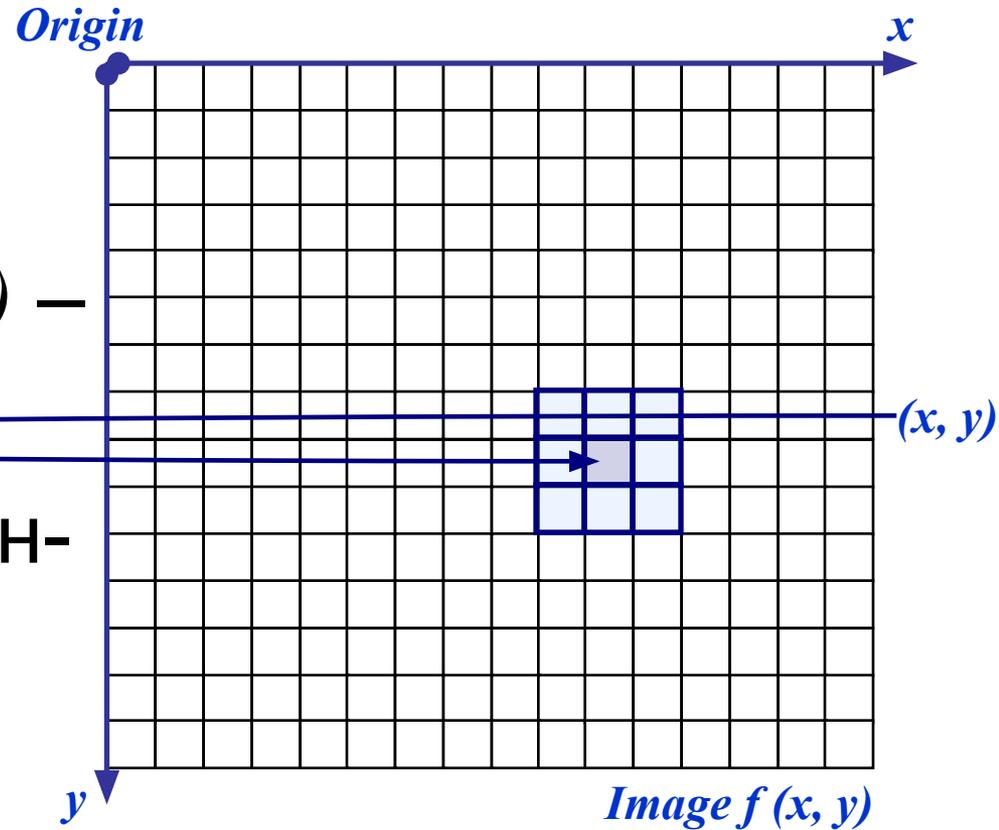
- Пространственные методы
 - Манипуляция пикселями
- Частотные методы
 - Манипуляция результатом преобразования Фурье или вейвлетного преобразования

Основы пространственных методов

Процессы пространственной обработки описываются уравнением

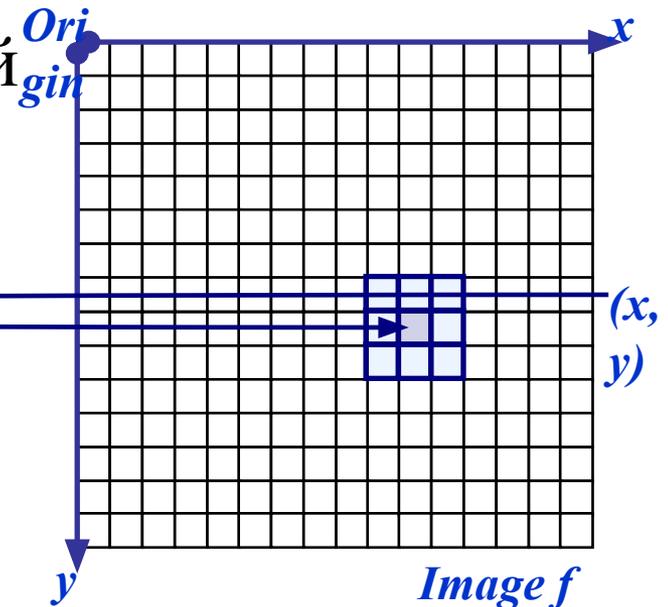
$$g(x, y) = T[f(x, y)]$$

где $f(x, y)$ входное изображение, $g(x, y)$ – обработанное, T – оператор, определенный в некоторой окрестности (x, y)



Классификация операторов

- **поэлементные (пиксельные)**
 - 1x1, не требуют данные от соседних пикселей
 - вычисления можно произвести за константное время при достаточном кол-ве процессоров
- **локальные**
 - требуют данные в некоторой окрестности (x, y)
- **глобальные**



1x1

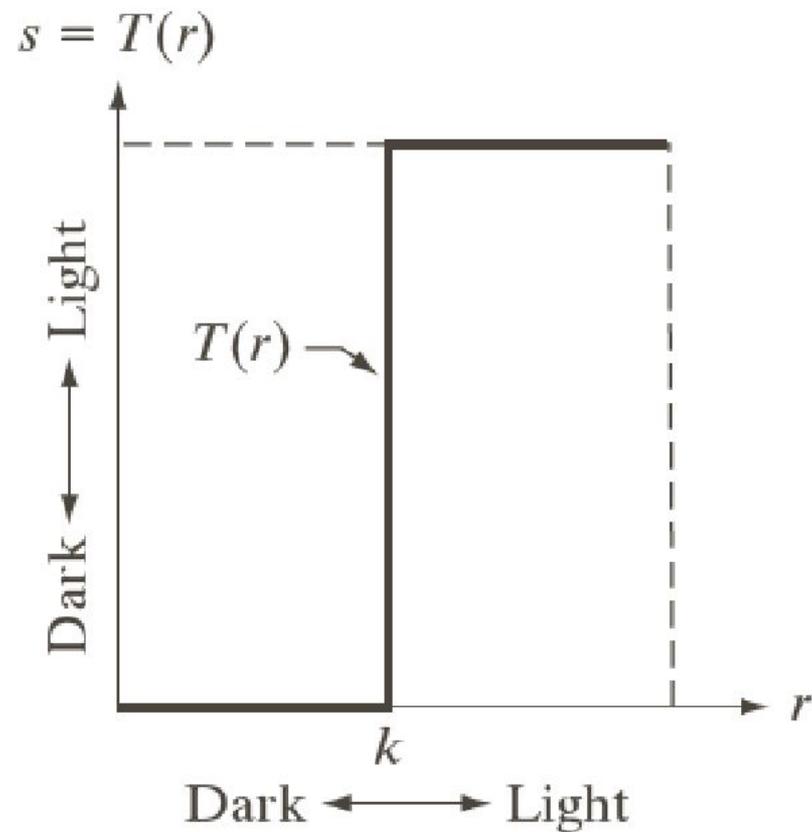
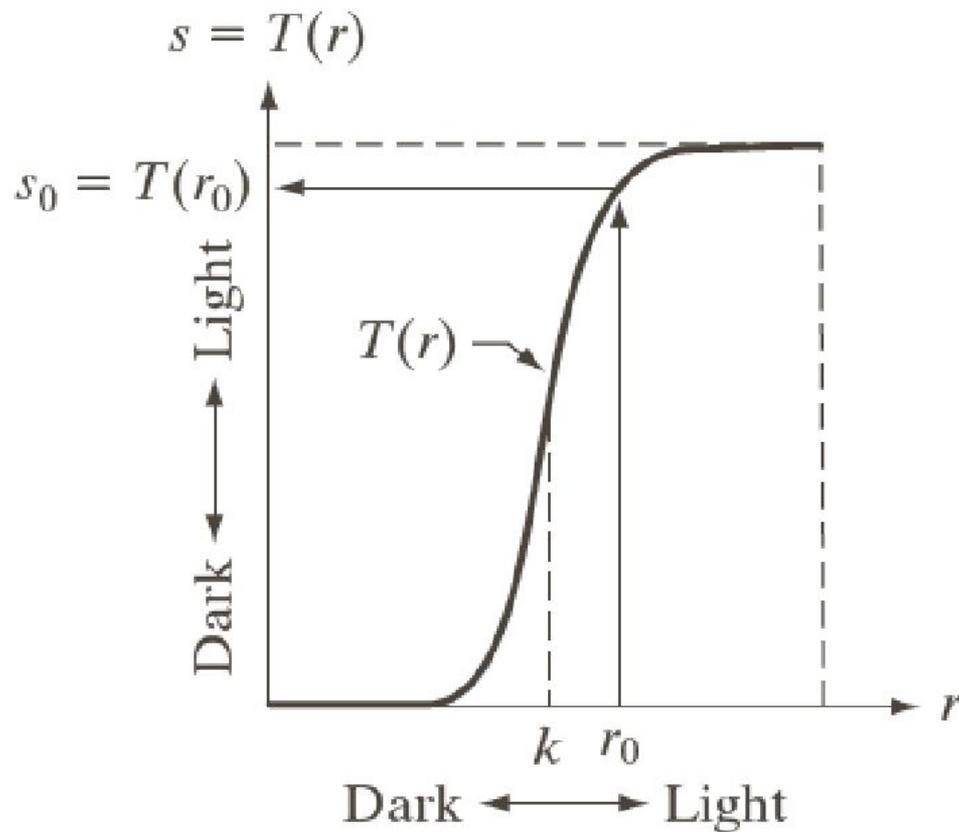
T называется *градационным преобразованием (grey level transformation function* или *a point processing operation)*

вида

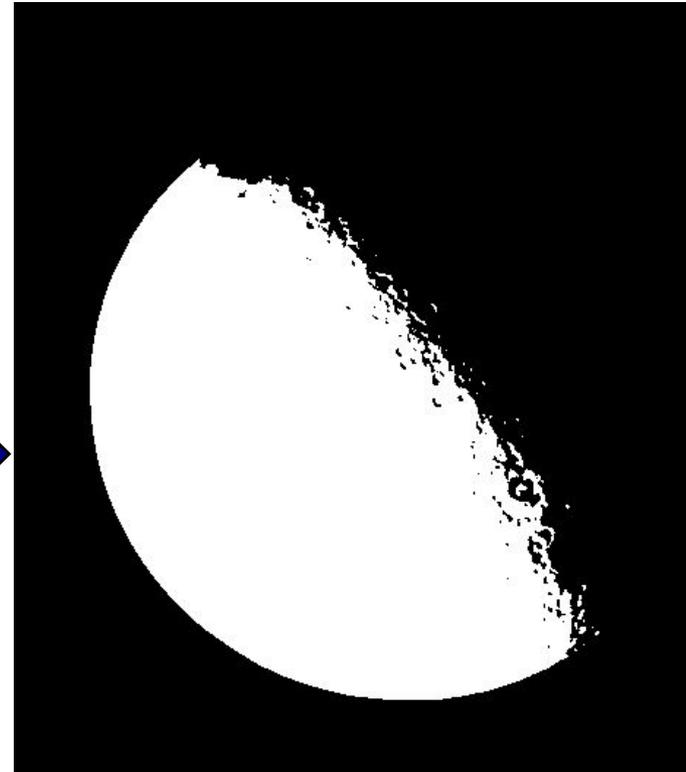
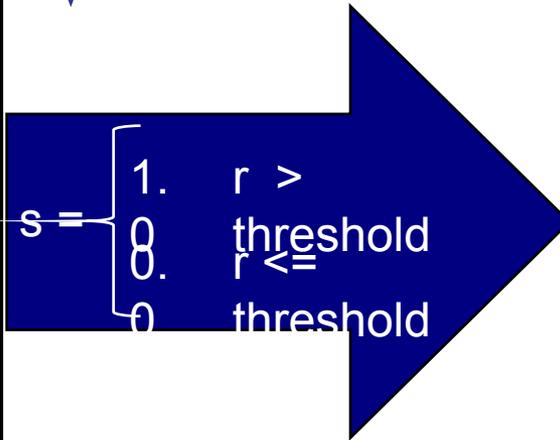
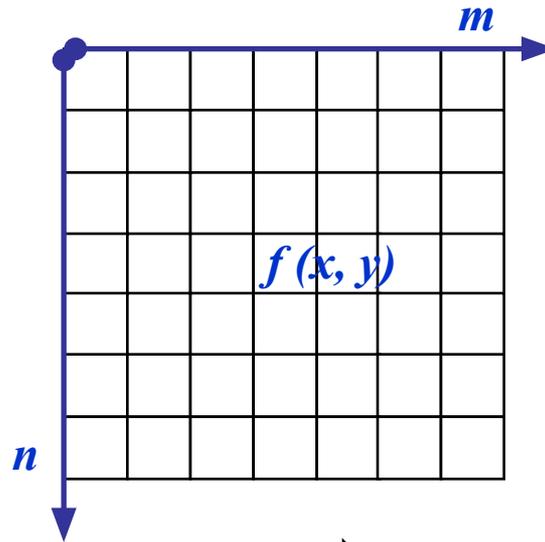
$$s = T (r)$$

где *s* интенсивность обработанного изображения, *r* – исходного

Градационное преобразование



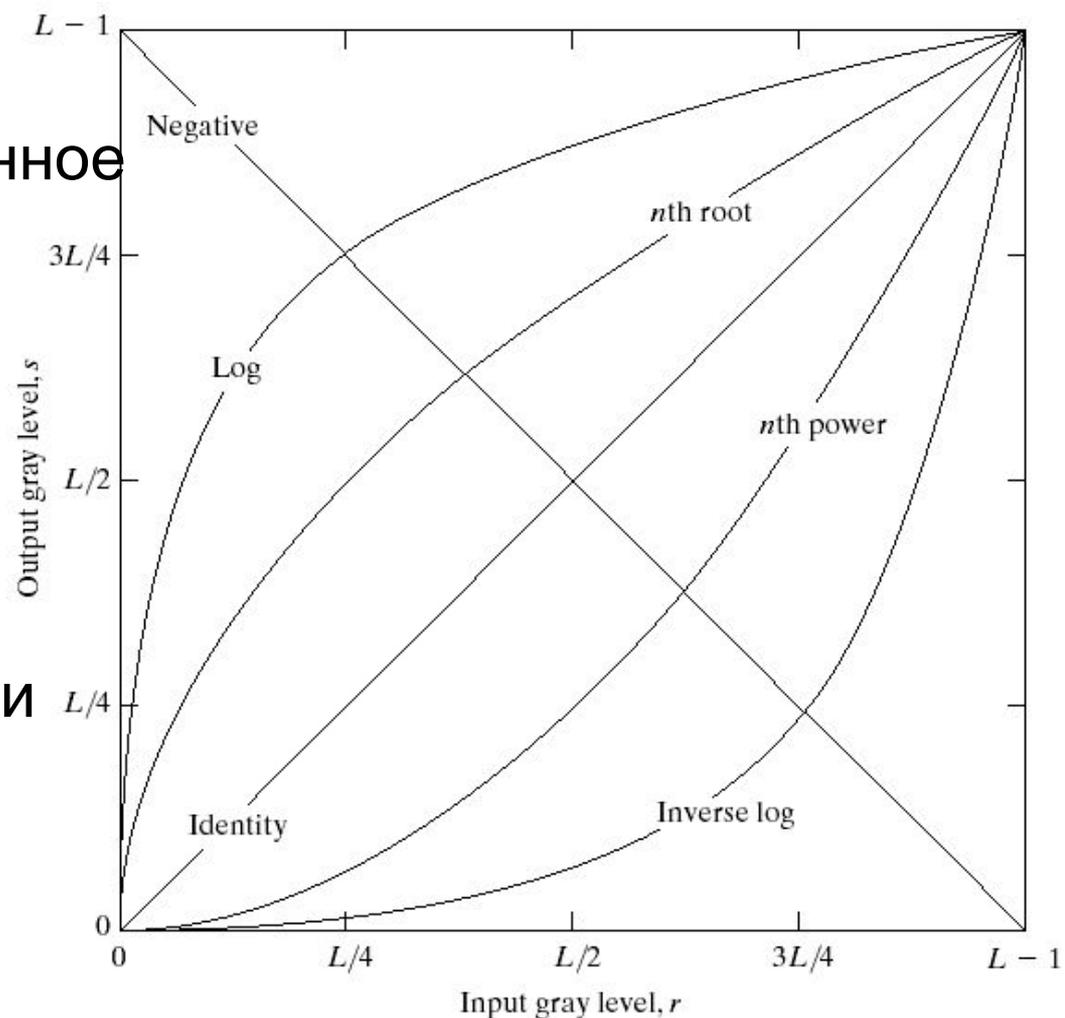
Пороговая поэлементная обработка



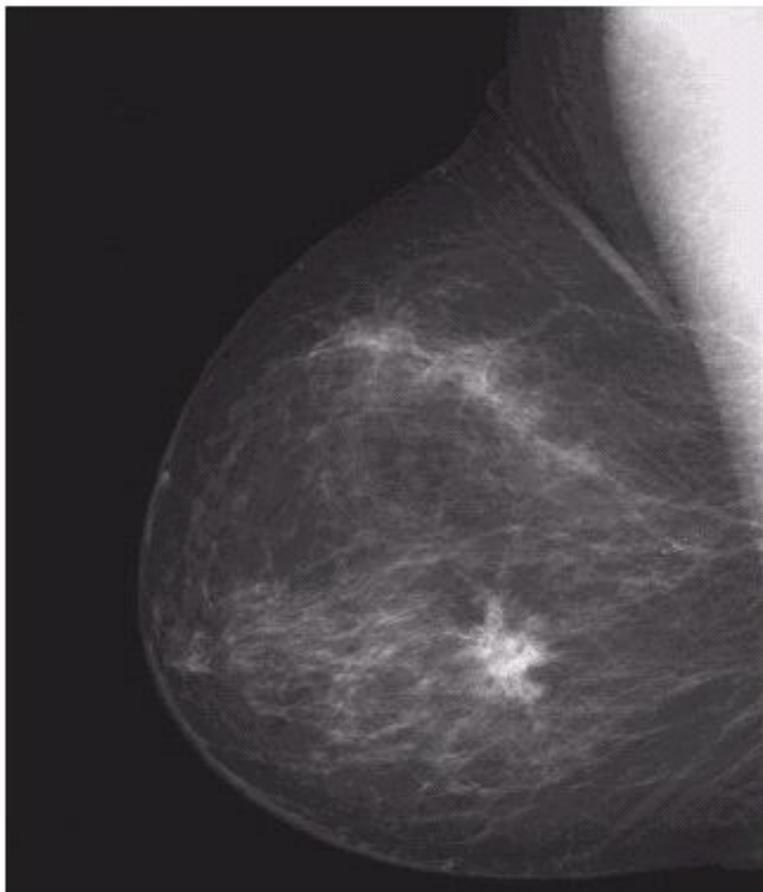
Основные градационные преобразования

3 основных типа:

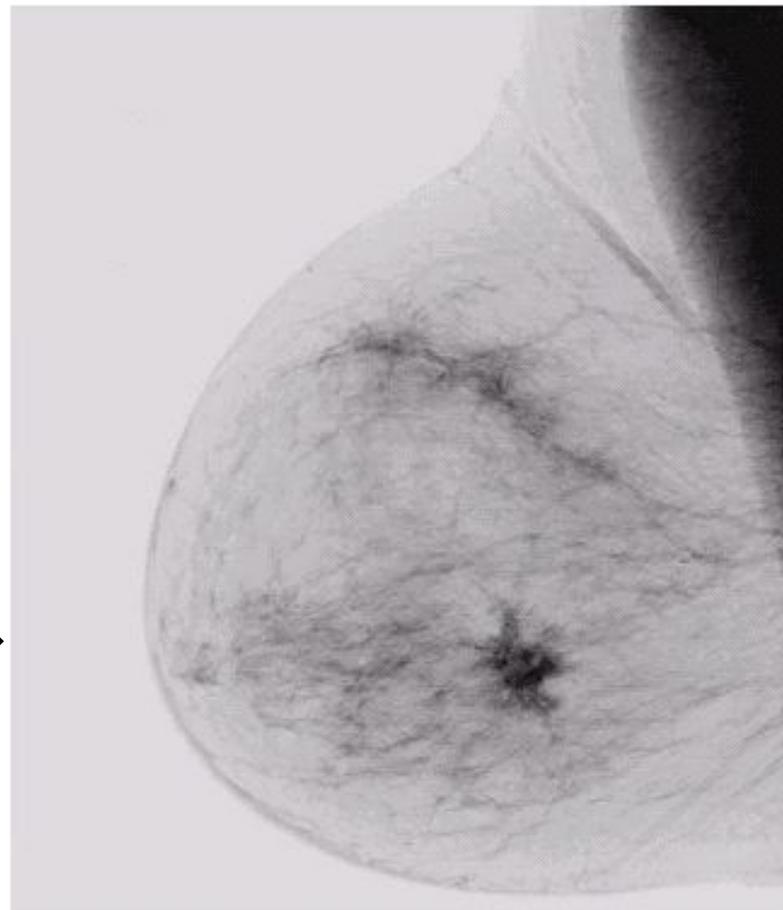
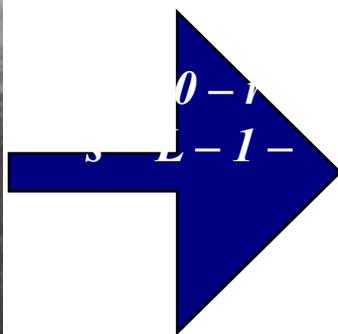
- линейное
 - негатив/тождественное преобразование
- логарифмическое
 - log/обратный log
- степенное
 - n -ая степень/
корень n -ой степени



$$s = max_intensity - r$$



оригинал

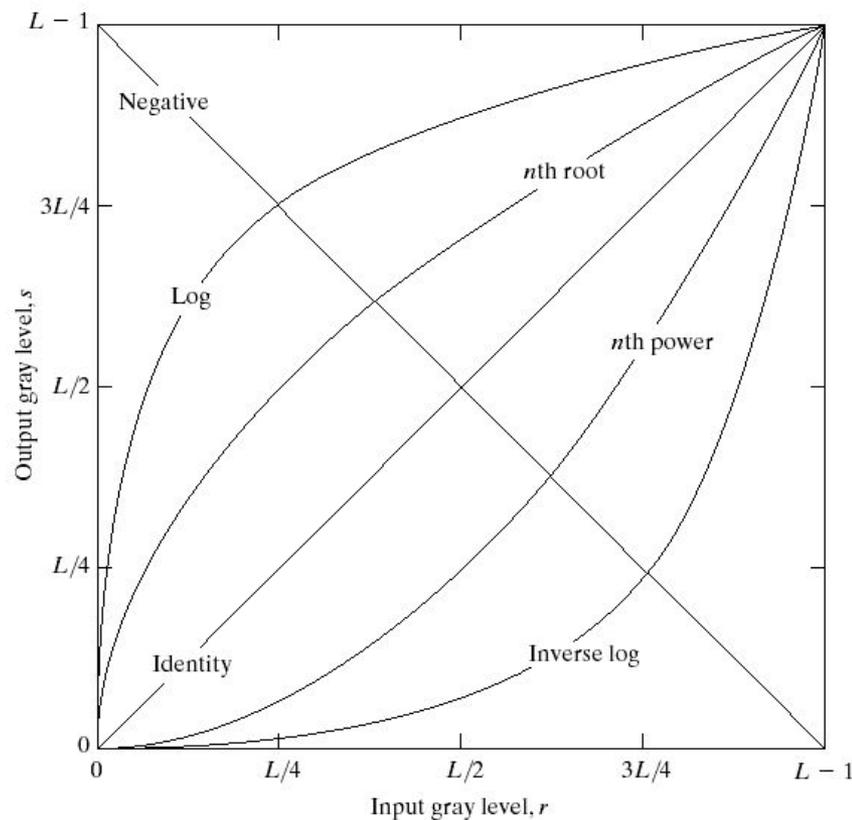


негатив

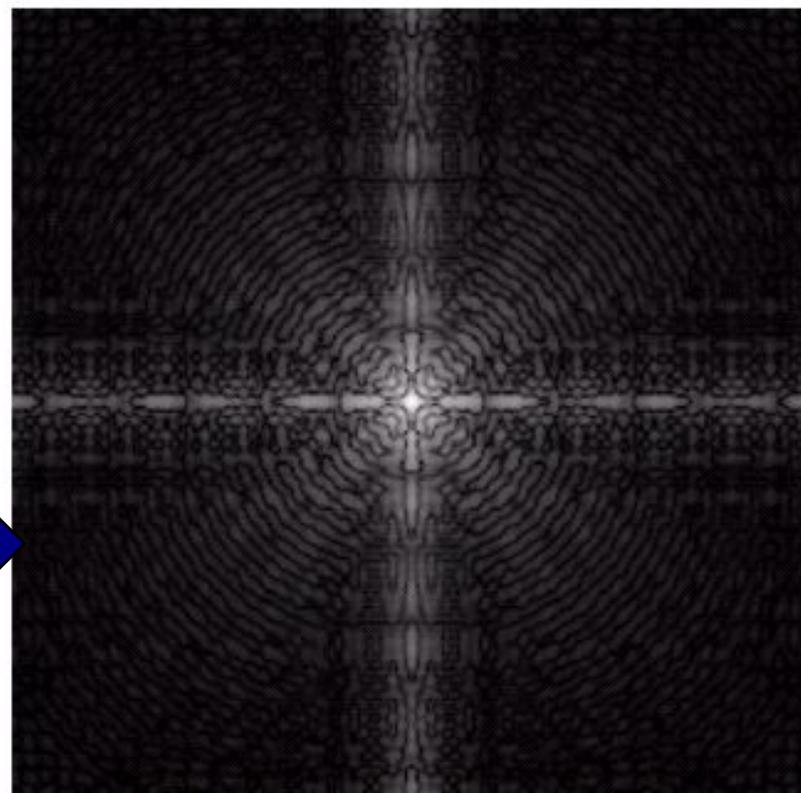
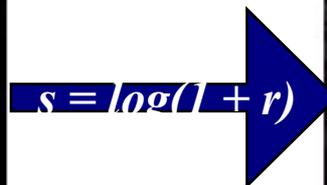
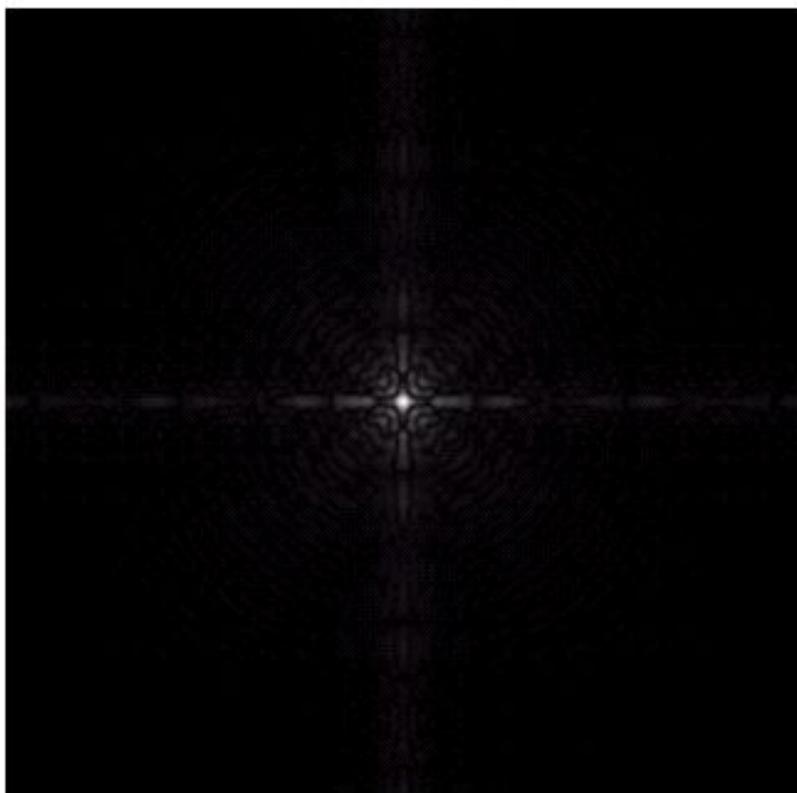
Общий вид

$$s = c * \log(1 + r),$$

где c – константа, $r \geq 0$.



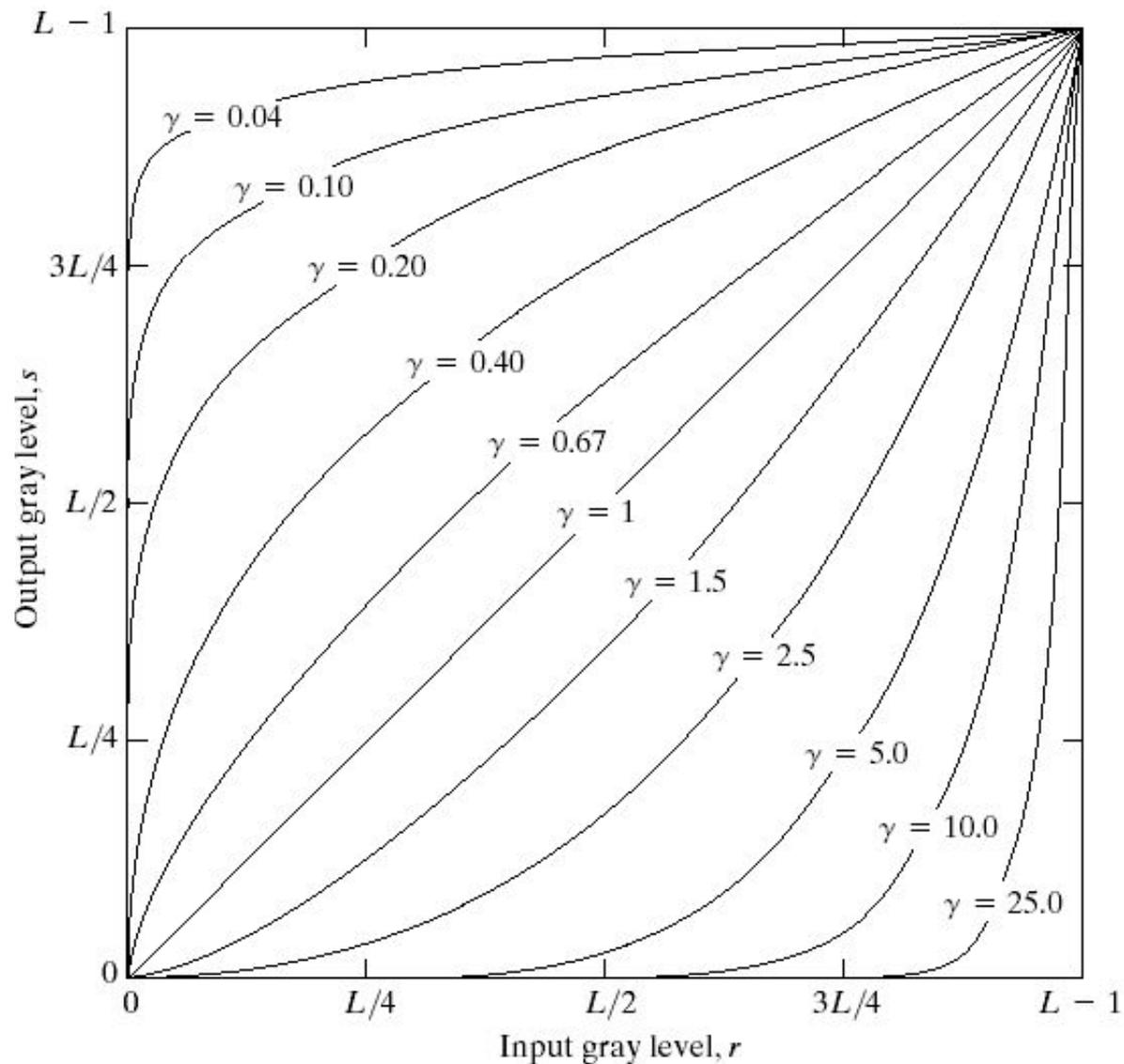
Логарифмическое преобразование



Степенные преобразования

Общий вид

$$s = c * r^\gamma$$

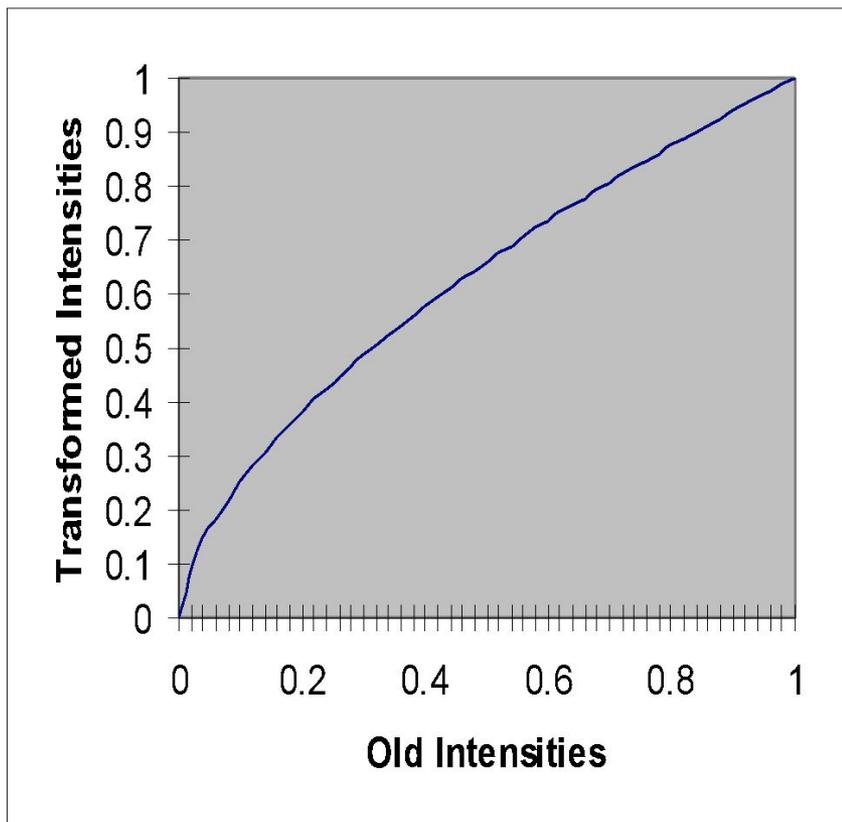


Степенные преобразования



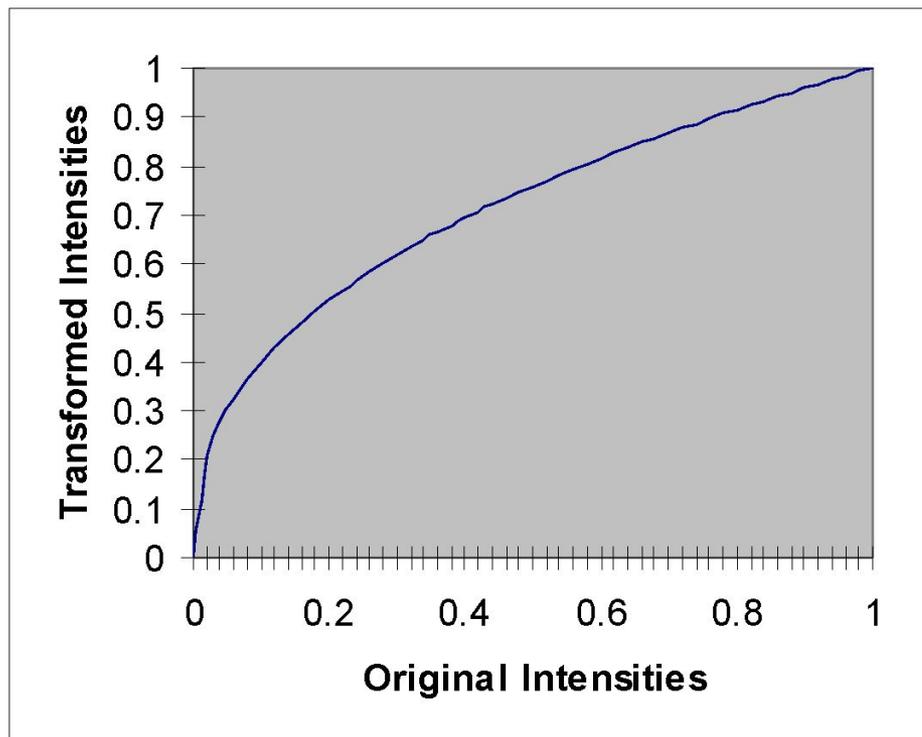
Степенные преобразования

$$\gamma = 0.6$$



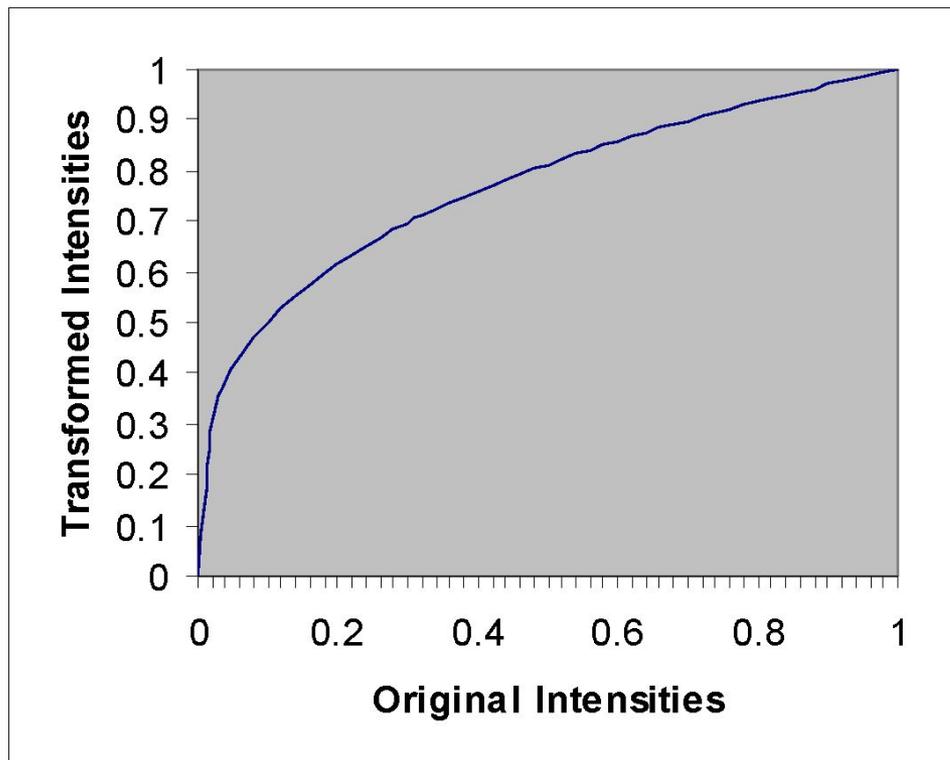
Степенные преобразования

$$\gamma = 0.4$$



Степенные преобразования

$$\gamma = 0.3$$



Степенные преобразования



Степенные преобразования



$\gamma = 3.0$



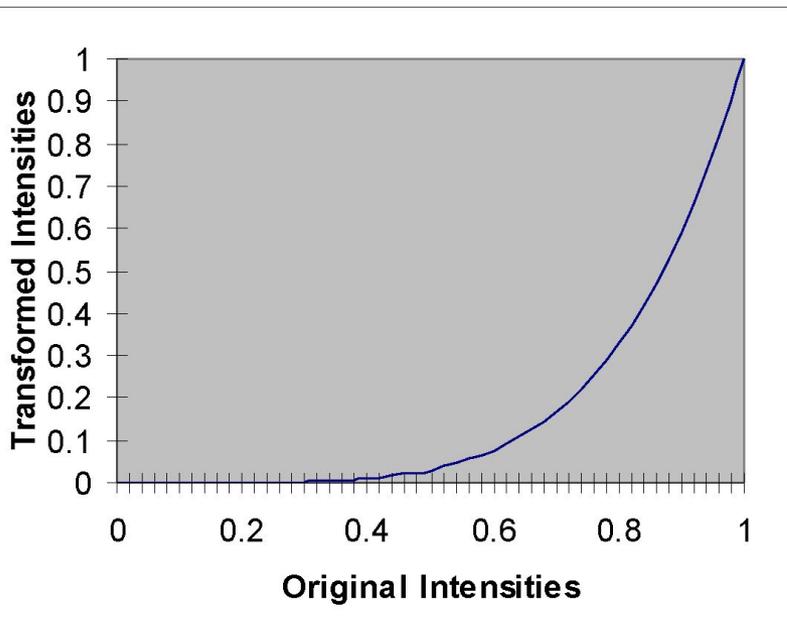
$\gamma = 4.0$



$\gamma = 5.0$

Степенные преобразования

$$\gamma = 5.0$$

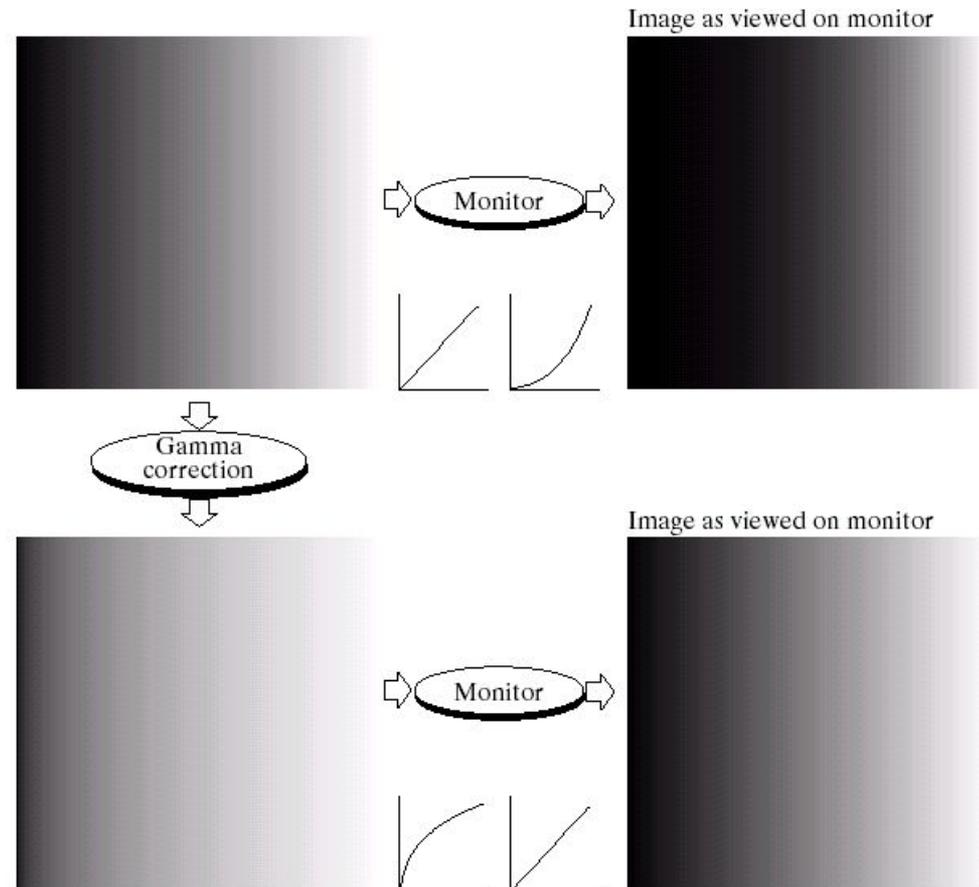


Gamma Correction

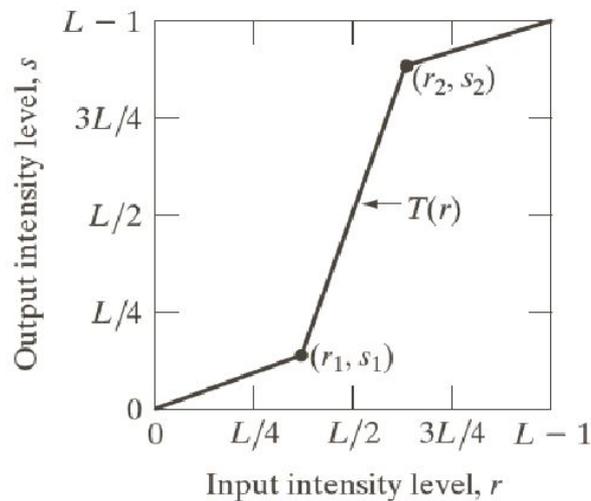
Many of you might be familiar with gamma correction of computer monitors

Problem is that display devices do not respond linearly to different intensities

Can be corrected using a log transform

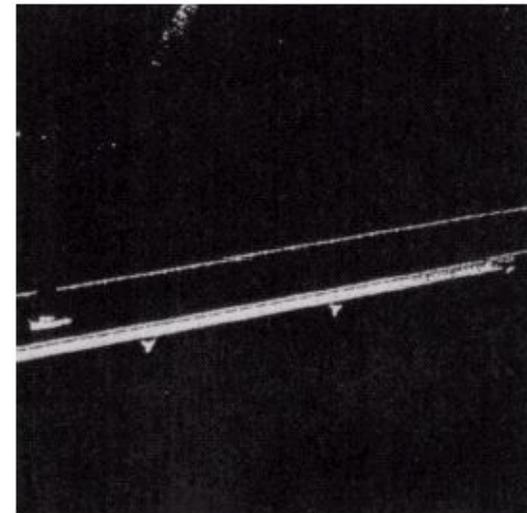
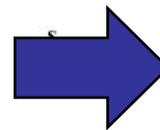
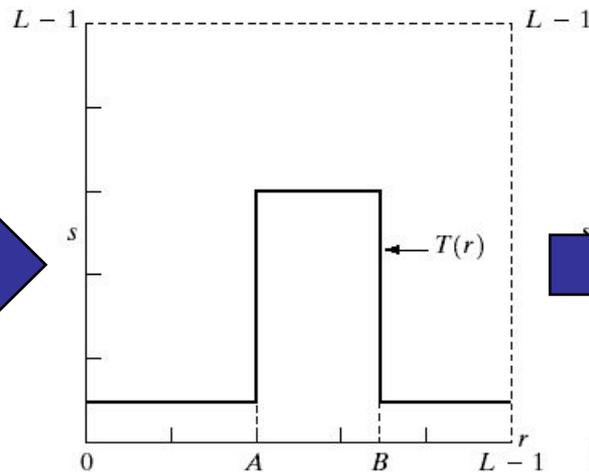
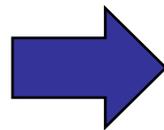
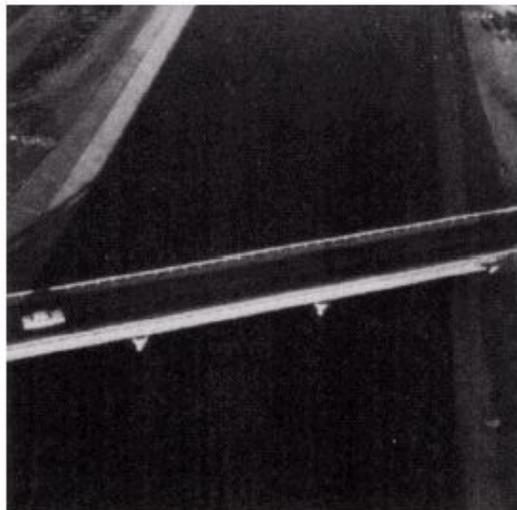
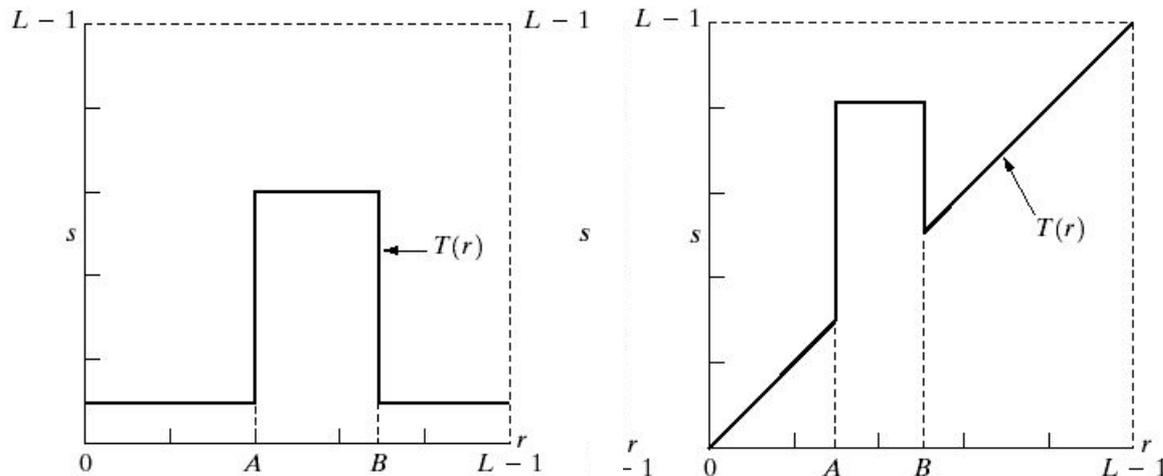


Кусочно-линейные функции преобразований

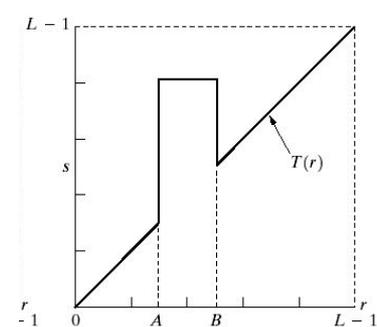
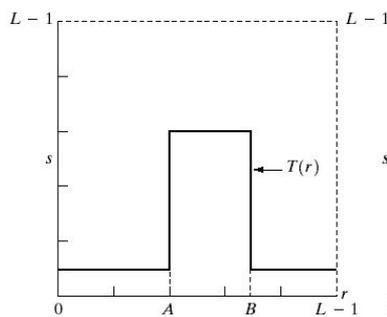


Вырезание диапазона яркостей

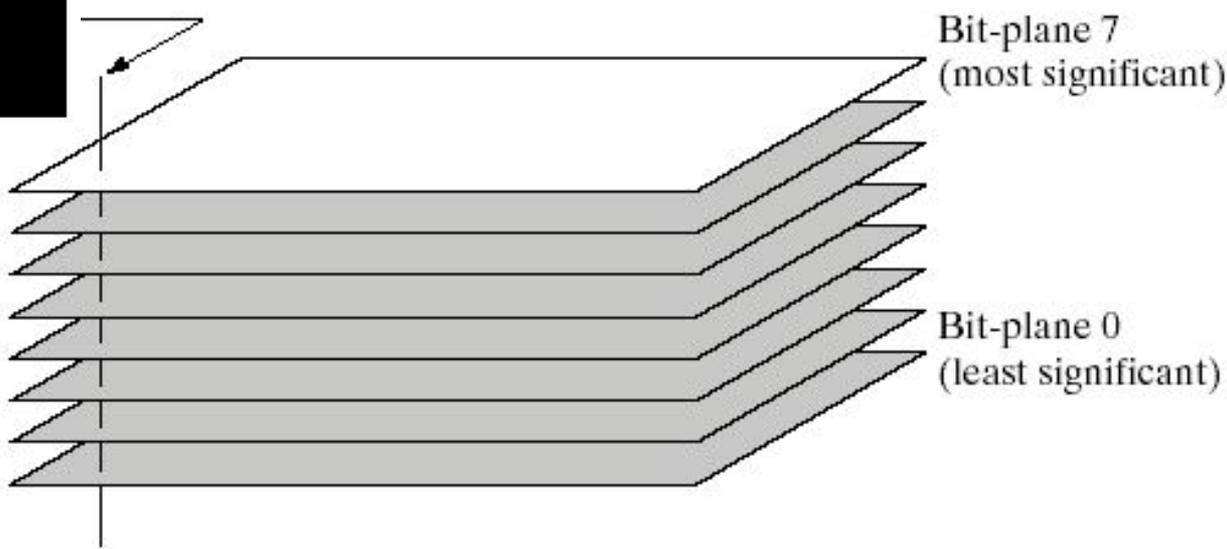
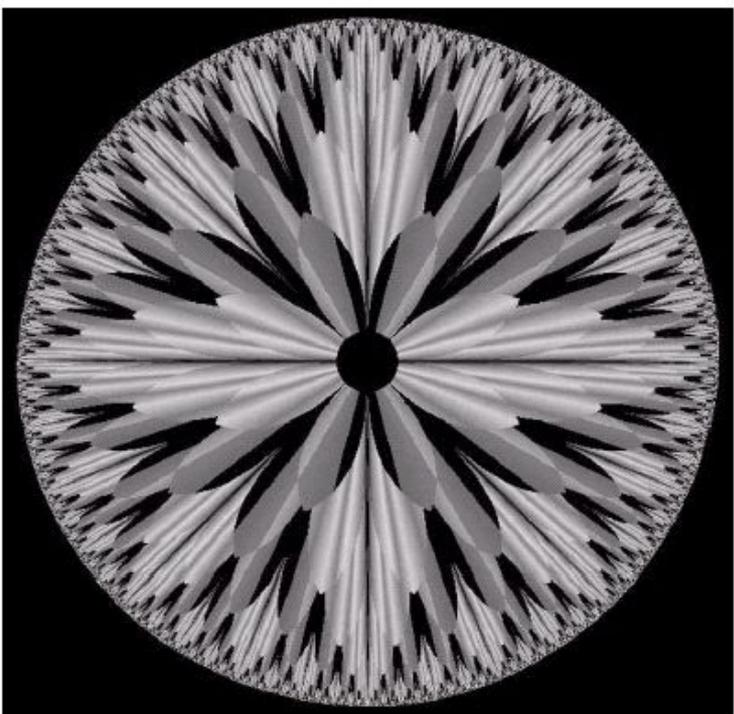
Выделение некоторого диапазона яркостей



Вырезание диапазона яркостей

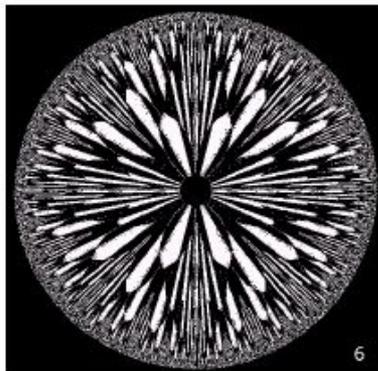
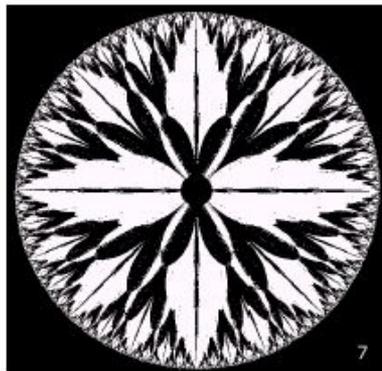


Вырезание битовых плоскостей



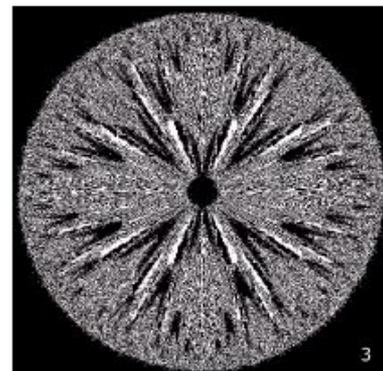
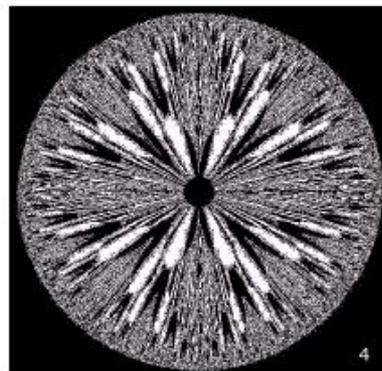
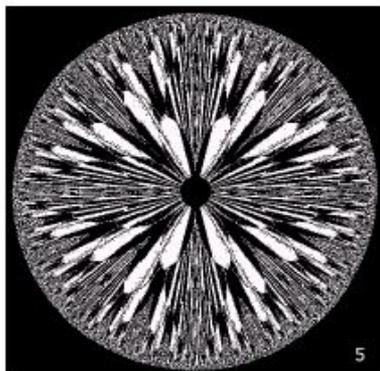
Вырезание битовых плоскостей

[10000000]



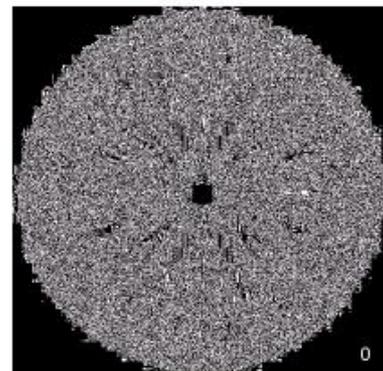
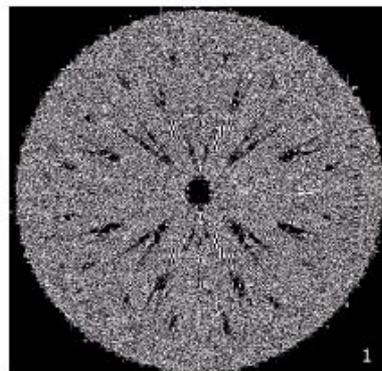
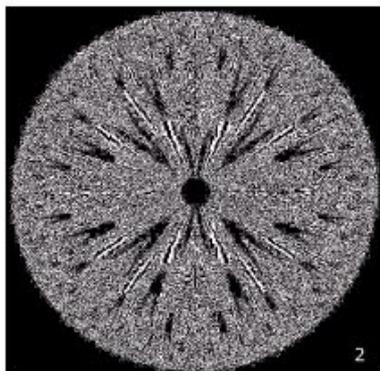
[01000000]

[00100000]



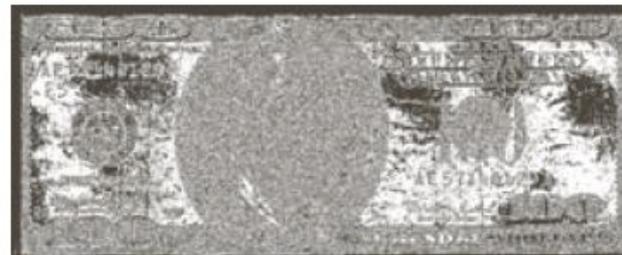
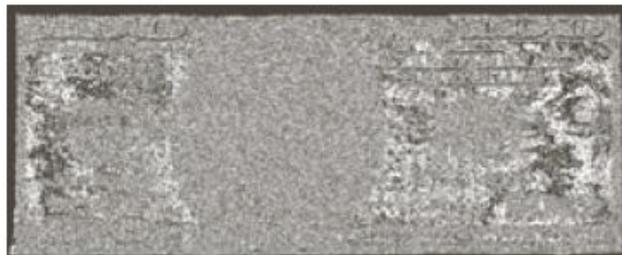
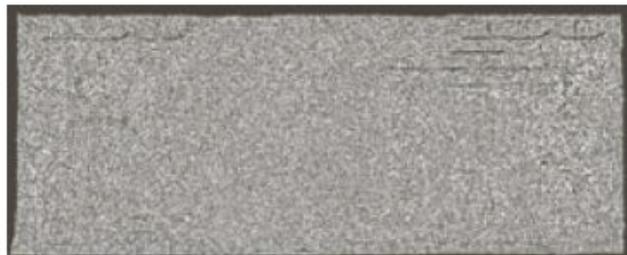
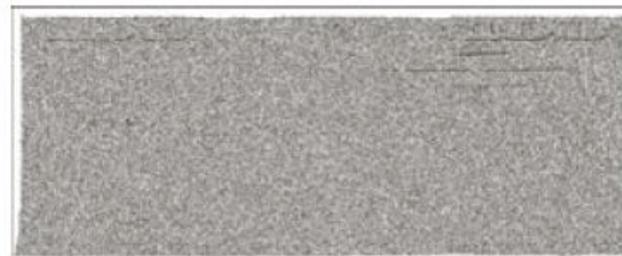
[00001000]

[00000100]



[00000001]

Вырезание битовых плоскостей



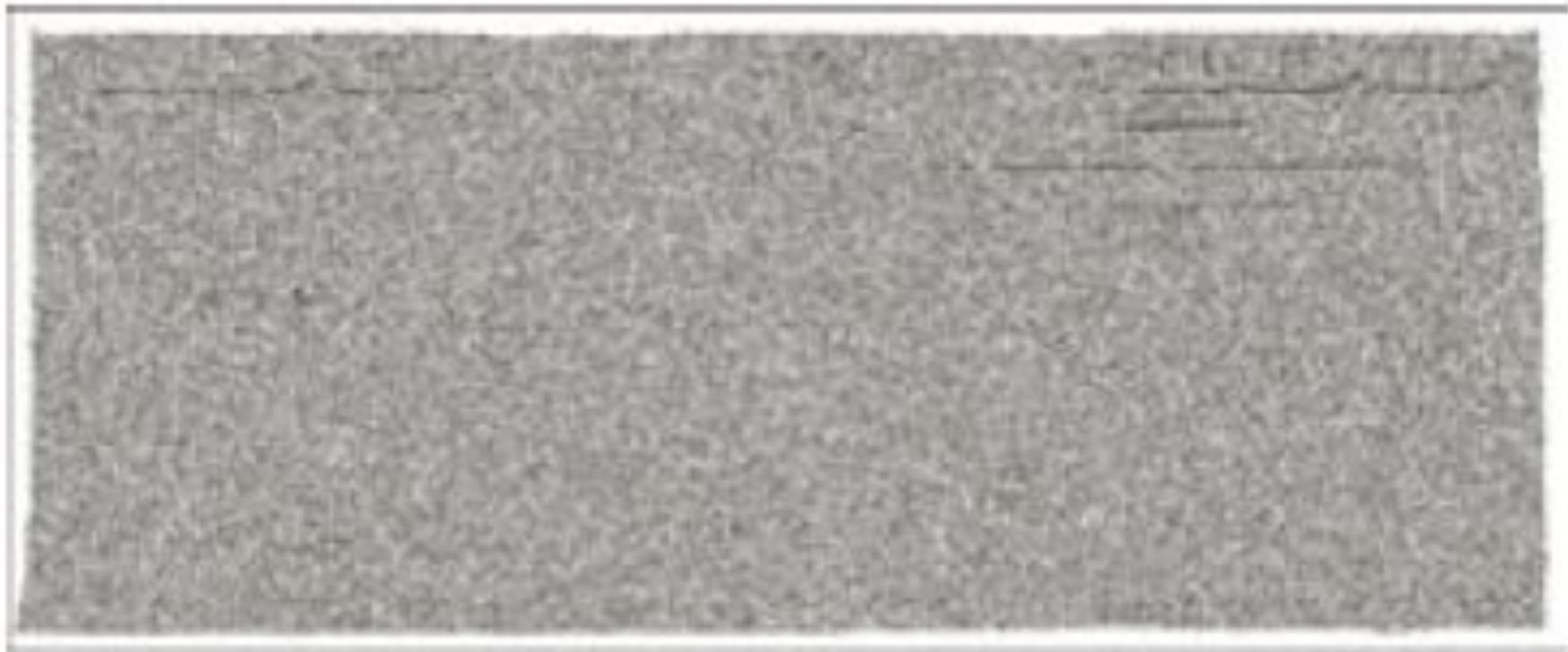
Вырезание битовых плоскостей



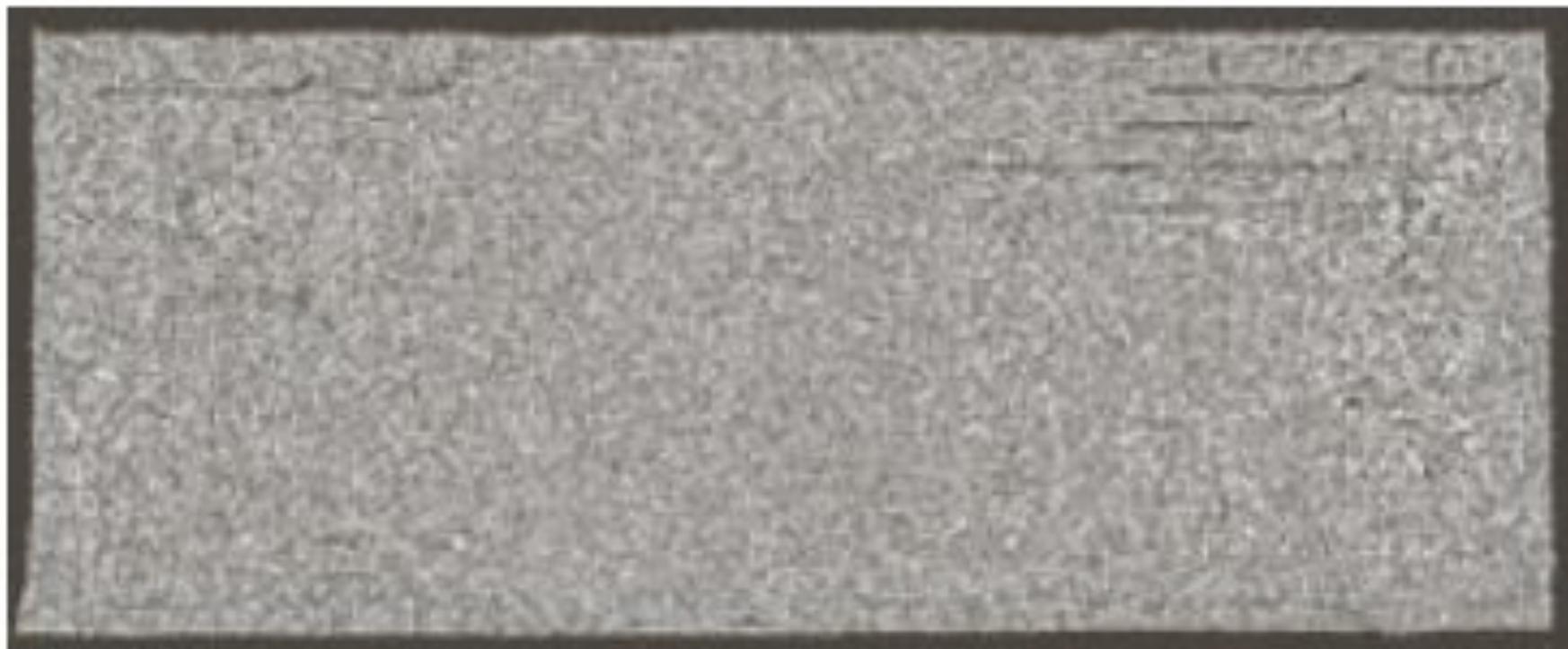
Вырезание битовых плоскостей



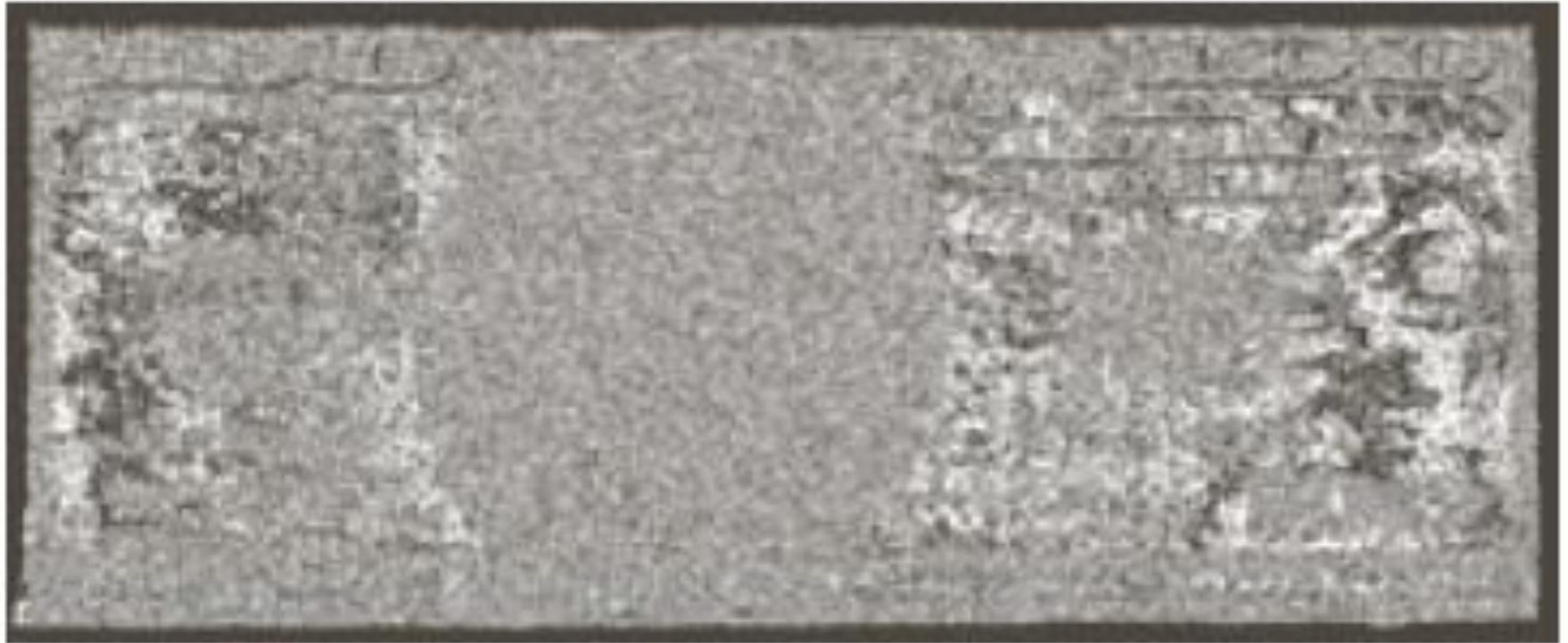
Вырезание битовых плоскостей



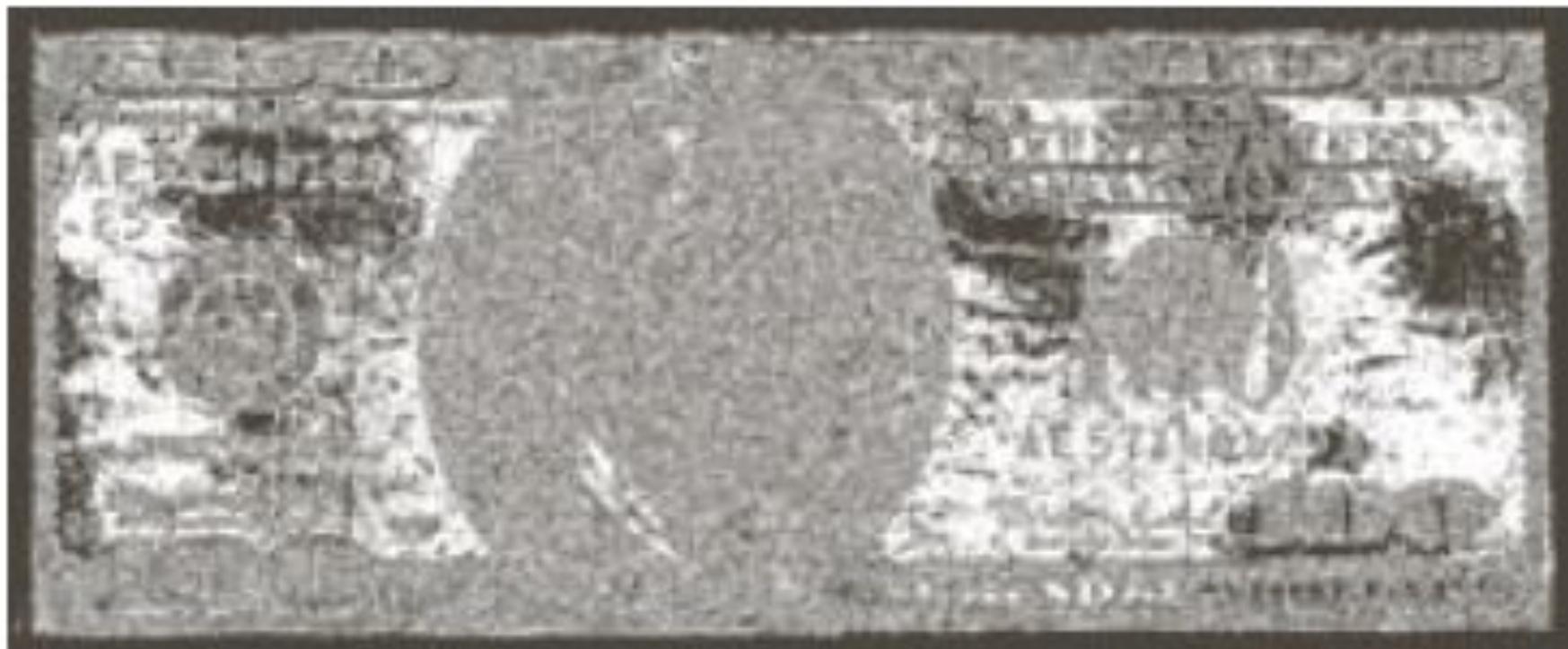
Вырезание битовых плоскостей



Вырезание битовых плоскостей



Вырезание битовых плоскостей



Вырезание битовых плоскостей



Вырезание битовых плоскостей



Вырезание битовых плоскостей



Вырезание битовых плоскостей



Reconstructed image using only bit planes 8 and 7



Reconstructed image using only bit planes 8, 7 and 6



Reconstructed image using only bit planes 7, 6 and 5