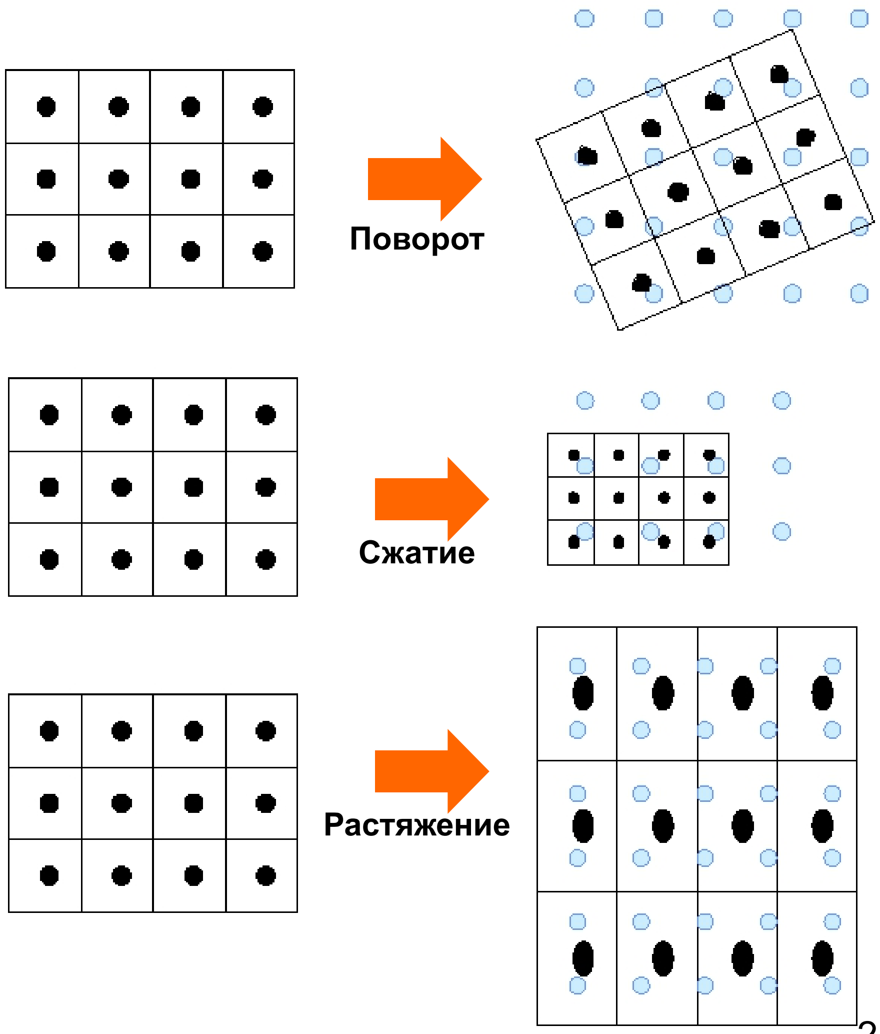
Геометрические преобразования изображений

Геометрические преобразования

Геометрические преобразования изображения включают в себя два этапа:

1. Определение нового местоположения пикселов
2. Определение значений пикселов на новом изображении



Виды геометрических преобразований

- Линейные
  - Проективные
  - Аффинные
  - Евклидовы
- Нелинейные
  - Полиномиальные
  - Кусочно-линейные
  - Сплайн-преобразование
  - …

Линейные геометрические преобразования

$$\mathbf{x} = \mathbf{T}\mathbf{x}$$

Проективные:

$\mathbf{T}$ – невырожденная матрица произвольного вида

Аффинные, евклидовы:

$\mathbf{T}$ – невырожденная матрица специального вида

Однородные координаты

Однородными координатами называют тройки $(\bar{x}, \bar{y}, w)$ чисел (одновременно не равные нулю), связанные с обычными координатами точек плоскости соотношением:

$$\begin{matrix} \bar{x} \\ \bar{y} \\ w \end{matrix} = w \begin{matrix} x \\ y \\ 1 \end{matrix} \qquad x = \bar{x}/w,\ y = \bar{y}/w$$

.

Однородные координаты

Евклидовы преобразования (преобразования движения), традиционно описываемые в матричной форме так:

$$\begin{matrix} x \\ y \end{matrix} = \mathbf{R} \begin{matrix} x \\ y \end{matrix} + \begin{matrix} t_x \\ t_y \end{matrix} \qquad \mathbf{R} = \begin{matrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{matrix} \qquad \mathbf{t} = \begin{matrix} t_x \\ t_y \end{matrix}$$

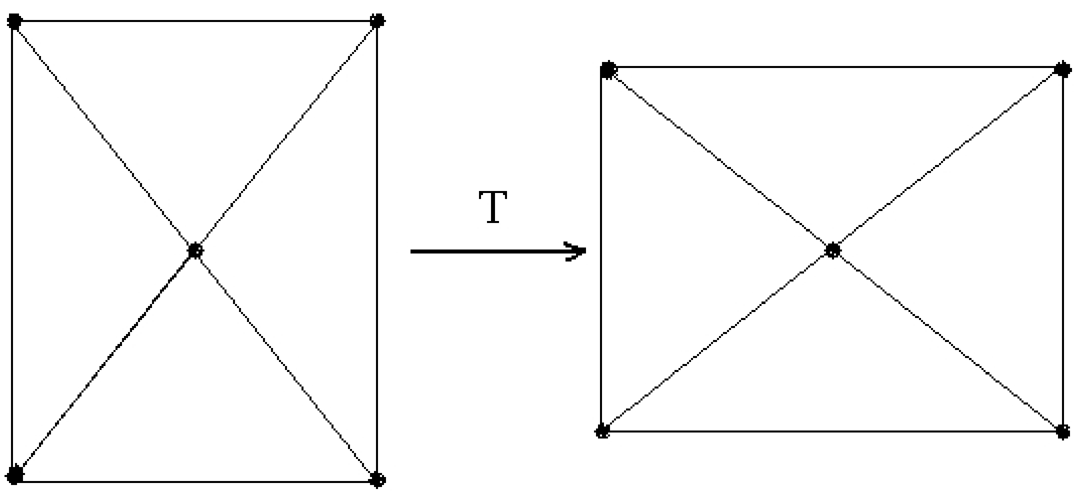
где $\mathbf{R}$ -- матрица поворота на угол φ, $\mathbf{t}$ -- вектор сдвига, в однородных координатах записываются более компактно:

$$\begin{matrix} \boldsymbol{x} \\ 1 \end{matrix} = \begin{matrix} \boldsymbol{R} & \boldsymbol{t} \\ \boldsymbol{0}^t & 1 \end{matrix} \begin{matrix} \boldsymbol{x} \\ 1 \end{matrix}, \quad \boldsymbol{0} = \begin{matrix} 0 \\ 0 \end{matrix}.$$

Евклидовы преобразования

$$\begin{pmatrix} x \\ y \end{pmatrix} = \mathbf{R} \begin{pmatrix} x \\ y \end{pmatrix} + \begin{pmatrix} t_x \\ t_y \end{pmatrix} \qquad \mathbf{R} = \begin{pmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{pmatrix}$$

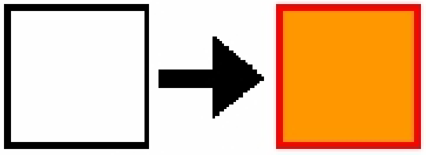
$$\begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0}^{\mathbf{t}} & 1 \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ 1 \end{pmatrix}, \mathbf{0} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$



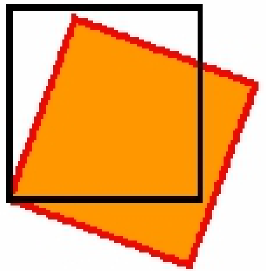
Евклидовы преобразования

Любое евклидово преобразование может быть сконструировано из двух базовых преобразований:

- сдвиг;
- поворот.



Сдвиг



Поворот

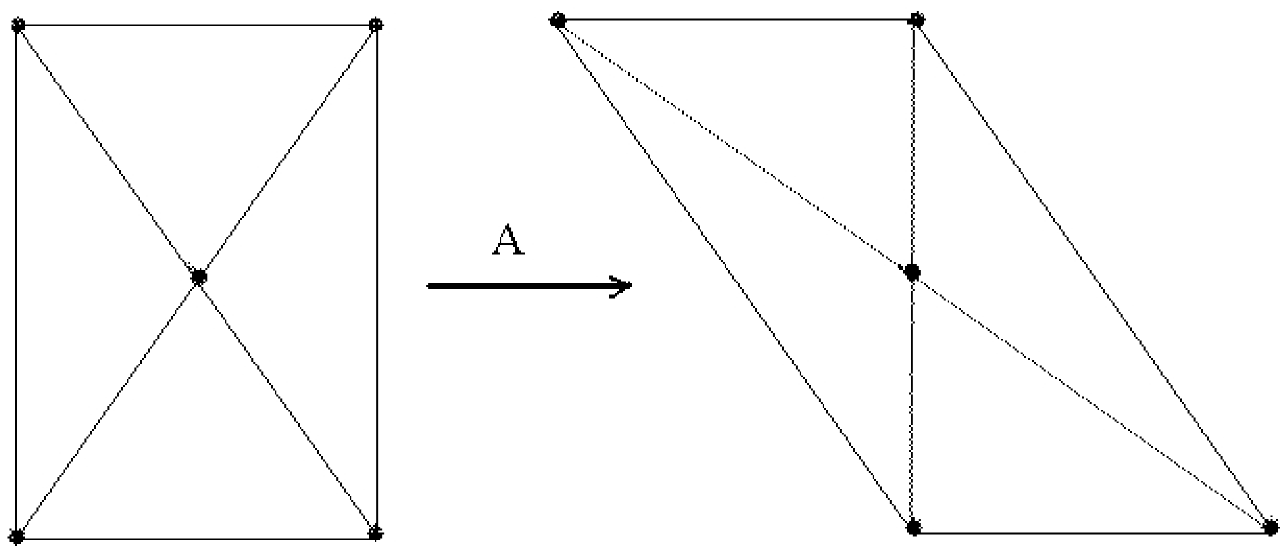
Евклидовы преобразования

.

Аффинные преобразования

$$x \ = ax + by + c, \qquad \boldsymbol{x} \ = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{c}, \ \boldsymbol{x} \ = \begin{matrix} x \\ y \end{matrix}, \ \boldsymbol{A} = \begin{matrix} a & b \\ d & e \end{matrix}, \ \boldsymbol{c} = \begin{matrix} c \\ f \end{matrix}$$
$$y \ = dx + ey + f,$$

$$\begin{matrix} \boldsymbol{x} \\ 1 \end{matrix} = \begin{matrix} \boldsymbol{A} & \boldsymbol{c} \\ \boldsymbol{0}^t & 1 \end{matrix} \begin{matrix} \boldsymbol{x} \\ 1 \end{matrix}, \ \boldsymbol{0} = \begin{matrix} 0 \\ 0 \end{matrix}.$$



Аффинные преобразования

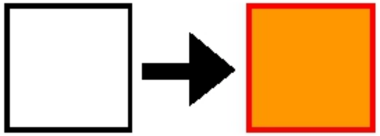
Любое аффинное преобразование может быть сконструировано из нескольких базовых преобразований:

– сдвиг;

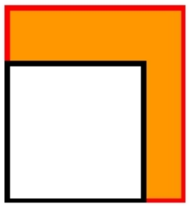
– масштабирование;

– поворот;

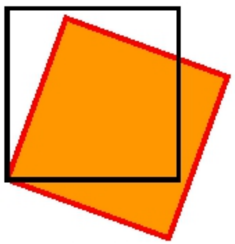
– скос.



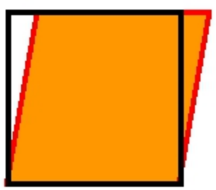
Сдвиг



Масштабирование



Поворот



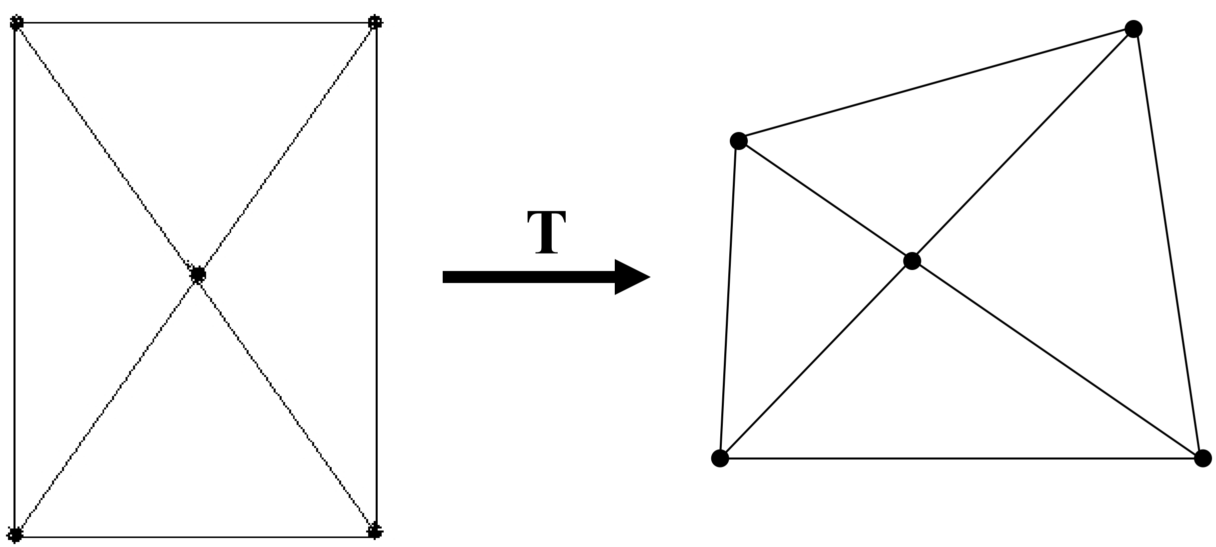
Скос

Проективные преобразования

$$\begin{pmatrix} \bar{x}' \\ \bar{y}' \\ w' \end{pmatrix} = \begin{pmatrix} w'x' \\ w'y' \\ w' \end{pmatrix} = \mathbf{T} \begin{pmatrix} x \\ y \\ w \end{pmatrix}.$$

$$x = \frac{t_{11}x + t_{12}y + t_{13}}{t_{31}x + t_{32}y + t_{33}},$$

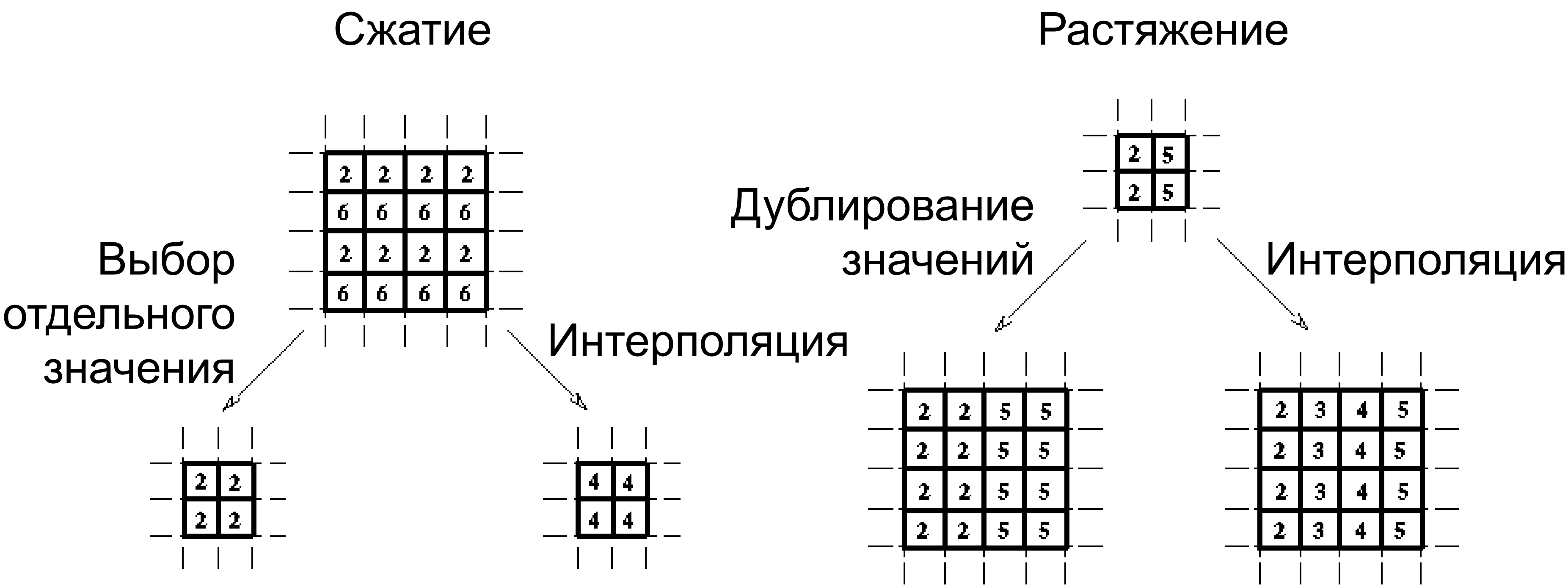
$$y = \frac{t_{21}x + t_{22}y + t_{23}}{t_{31}x + t_{32}y + t_{33}},$$



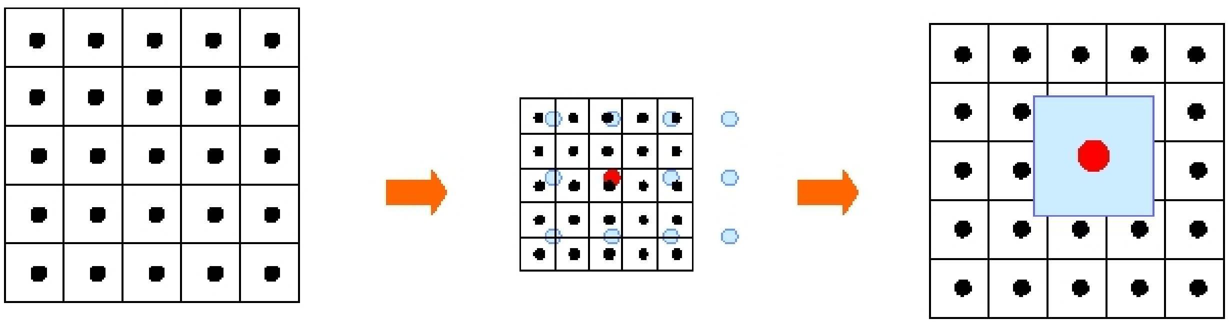
Перспективные преобразования

.

Реализация масштабирования: простые способы



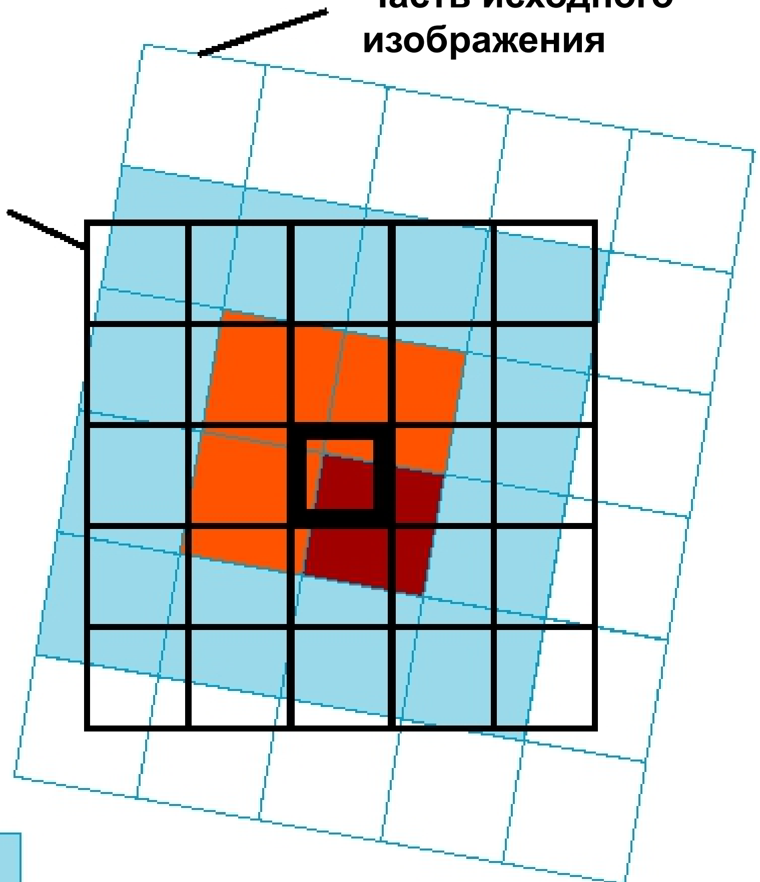
Реализация масштабирования: более общий подход



Методы интерполяции значений пикселов

Часть исходного изображения

Часть преобразованного изображения. Жирной линией изображен пиксел, значение которого подлежит определению путем интерполяции значений соседних пикселов

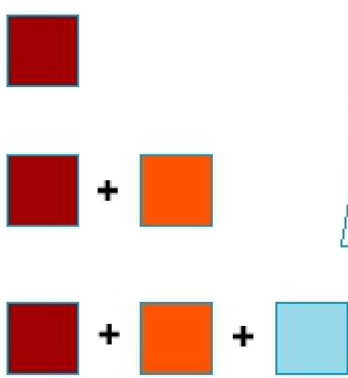


Соседние пикселы, задействуемые различными методами интерполяции

Интерполяция по ближайшему соседу

Билинейная интерполяция

Бикубическая интерполяция



Интерполяция по ближайшему соседу

Интенсивность пиксела на результирующем изображении отождествляется с интенсивностью пиксела наименее удаленного от его прообраза на исходном изображении.

Билинейная интерполяция

- обобщение линейной интерполяции одной переменной для функций двух переменных.

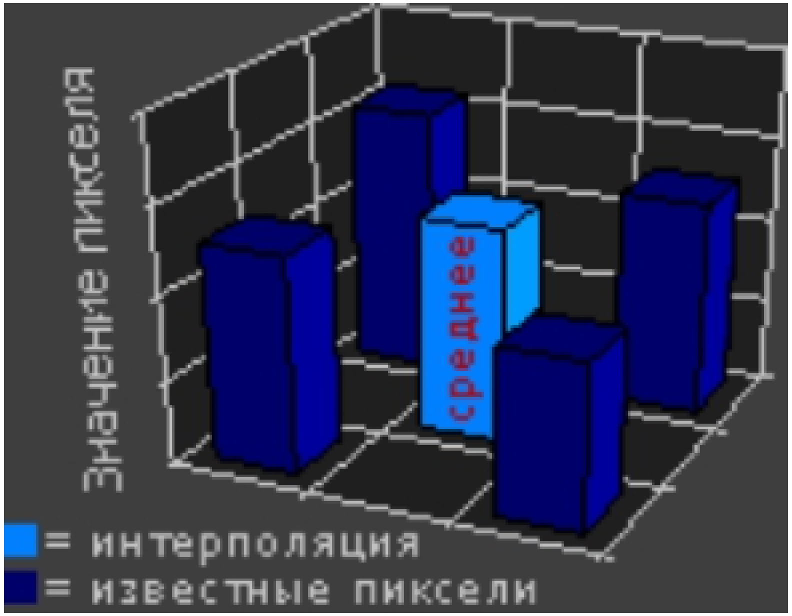
Обобщение основано на применении обычной линейной интерполяции сначала в направлении одной из координат, а затем в перпендикулярном направлении.

Билинейная интерполяция

Билинейная интерполяция рассматривает квадрат 2x2 известных пикселя, окружающих неизвестный. В качестве интерполированного значения используется взвешенное усреднение этих четырёх пикселей. В результате изображения выглядят значительно более гладко, чем результат работы метода ближайшего соседа.

Билинейная интерполяция

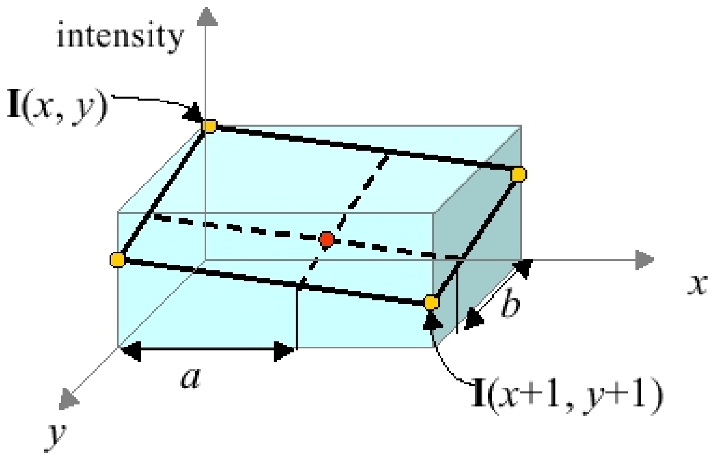
Диаграмма относится к случаю, когда все известные пиксели равны, так что интерполированное значение просто является их суммой, поделенной на 4.

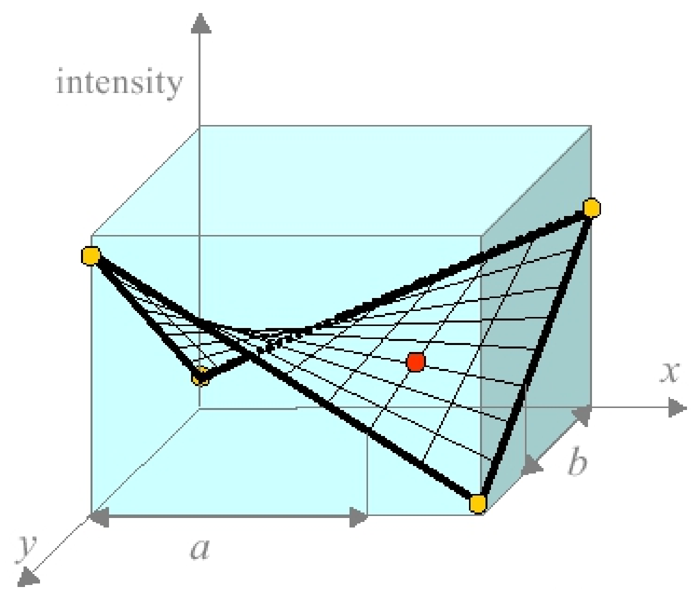


Билинейная интерполяция

Интерполяция значения **I(x’,y’)** (показано красным цветом) по значениям **I(x,y), I(x+1,y), I(x,y +1), I(x+1,y +1)** (показаны желтым цветом)

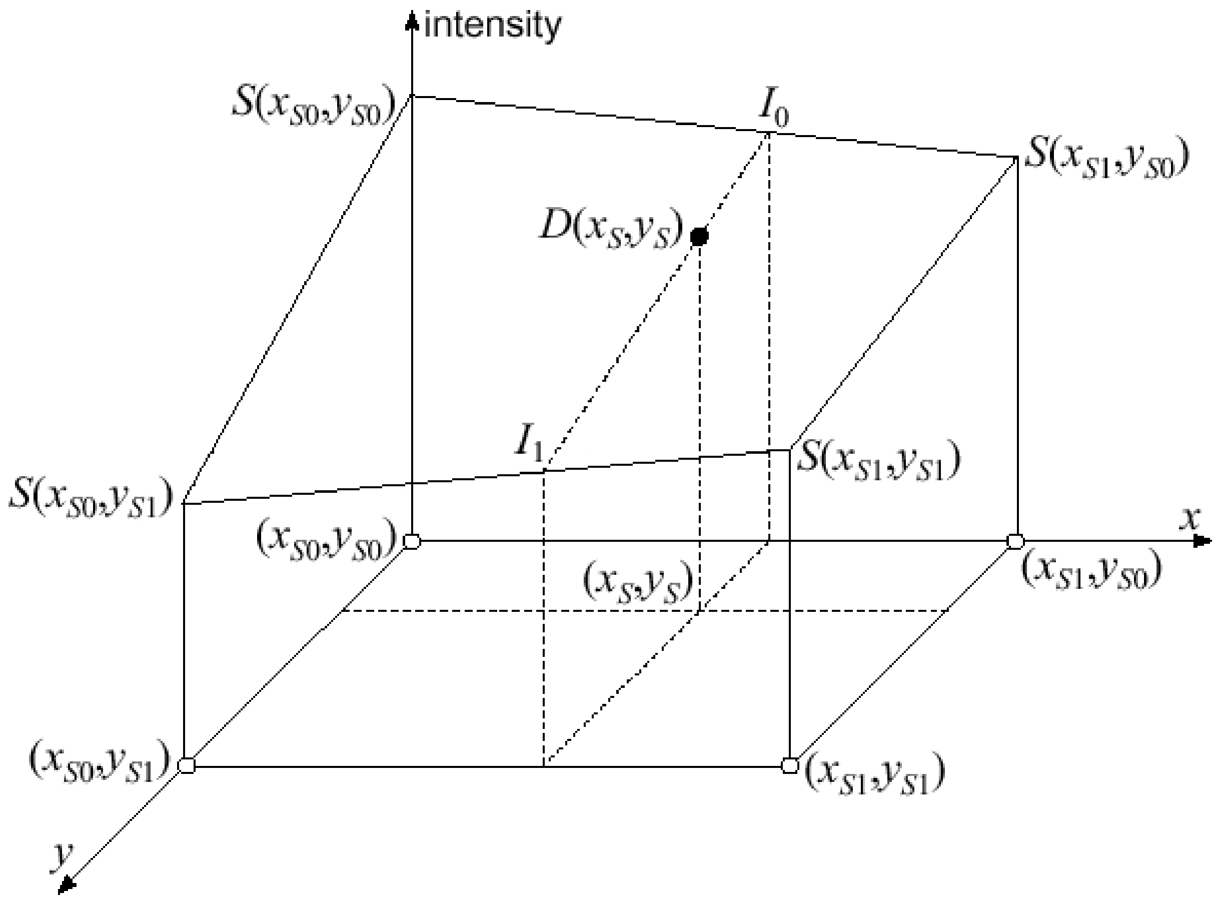
$$\mathbf{I}(x',y') = \begin{bmatrix} 1-b & b \end{bmatrix} \begin{bmatrix} \mathbf{I}(x,y) & \mathbf{I}(x+1,y) \\ \mathbf{I}(x,y+1) & \mathbf{I}(x+1,y+1) \end{bmatrix} \begin{bmatrix} 1-a \\ a \end{bmatrix}$$



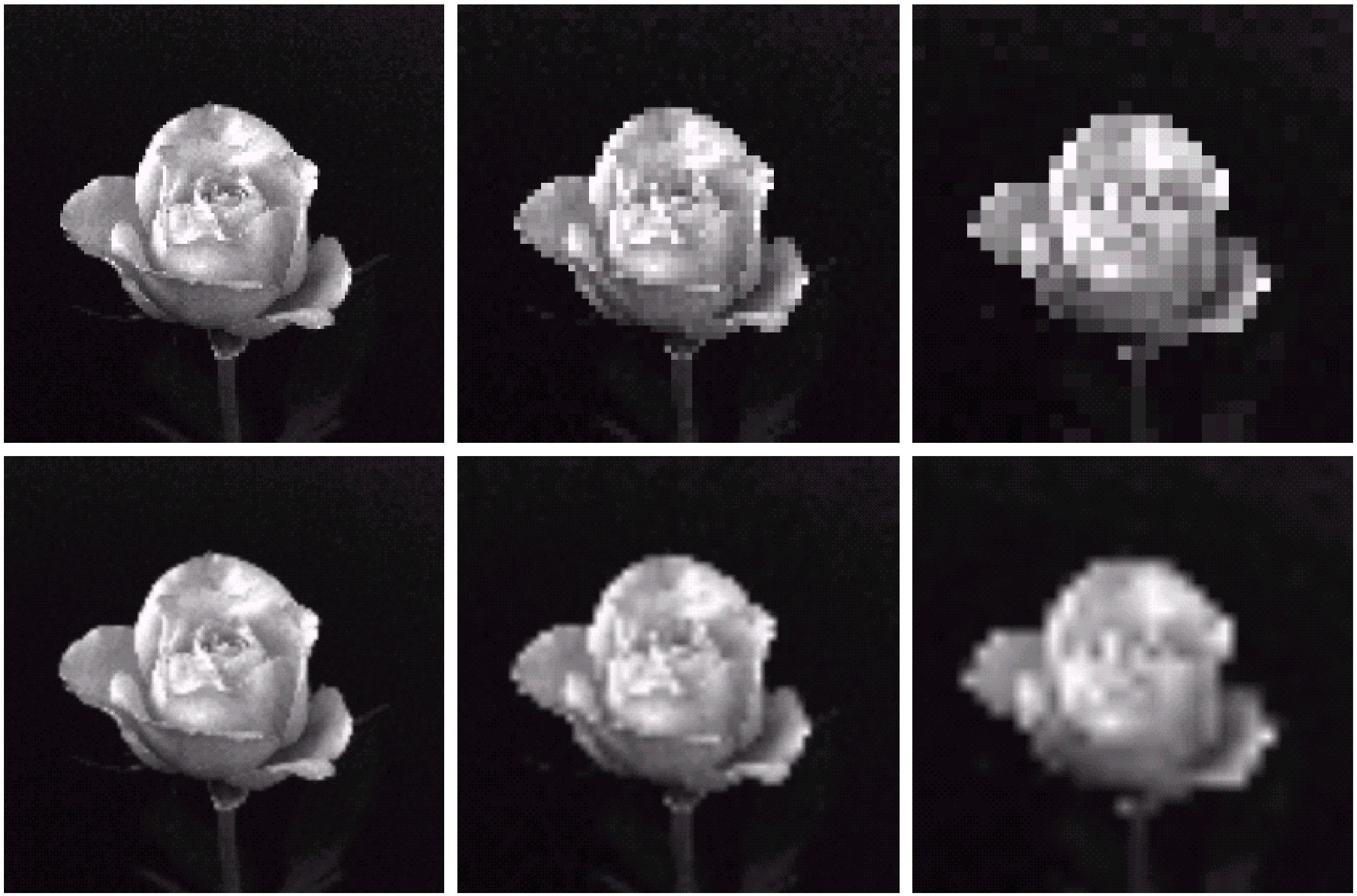


Восстанавливаемая поверхность – линейчатая (не является плоскостью). Ее сечения вдоль координатных плоскостей – прямые.

Билинейная интерполяция



Интерполяция по ближайшему соседу и билинейная



a b c
d e f

FIGURE 2.25 Top row: images zoomed from 128 × 128, 64 × 64, and 32 × 32 pixels to 1024 × 1024 pixels, using nearest neighbor gray-level interpolation. Bottom row: same sequence, but using bilinear interpolation.

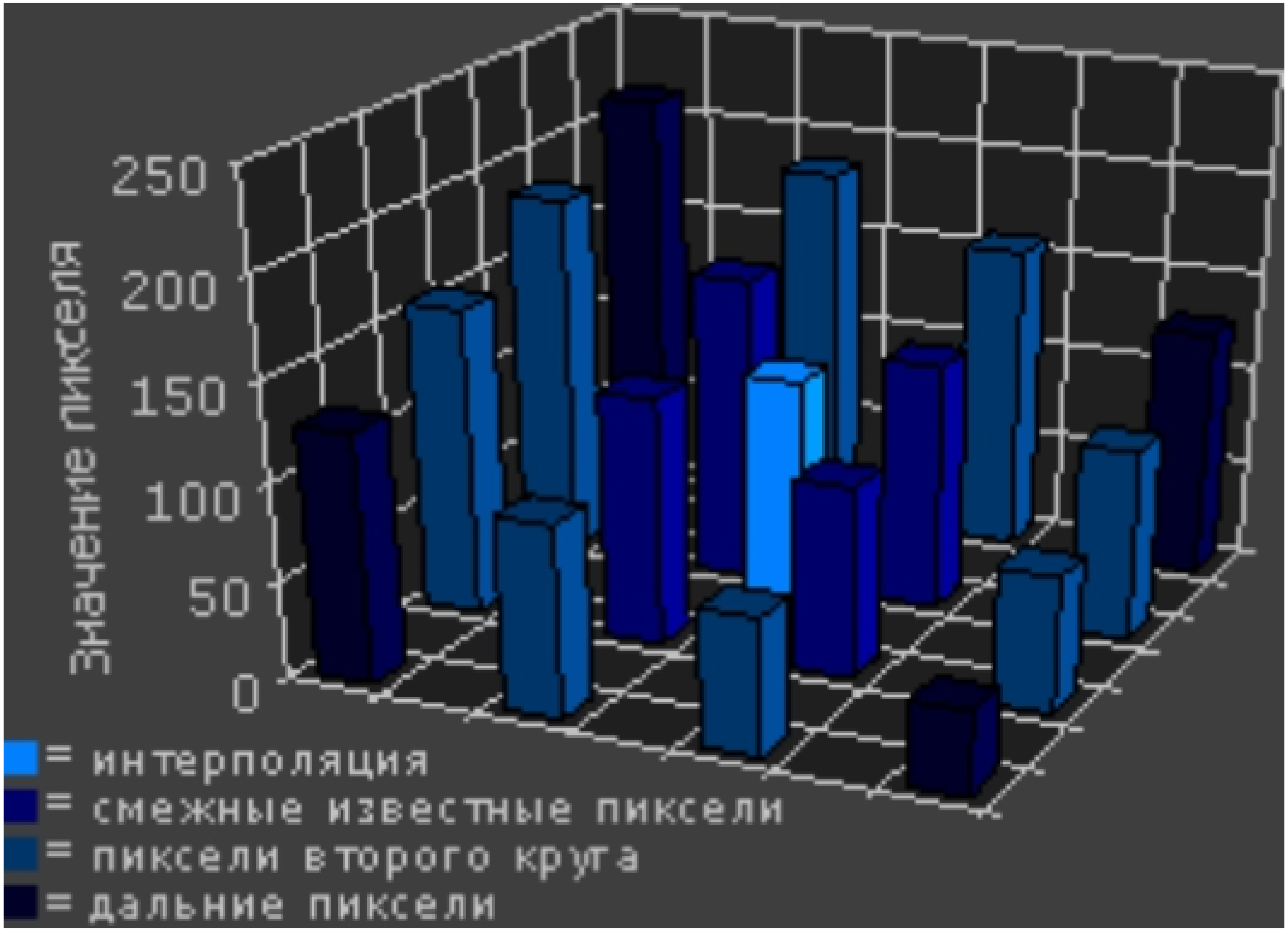
Бикубическая интерполяция

- расширение кубической интерполяции на случай функции двух переменных, значения которой заданы на двумерной регулярной сетке. Поверхность, полученная в результате бикубической интерполяции, является гладкой функцией на границах соседних квадратов, в отличие от поверхностей, полученных в результате билинейной интерполяции или интерполяции методом ближайшего соседа.

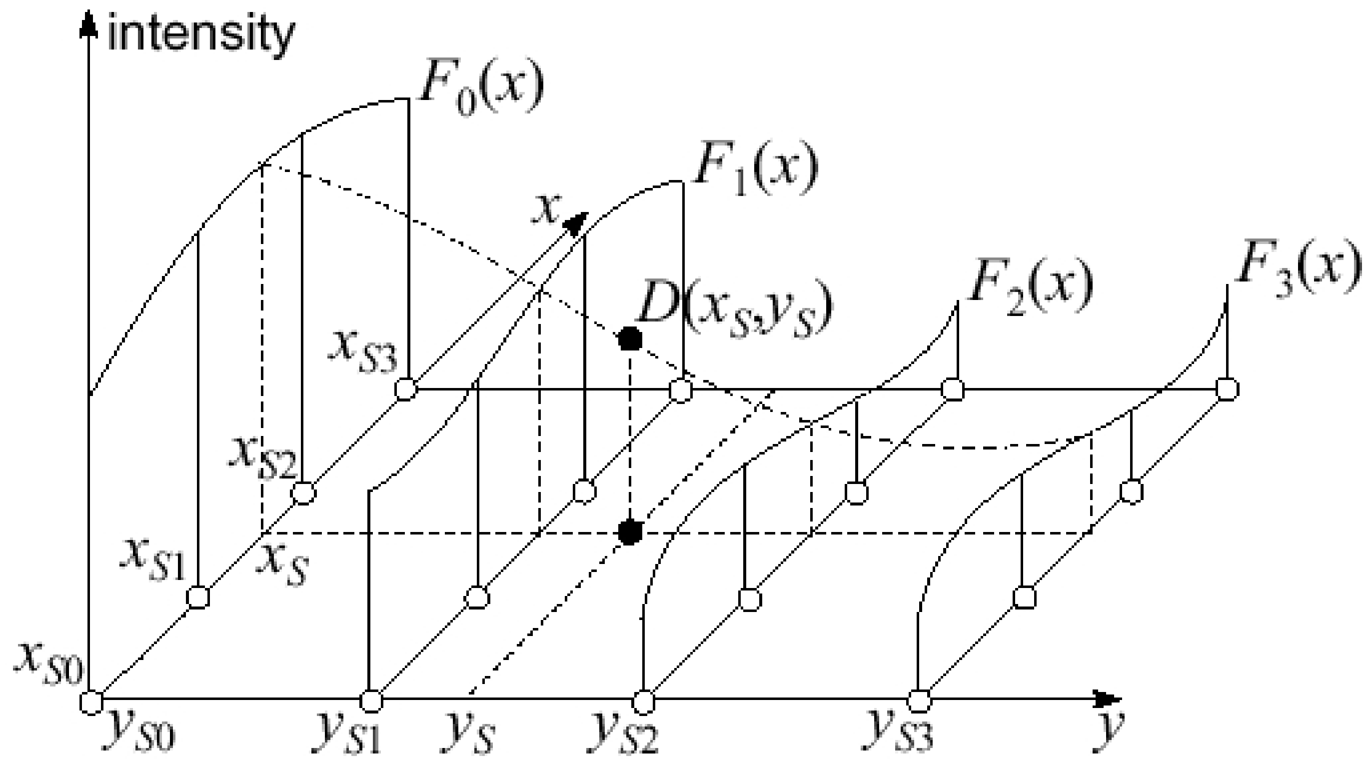
Бикубическая интерполяция

Бикубическая интерполяция рассматривает массив из 4x4 окружающих пикселей — всего 16. Поскольку они находятся на разных расстояниях от неизвестного пикселя, ближайшие пиксели получают при расчёте больший вес. Бикубическая интерполяция производит значительно более резкие изображения, чем предыдущие два метода, и возможно, является оптимальной по соотношению времени обработки и качества на выходе. По этой причине она стала стандартной для многих программ редактирования изображений, драйверов принтеров и встроенной интерполяции камер.

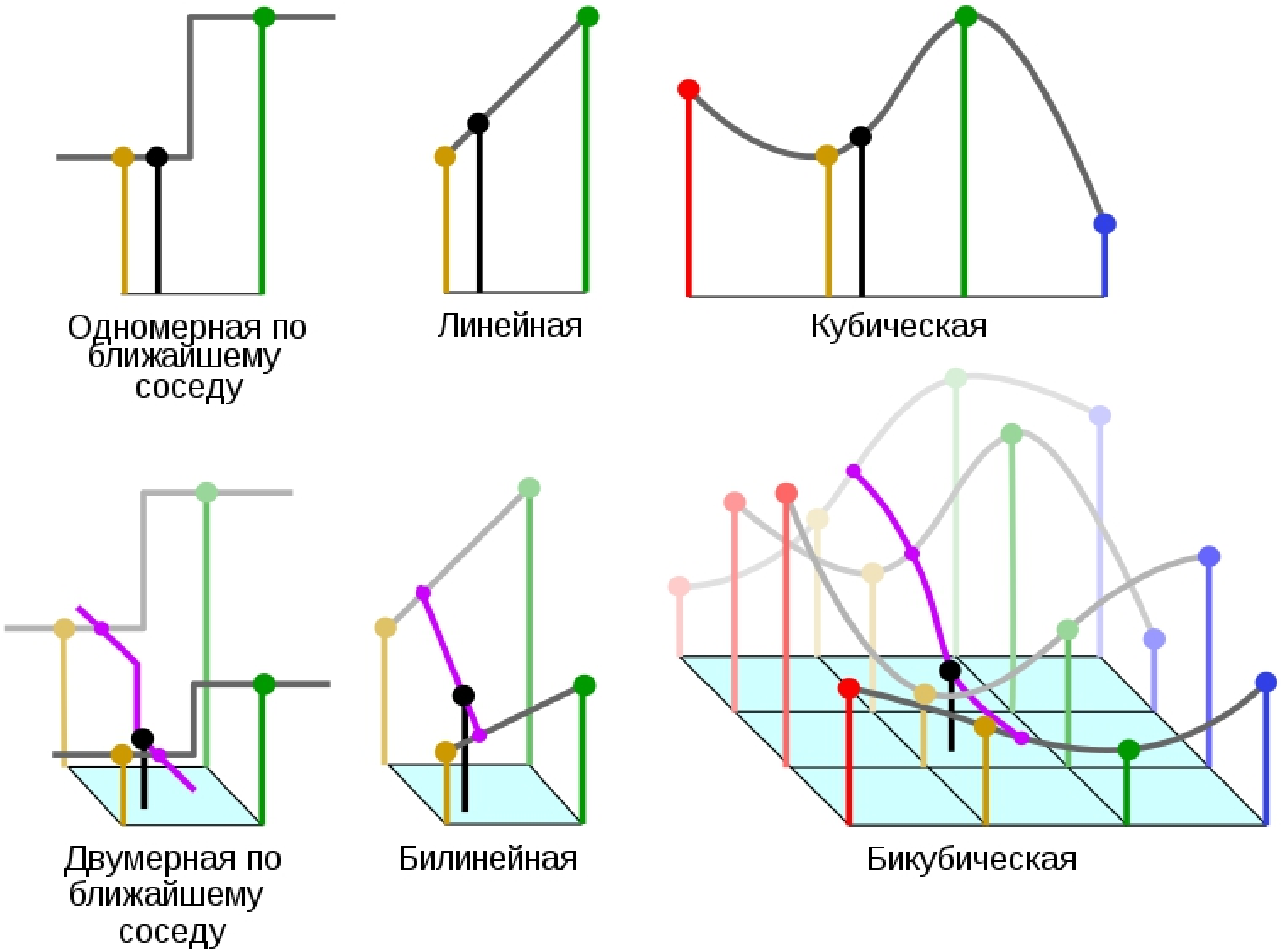
Бикубическая интерполяция



Бикубическая интерполяция



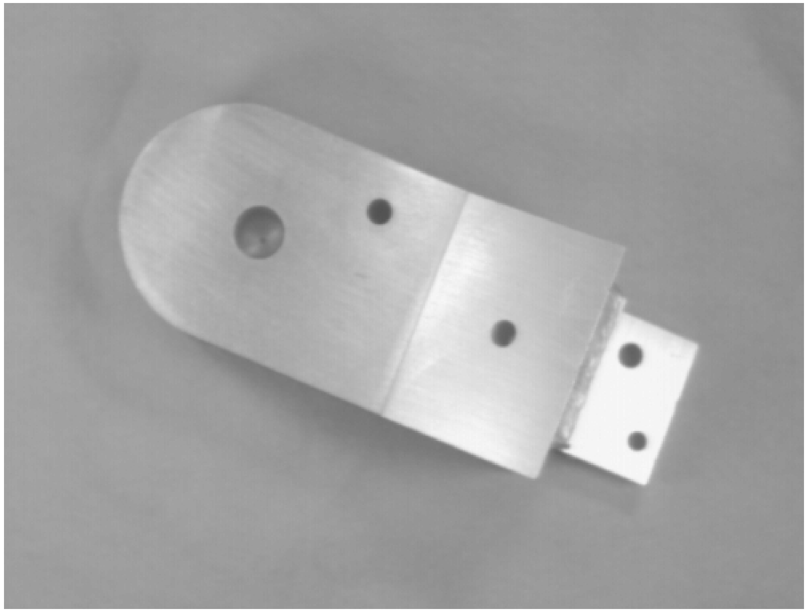
Бикубическая интерполяция



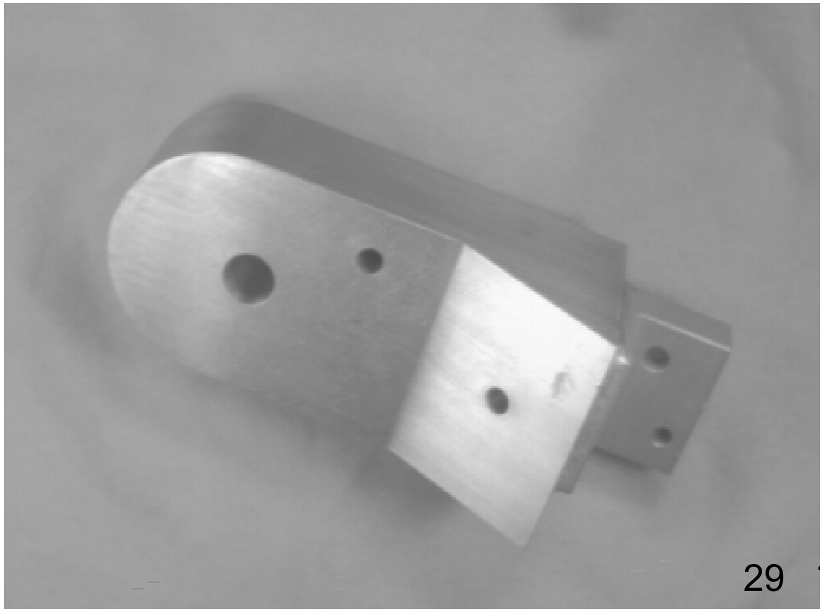
Измерения на изображениях

Измерениям на изображениях довольно часто препятствуют искажения.

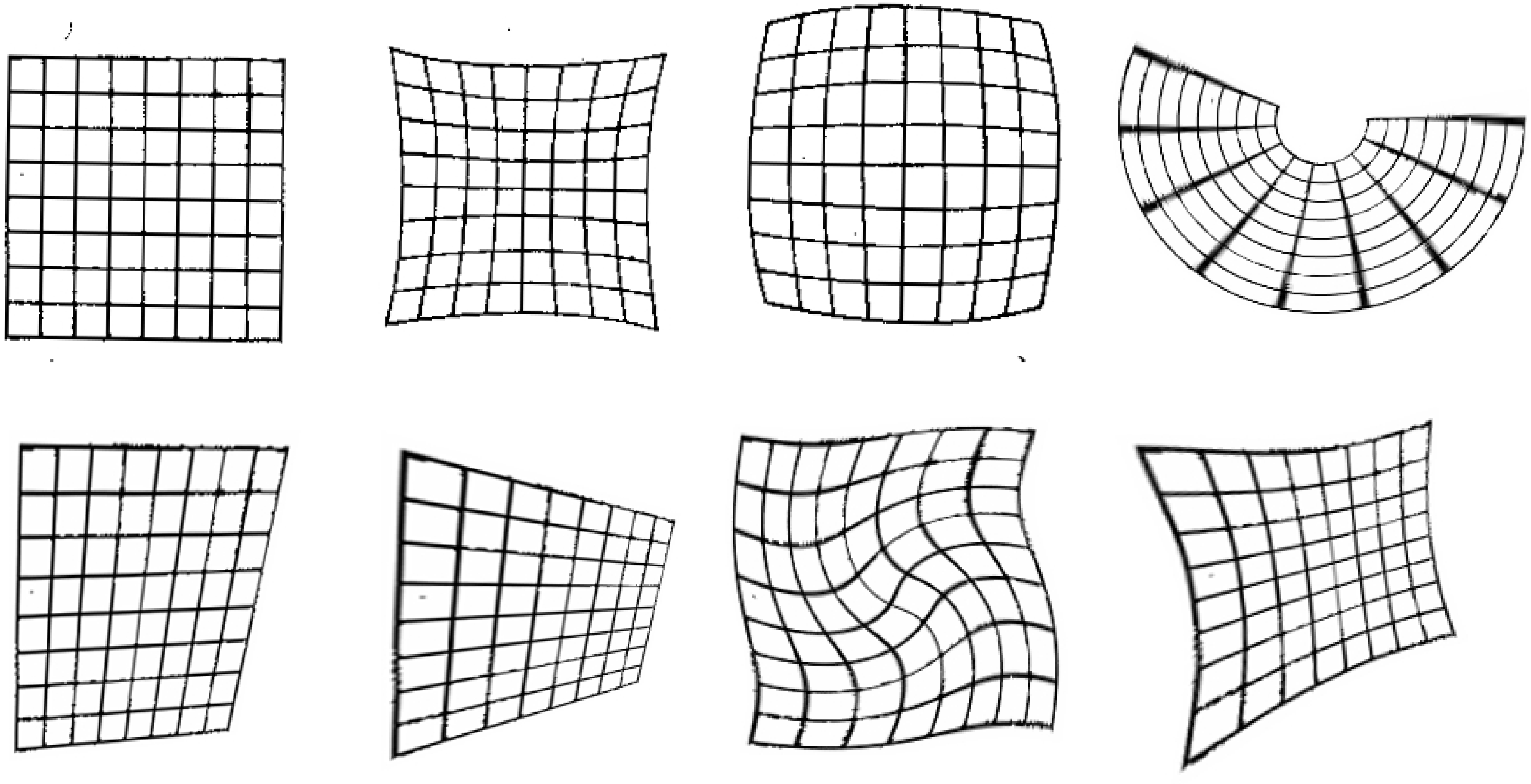
Изображение «в плане»
Сохраняется ортогональность линий, контуры отверстий – окружности.



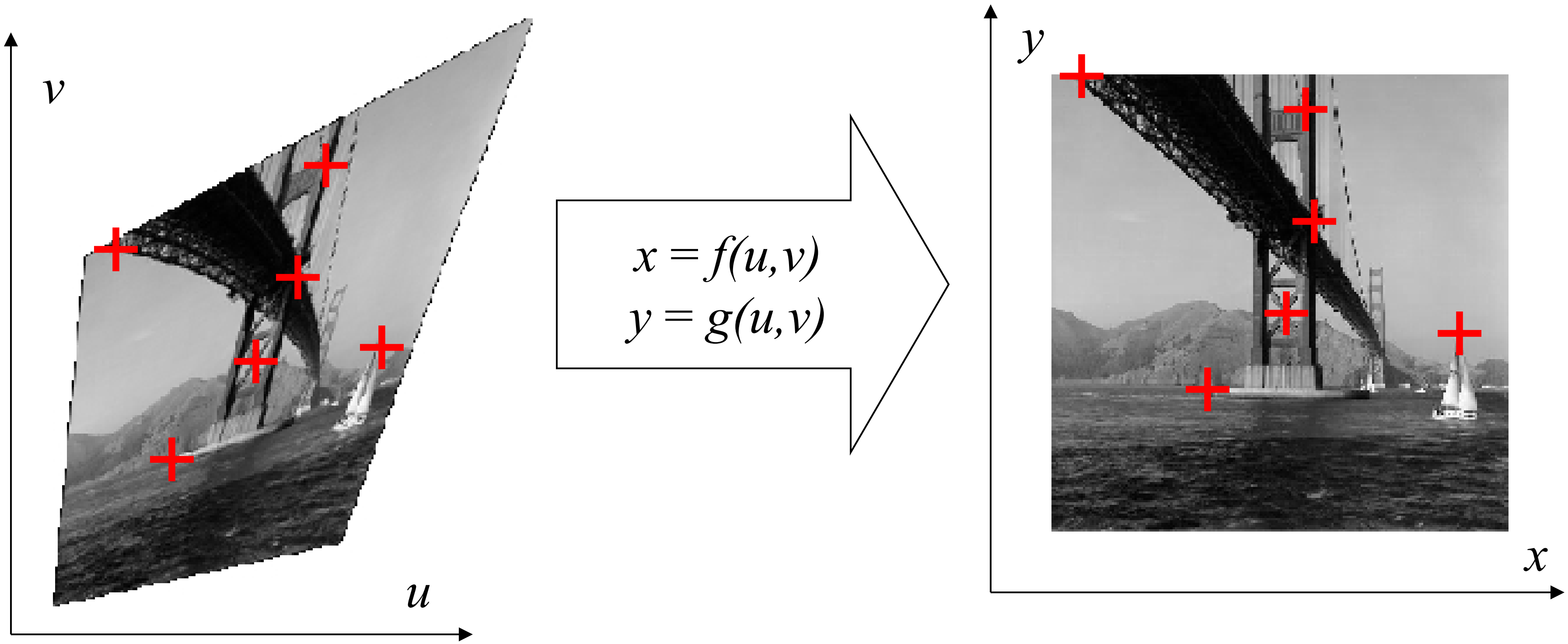
Изображение под углом
Нарушена ортогональность линий, контуры отверстий – эллипсы (аффинные искажения)



Примеры искажений на изображениях



Метод опорных точек



Параметры геометрического преобразования (функций f и g), устраняющего искажения на изображении, восстанавливается путем *интерполяции* по точкам, координаты которых известны как на искаженном так и на откорректированном изображении.

Метод опорных точек. Задача интерполяции

Таблица опорных точек

Искаженное изображение		Результирующее изображение	
u	v	x	y
u_1	v_1	x_1	y_1
u_2	v_2	x_2	y_2
…	…	…	…
u_N	v_N	x_N	y_N

+

Функция преобразования

$$x = f(u,v)$$
$$y = g(u,v)$$

=

Системы уравнений относительно параметров функций f и g

$$x_1 = f(u_1, v_1)$$
$$x_2 = f(u_2, v_2)$$
$$\ldots$$
$$x_N = f(u_N, v_N)$$

$$y_1 = g(u_1, v_1)$$
$$y_2 = g(u_2, v_2)$$
$$\ldots$$
$$y_N = g(u_N, v_N)$$

Метод опорных точек. Задача интерполяции

Вид уравнений в системах зависит от вида функций f и g.

Для линейных преобразований – уравнения линейны.

$$U \cdot A = X$$

$$\begin{aligned}
x_1 &= a_1 u_1 + a_2 v_1 + a_3 \\
x_2 &= a_1 u_2 + a_2 v_2 + a_3 \\
&\ldots \\
x_N &= a_1 u_N + a_2 v_N + a_3
\end{aligned}
\qquad
\begin{pmatrix} u_1 & v_1 & 1 \\ u_2 & v_2 & 1 \\ \vdots & \vdots & \vdots \\ u_N & v_N & 1 \end{pmatrix}
\begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}
=
\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}$$

$$\begin{aligned}
y_1 &= b_1 u_1 + b_2 v_1 + b_3 \\
y_2 &= b_1 u_2 + b_2 v_2 + b_3 \\
&\ldots \\
y_N &= b_1 u_N + b_2 v_N + b_3
\end{aligned}
\qquad
\begin{pmatrix} u_1 & v_1 & 1 \\ u_2 & v_2 & 1 \\ \vdots & \vdots & \vdots \\ u_N & v_N & 1 \end{pmatrix}
\begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix}
=
\begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}$$

$$U \cdot B = Y$$

Метод опорных точек. Задача интерполяции

Вид уравнений в системах зависит от вида функций f и g.

Нелинейные преобразования часто аппроксимируют полиномами двух переменных.

$$x = \sum_{i=0}^{n} \sum_{j=0}^{n-i} a_{ij} u^i v^j$$

$$y = \sum_{i=0}^{n} \sum_{j=0}^{n-i} b_{ij} u^i v^j$$

Пример для $n = 2$:

$$x = a_{00} + a_{01} v + a_{02} v^2 + a_{10} u + a_{11} uv + a_{20} u^2$$

$$y = b_{00} + b_{01} v + b_{02} v^2 + b_{10} u + b_{11} uv + b_{20} u^2$$

Метод опорных точек. Задача интерполяции

Выбор порядка n интерполирующего полинома – решение задачи о структуре и шуме в исходных данных.

На практике порядок полинома редко выбирается большим 3.

Метод опорных точек. Задача интерполяции

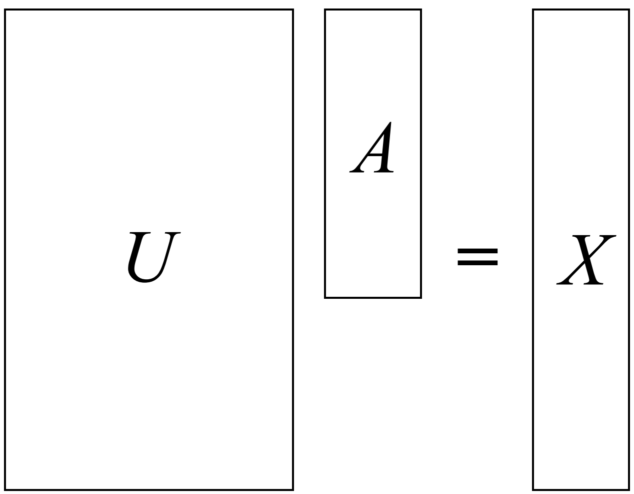
Параметры функций f и g отыскиваются из систем уравнений:

$$U \cdot A = X \qquad\qquad U \cdot B = Y$$

$$\begin{matrix} 1 & v_1 & v_1^2 & u_1 & u_1 v_1 & u_1^2 \\ 1 & v_2 & v_2^2 & u_2 & u_2 v_2 & u_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & v_N & v_N^2 & u_N & u_N v_N & u_N^2 \end{matrix} \quad \begin{matrix} a_{00} \\ a_{01} \\ a_{02} \\ a_{10} \\ a_{11} \\ a_{20} \end{matrix} = \begin{matrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{matrix} \qquad \begin{matrix} 1 & v_1 & v_1^2 & u_1 & u_1 v_1 & u_1^2 \\ 1 & v_2 & v_2^2 & u_2 & u_2 v_2 & u_2^2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & v_N & v_N^2 & u_N & u_N v_N & u_N^2 \end{matrix} \quad \begin{matrix} b_{00} \\ b_{01} \\ b_{02} \\ b_{10} \\ b_{11} \\ b_{20} \end{matrix} = \begin{matrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{matrix}$$

Решение переопределенных систем линейных уравнений

Система уравнений $U{\cdot}A = X$, как правило, переопределена, т.е. количество уравнений превышает количество переменных.



Параметры преобразования (вектор A) могут быть оценены, например, методом наименьших квадратов, т.е. выбраны так, чтобы минимизировать среднеквадратичную ошибку аппроксимации фактически «наблюдаемых» координат (X) их оценкой ($U{\cdot}A$) для набора заданных опорных точек.

Глобальные и локальные преобразования

- Все рассмотренные выше типы преобразований относятся к глобальным
- Характер воздействия глобального преобразования не зависит от (местоположения) преобразуемой точки
- В некоторых случаях вид искажений может существенно зависеть от местоположения на изображении
- Трансформирующая функция должна моделировать локальные особенности искажений в каждой области изображения

Локальные преобразования

- Кусочно-линейное преобразование (метод конечных элементов)
- Функции радиального базиса
  - Функции Грина
  - Мультиквадрики Харди

Кусочно-линейное преобразование

- В зависимости от формы конечных элементов преобразование может быть
  - линейным (для треугольников)
  - билинейным (для прямоугольников)

Вид преобразования

$$x = a_0 + a_1 u + a_2 v$$
$$y = b_0 + b_1 u + b_2 v$$

$$x = a_0 + a_1 u + a_2 v + a_3 uv$$
$$y = b_0 + b_1 u + b_2 v + b_3 uv$$

Системы уравнений для поиска коэффициентов

$$\begin{array}{cc} x_0 & y_0 \\ x_1 & y_1 \\ x_2 & y_2 \end{array} = \begin{array}{ccc} u_0 & v_0 & 1 \\ u_1 & v_1 & 1 \\ u_2 & v_2 & 1 \end{array} \begin{array}{cc} a_2 & b_2 \\ a_1 & b_1 \\ a_0 & b_0 \end{array}$$

$$\begin{array}{cc} x_0 & y_0 \\ x_1 & y_1 \\ x_2 & y_2 \\ x_3 & y_3 \end{array} = \begin{array}{cccc} u_0 v_0 & u_0 & v_0 & 1 \\ u_1 v_1 & u_1 & v_1 & 1 \\ u_2 v_2 & u_2 & v_2 & 1 \\ u_3 v_3 & u_3 & v_3 & 1 \end{array} \begin{array}{cc} a_3 & b_3 \\ a_2 & b_2 \\ a_1 & b_1 \\ a_0 & b_0 \end{array}$$

Функции радиального базиса

- Все преобразования данного типа являются глобально-локальными
  – Глобальны, поскольку для построения преобразования задействуются все контрольные точки
  – Локальны, так как на отклик преобразования в определенной точке контрольные точки влияют с разными весами: чем ближе контрольная точка, тем больший вклад она дает

Функции Грина

- Функции Грина также называют thin plate splines (TPS, TP-сплайны)

Вид преобразования

$$x = F(u,v) = a_0 + a_1 u + a_2 v + \sum_{i=1}^{N} f_i r_i^2 \ln r_i^2$$

$$y = G(u,v) = b_0 + b_1 u + b_2 v + \sum_{i=1}^{N} g_i r_i^2 \ln r_i^2$$

$$r_i^2 = \left[r_i(u,v)^2 \right] = (u - u_i)^2 + (v - v_i)^2$$

$$\sum_{i=1}^{N} f_i = \sum_{i=1}^{N} f_i u_i = \sum_{i=1}^{N} f_i v_i = 0$$

$$\sum_{i=1}^{N} g_i = \sum_{i=1}^{N} g_i u_i = \sum_{i=1}^{N} g_i v_i = 0$$

Функции Грина

Система уравнений на параметры

$$
\begin{array}{ccccccccccc}
0 & 0 & 0 & 1 & 1 & \dots & 1 & a_0 & b_0 & & 0 & 0 \\
0 & 0 & 0 & u_1 & u_2 & \dots & u_n & a_1 & b_1 & & 0 & 0 \\
0 & 0 & 0 & v_1 & v_2 & \dots & v_n & a_2 & b_2 & & 0 & 0 \\
1 & u_1 & v_1 & 0 & r_{12}^2 \ln r_{12} & \dots & r_{1n}^2 \ln r_{1n} & f_1 & g_1 & = & x_1 & y_1 \\
1 & u_2 & v_2 & r_{21}^2 \ln r_{21} & 0 & \dots & r_{2n}^2 \ln r_{2n} & f_2 & g_2 & & x_2 & y_2 \\
\dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & & \dots & \dots \\
1 & u_n & v_n & r_{n1}^2 \ln r_{n1} & r_{n2}^2 \ln r_{n2} & \dots & 0 & f_n & g_n & & x_n & y_n
\end{array}
$$

где $r_{ij} = r_i(u_j,v_j)$, т.е. расстоянию от контрольной точки (x_i,y_i) до контрольных точек (u_j,v_j)

Мультиквадрики Харди

Вид преобразования

$$x = F(u,v) = \sum_{i=1}^{N} f_i \sqrt{(x - x_i)^2 + (y - y_i)^2 + R^2} = \sum_{i=1}^{N} f_i \sqrt{(r_i^2) + R^2}$$

$$y = G(u,v) = \sum_{i=1}^{N} g_i \sqrt{(x - x_i)^2 + (y - y_i)^2 + R^2} = \sum_{i=1}^{N} g_i \sqrt{(r_i^2) + R^2}$$

или

$$x = F(u,v) = \sum_{i=1}^{N} f_i \, 1 / \sqrt{(x - x_i)^2 + (y - y_i)^2 + R^2} = \sum_{i=1}^{N} f_i \, 1 / \sqrt{(r_i^2) + R^2}$$

$$y = G(u,v) = \sum_{i=1}^{N} g_i \, 1 / \sqrt{(x - x_i)^2 + (y - y_i)^2 + R^2} = \sum_{i=1}^{N} g_i \, 1 / \sqrt{(r_i^2) + R^2}$$

где

$$R^2 = 0.6 \min\left(\sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}\right)$$

Мультиквадрики Харди

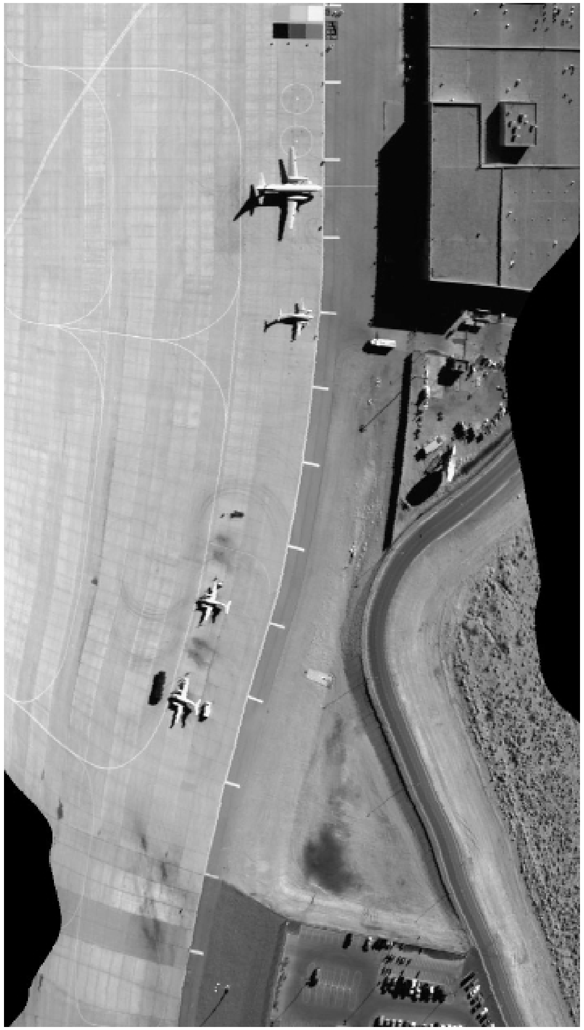
Система уравнений на параметры

$$
\begin{array}{ccccc}
0 & 1 & 1 & \dots & 1 \\
1 & \sqrt{r_{11}^2 + R^2} & \sqrt{r_{12}^2 + R^2} & \dots & \sqrt{r_{1n}^2 + R^2} \\
1 & \sqrt{r_{21}^2 + R^2} & \sqrt{r_{22}^2 + R^2} & \dots & \sqrt{r_{2n}^2 + R^2} \\
\dots & \dots & \dots & \dots & \dots \\
1 & \sqrt{r_{n1}^2 + R^2} & \sqrt{r_{n2}^2 + R^2} & \dots & \sqrt{r_{nn}^2 + R^2}
\end{array}
\quad
\begin{array}{cc}
a_0 & b_o \\
f_1 & g_1 \\
f_2 & g_2 \\
\dots & \dots \\
f_n & g_n
\end{array}
=
\begin{array}{cc}
0 & 0 \\
x_1 & y_1 \\
x_2 & y_2 \\
\dots & \dots \\
x_n & y_n
\end{array}
$$

где $r_{ij} = r_i(u_j, v_j)$, т.е. расстоянию от контрольной точки (u_i, y_i) до контрольных точек (u_j, v_j)

Примеры действия преобразований

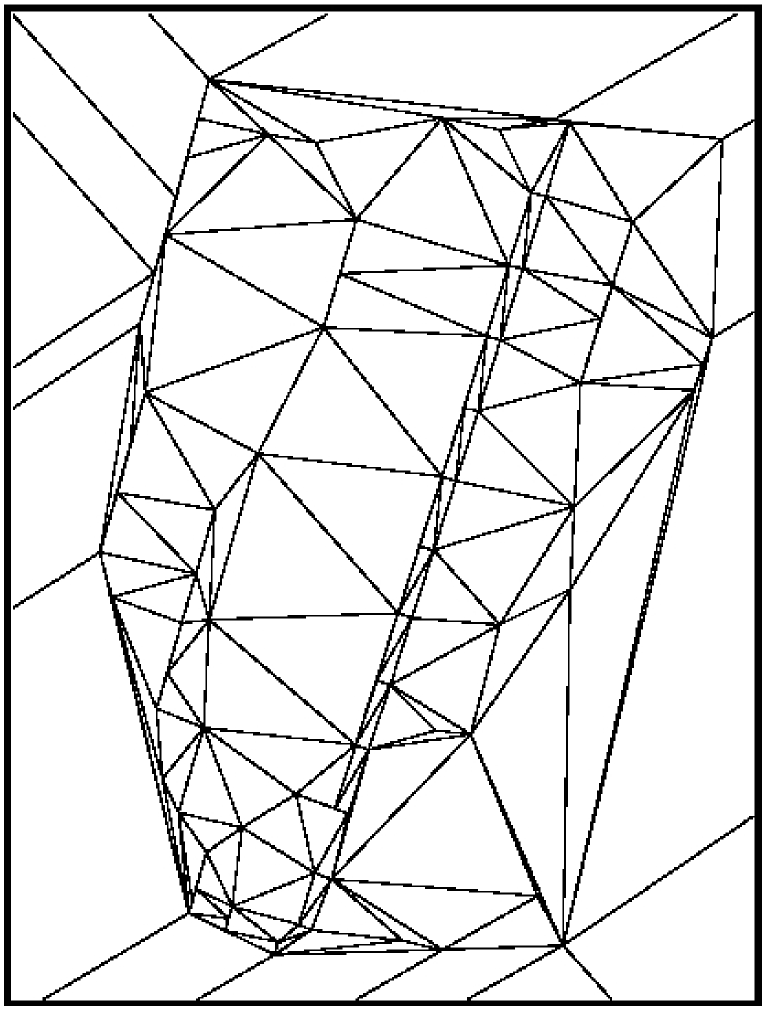
Искаженное изображение



Эталонное изображение

Триангуляция множества контрольных точек

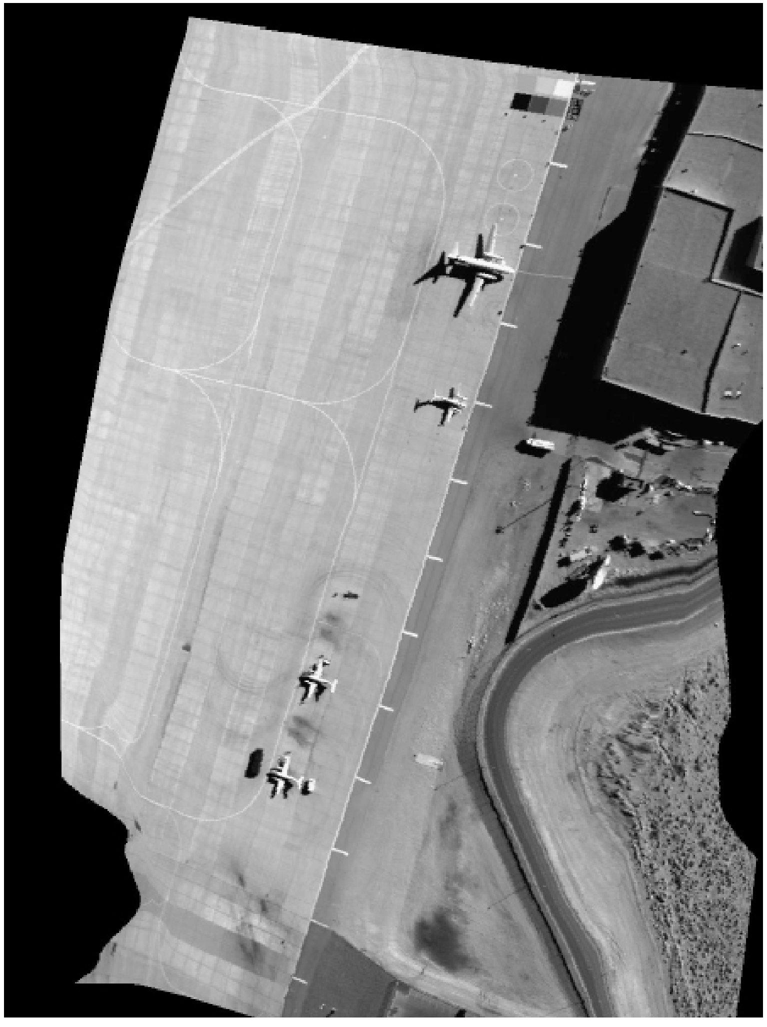
Триангуляция



Эталонное изображение

Кусочно-линейное преобразование

Преобразованное изображение



Эталонное изображение

Полином первого порядка

Преобразованное изображение

Эталонное изображение

Полином третьего порядка

Преобразованное изображение

Эталонное изображение

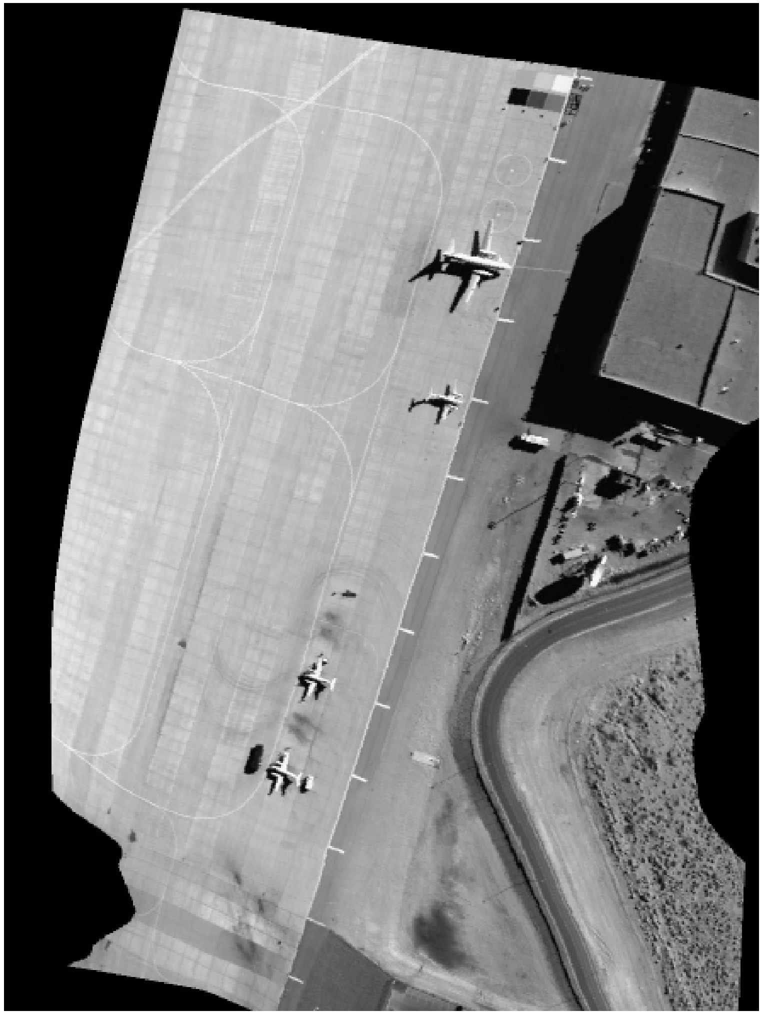
Полином третьего порядка+ мультиквадрики

Преобразованное изображение

Эталонное изображение

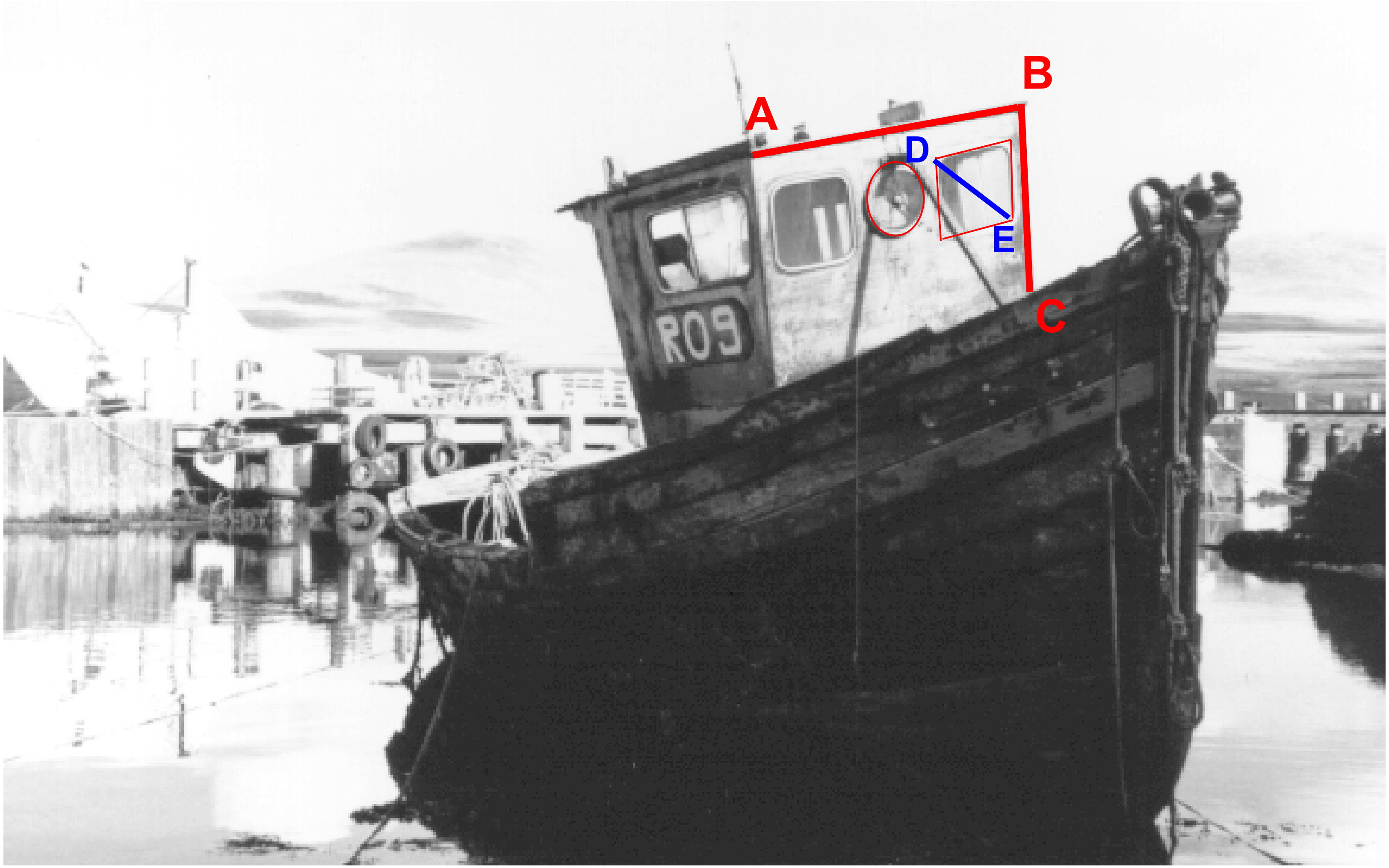
Функции Грина

Преобразованное изображение

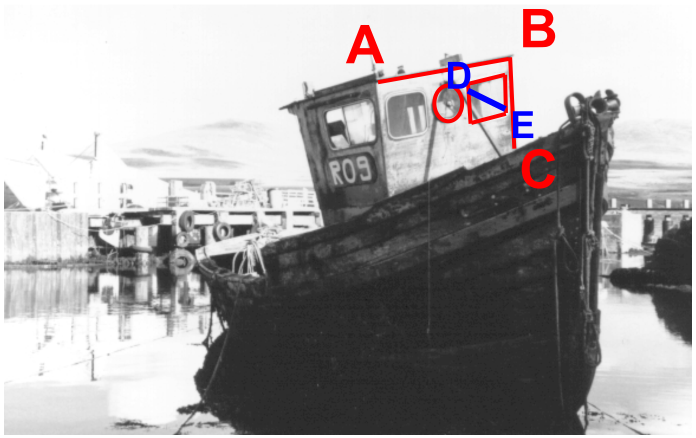


Эталонное изображение

Измерения на изображениях



Измерения на изображениях



Дано:

AB = 2 м

BC = 1.35 м

∠ABC = 90°

Найти:

DE = ? м

Решение:

В пиксельных координатах: A = (314,64), B=(428,43), C=(433,122),
AB = 116 px, BC = 79 px   Res=2/116 м/px

Строим A', B', C': B' = B, A' = (B'_x – AB, B'_y), C' = (B'_x, B'_y + BC)

Строим аффинное преобразование, переводящее A, B, C в A', B', C'.

$$x = a*u+b*v + c$$
$$y = d*u + e*v +f$$

Преобразуем изображение.

Измерения на изображениях

Измеряем DE: DE = 40 px*Res = 0.68 м