

Digital Image & Video Processing

Lecture 4

Image Pre-processing (Geometric Transformations)

Lecturer: Associate Prof. Lý Quốc Ngọc

3. Image Pre-processing

3.1. Color Transformations

3.2. Geometric Transformations

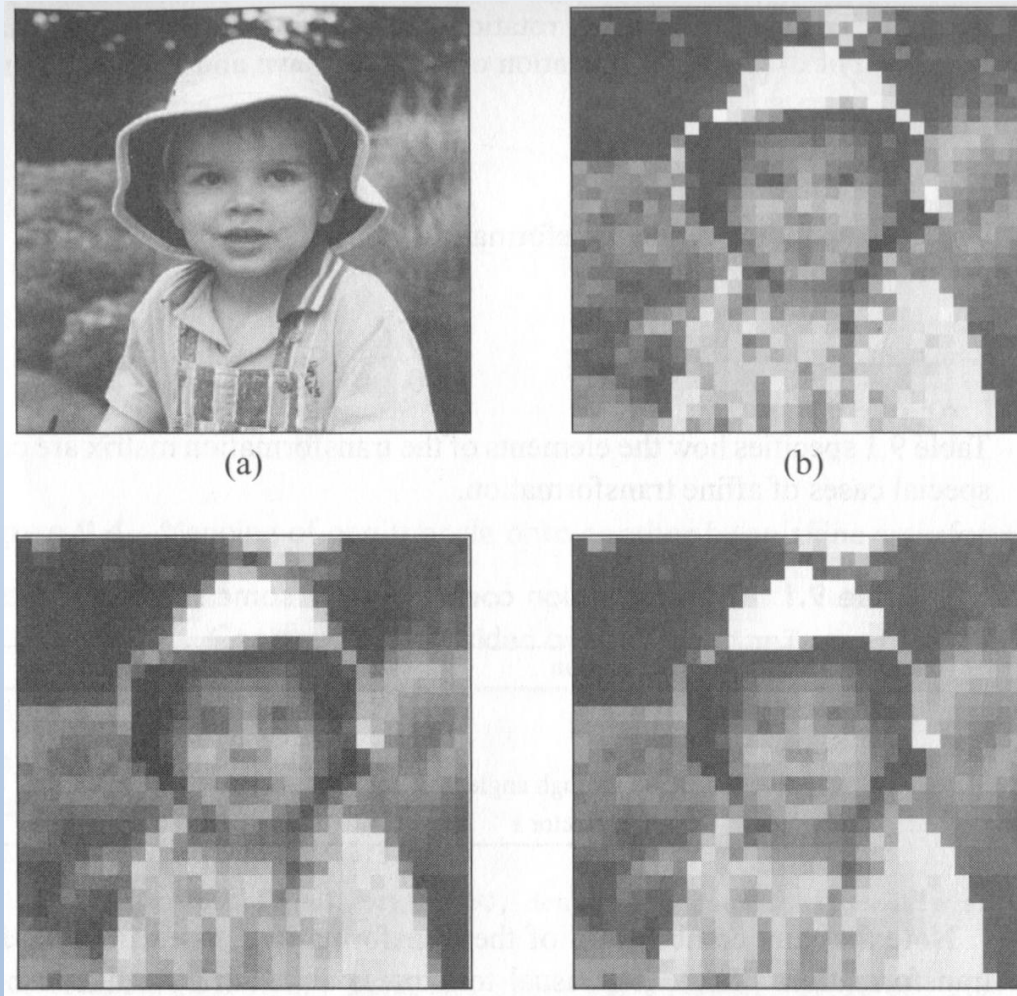
3.3. Local Pre-processing

3.2. Geometric Transformations

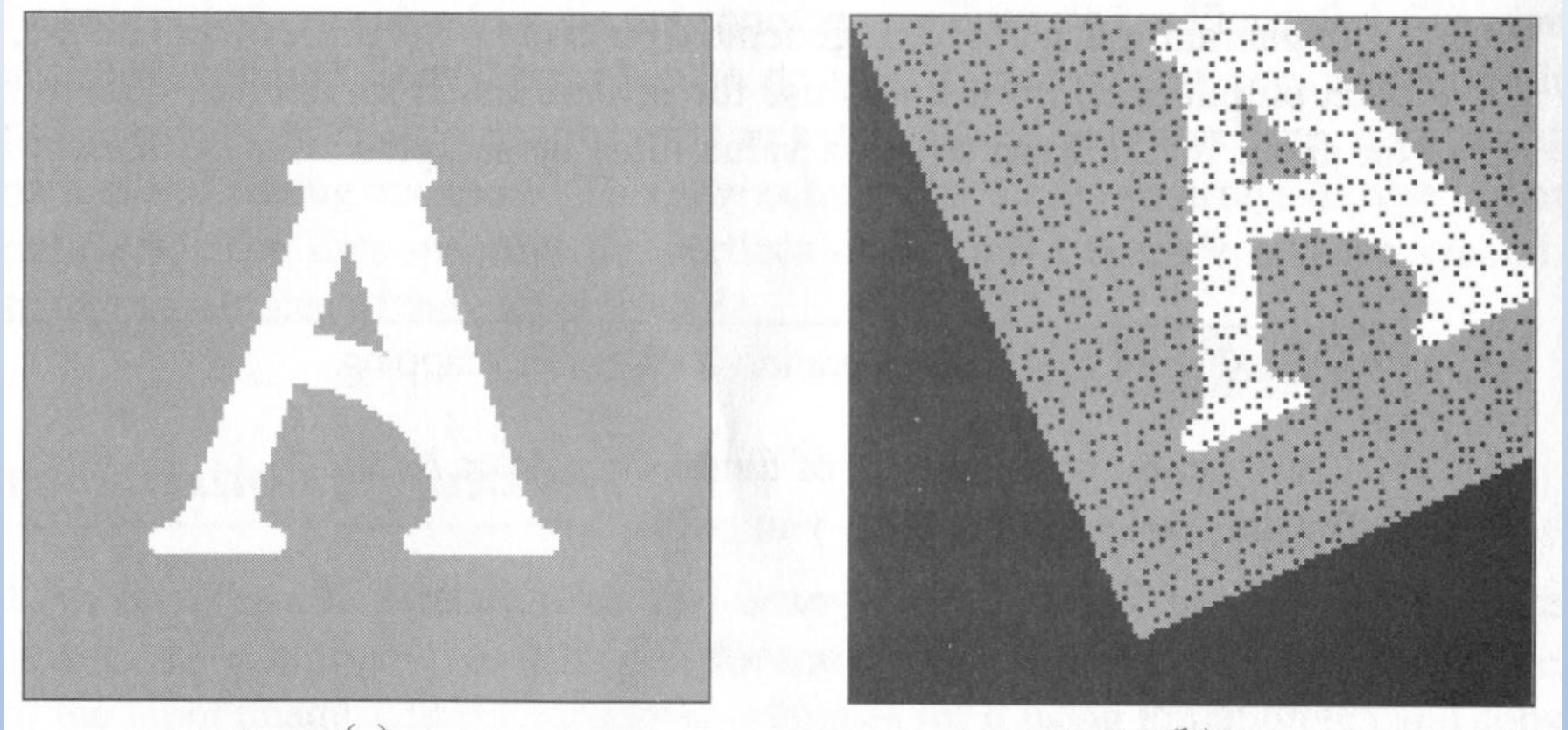
3.2.1. Pixel co-ordinate transformations

3.2.2. Brightness interpolation

3.2. Geometric Transformations



3.2. Geometric Transformations



3.2. Geometric Transformations

3.2.1. Pixel co-ordinate transformations

The co-ordinates of a point in the output image after a geometric transform:

$$x' = T_x(x, y), \quad y' = T_y(x, y)$$

It is usually approximated by a polynomial equation

$$x' = \sum_{r=0}^m \sum_{k=0}^{m-r} a_{rk} x^r y^k, \quad y' = \sum_{r=0}^m \sum_{k=0}^{m-r} b_{rk} x^r y^k$$

3.2. Geometric Transformations

3.2.1. Pixel co-ordinate transformations

Bilinear transform:

$$x' = a_0 + a_1x + a_2y + a_3xy$$

$$y' = b_0 + b_1x + b_2y + b_3xy$$

Affine transform:

$$x' = a_0 + a_1x + a_2y$$

$$y' = b_0 + b_1x + b_2y$$

3.2. Geometric Transformations

3.2.1. Pixel co-ordinate transformations

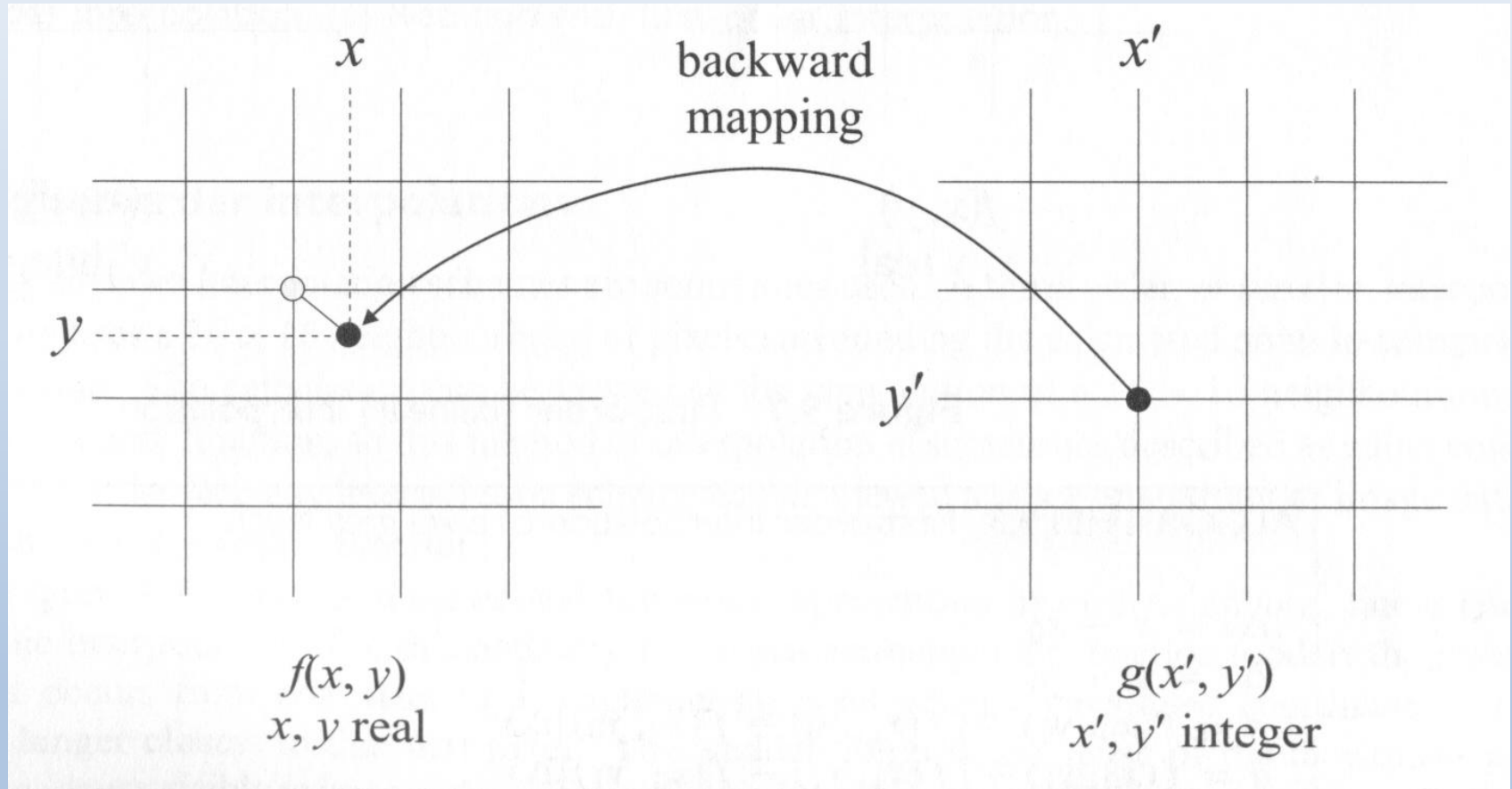
✚ The new point co-ordinates (x', y') **does not** in general **fit the discrete raster** of the output image and they have **non-integer co-ordinates**.

✚ One solution is to compute the brightness value of pixel (x', y') on the discrete raster in the output image by **inverting the planar transformation**:

$$(x, y) = T^{-1}(x', y')$$

3.2. Geometric Transformations

3.2.1. Pixel co-ordinate transformations



3.2. Geometric Transformations

3.2.2. Brightness interpolation

✚ Nearest-neighborhood interpolation

$$f'(x', y') = f(\text{round}(x), \text{round}(y))$$

✚ Linear interpolation

$$\begin{aligned} f'(x', y') = & (1-a)(1-b)f(l, k) + a(1-b)f(l+1, k) \\ & + b(1-a)f(l, k+1) + abf(l+1, k+1) \end{aligned}$$

$$l = \text{round}(x), a = x - l; \quad k = \text{round}(y), b = y - k$$