SUMMARY

2016.10.8

Knowledge

Fault&Experience

Plan

Finished Codes

Reduce Spatial Resolution

Image Interpolation

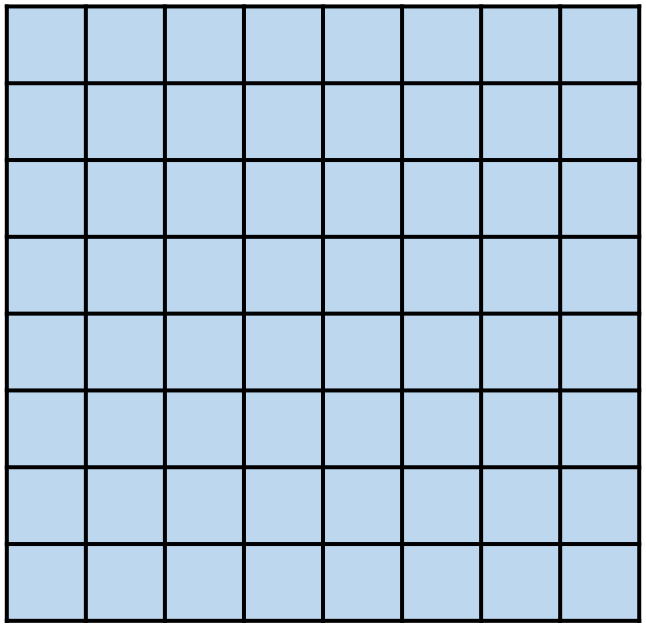
Arithmetic Operations

Set Operations

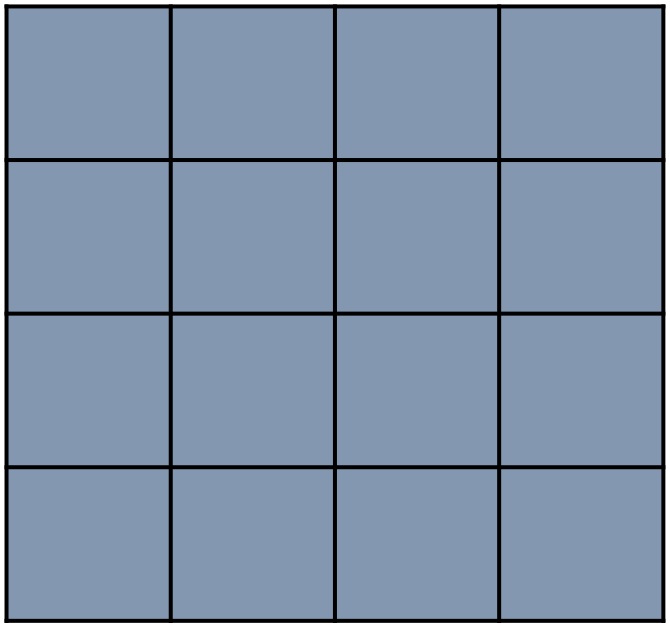
Geometric Spatial Transformations

Reduce Spatial Resolution

Reduce Spatial Resolution

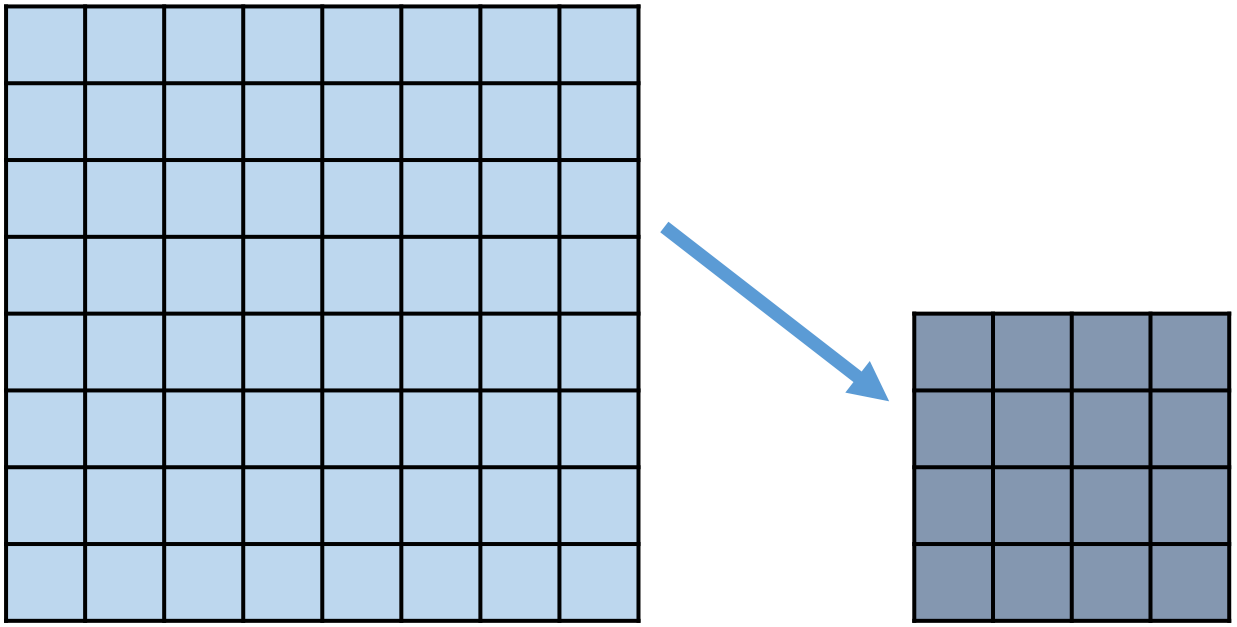


a



b

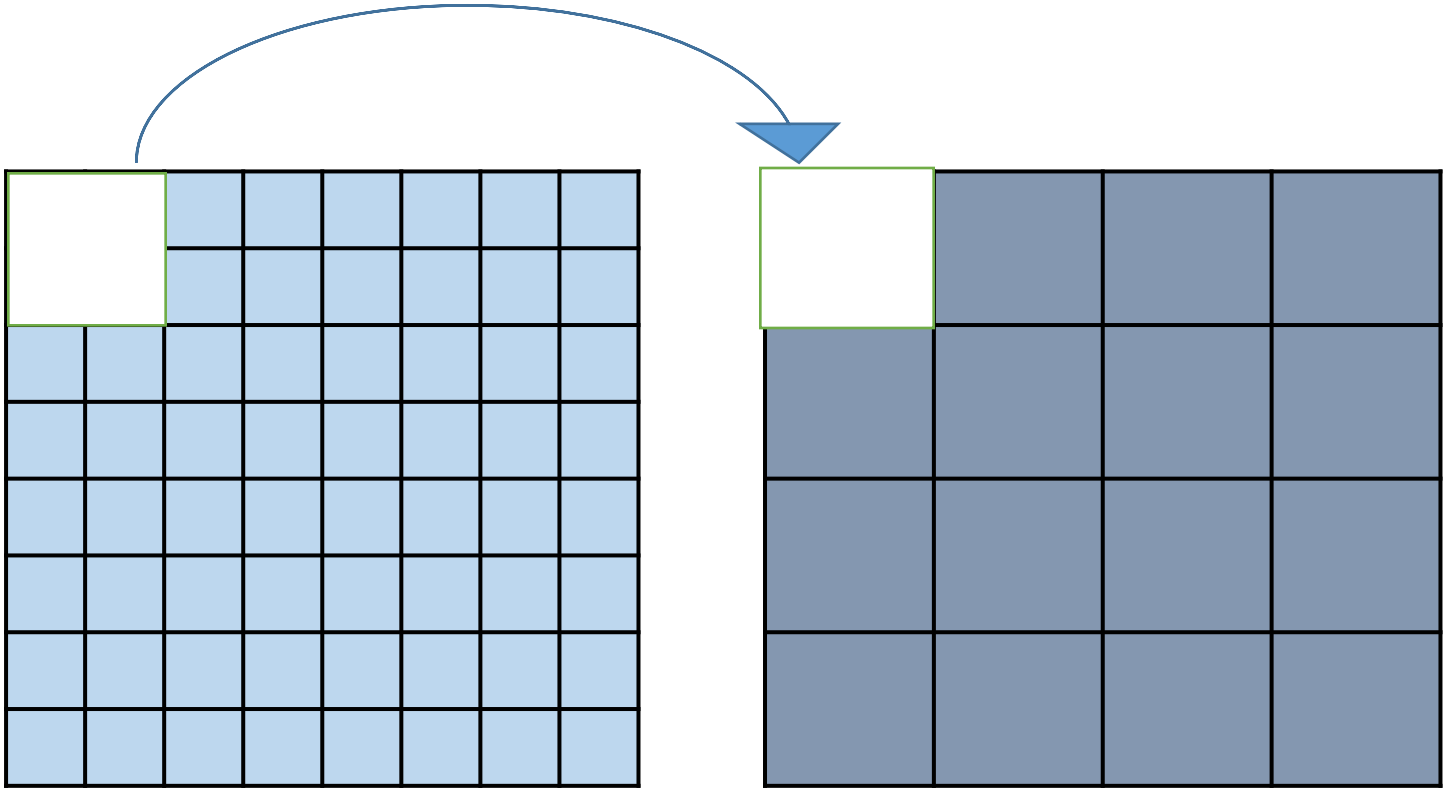
Reduce Spatial Resolution



8 X 8

4 X 4

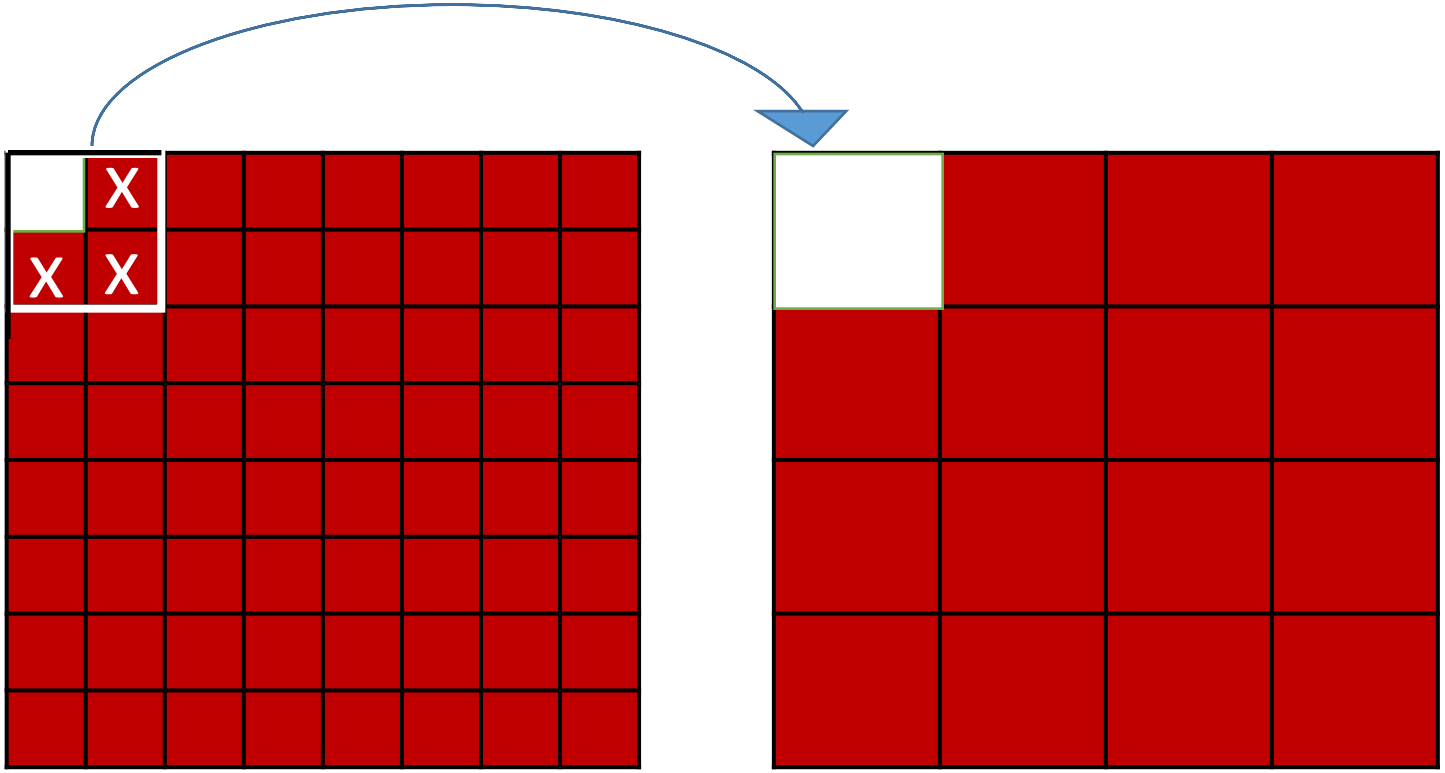
Reduce Spatial Resolution



8 X 8

4 X 4

Fault&Experience



Reduce Spatial Resolution

Create a new image

↓

Traverse all pixels in new image

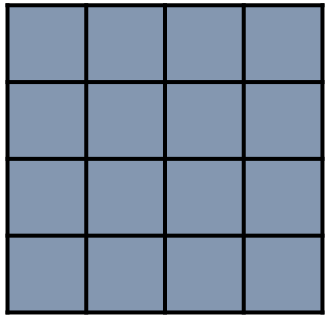
↓

Assignment by calculating

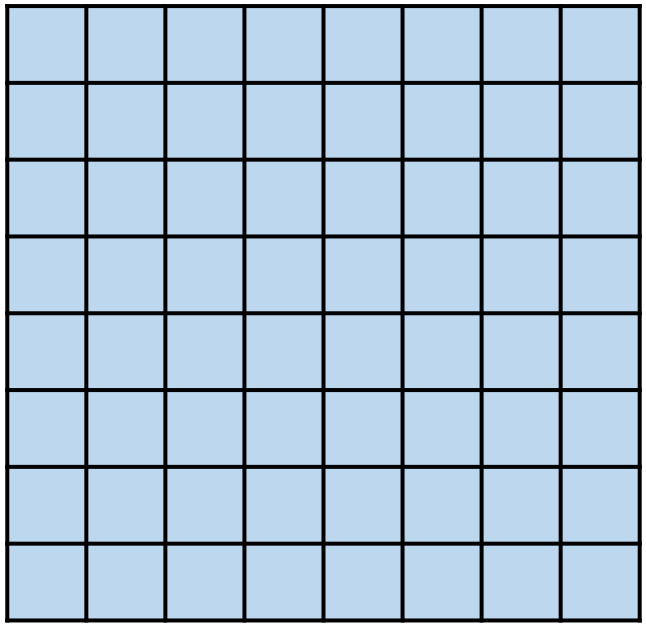
Before

After

Question



4 X 4



8 X 8

Image Interpolation

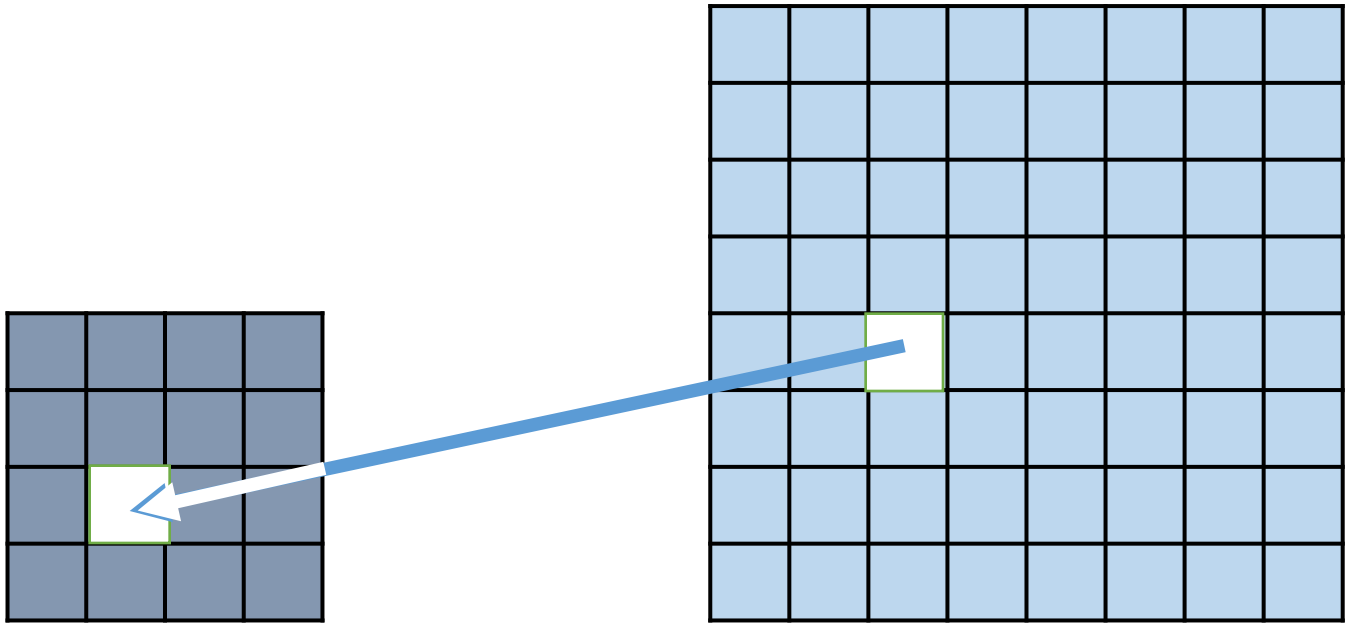
Image Interpolation

Nearest neighbor interpolation

Bilinear interpolation

Bicubic interpolation

Nearest Neighbor Interpolation



4 X 4

8 X 8

Nearest Neighbor Interpolation

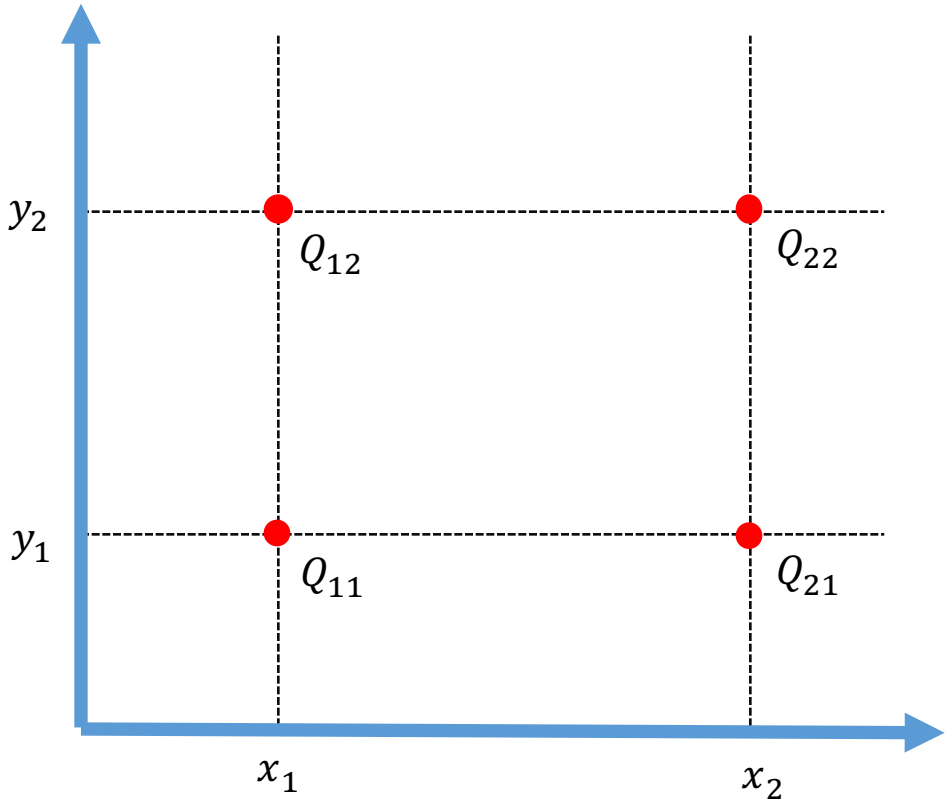
Code

Easy

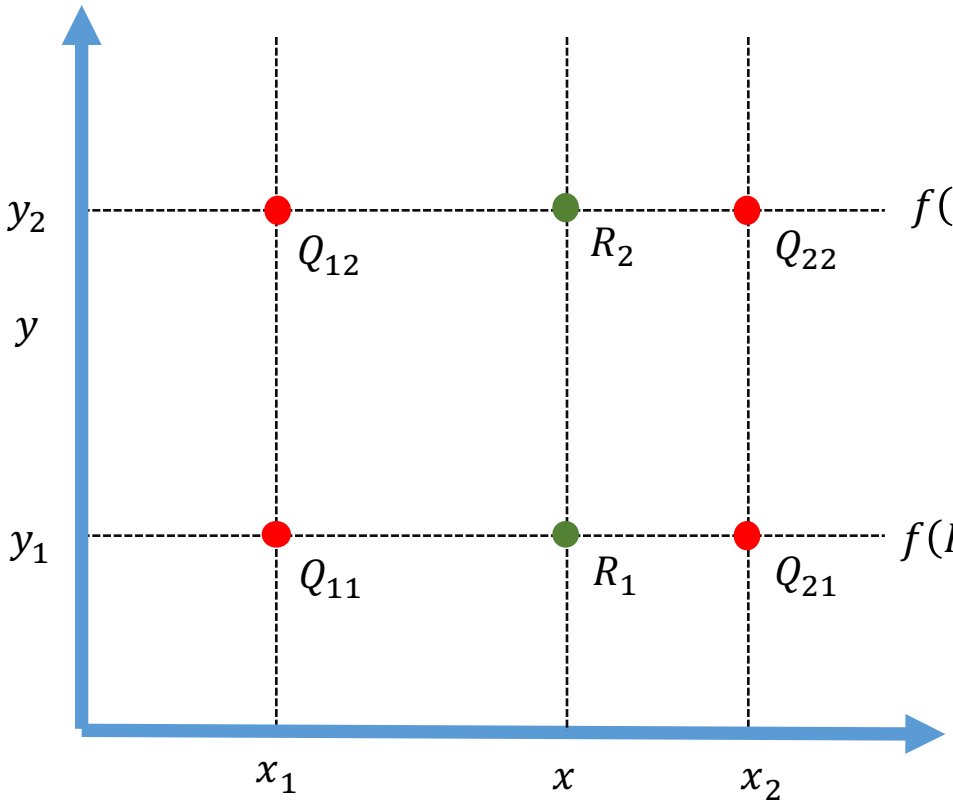
Result

Poor

Bilinear Interpolation



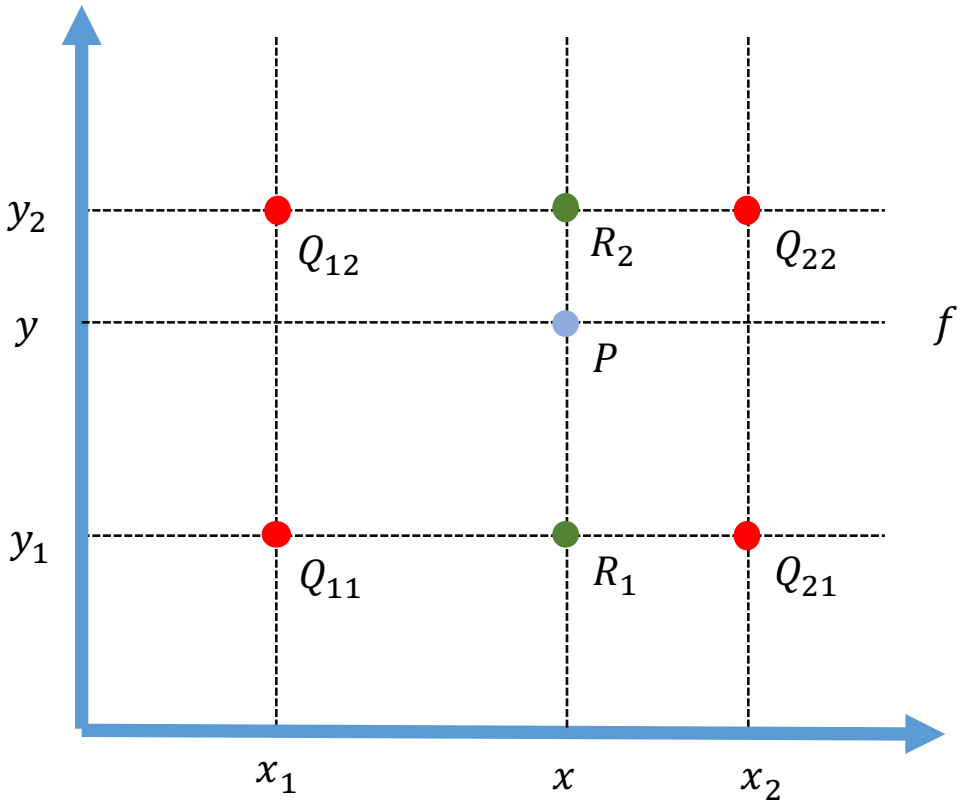
Bilinear Interpolation



$$f(R_2) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22})$$

$$f(R_1) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21})$$

Bilinear Interpolation



$$f(P) \approx \frac{y_2 - y}{y_2 - y_1} f(R_1) + \frac{y - y_1}{y - y_1} f(R_2)$$

Bilinear Interpolation

$$f(x,y) \approx \frac{(x_2-x)(y_2-y)}{(x_2-x_1)(y_2-y_1)} f(Q_{11}) + \frac{(x-x_1)(y_2-y)}{(x_2-x_1)(y_2-y_1)} f(Q_{21}) +$$

$$\frac{(x_2-x)(y-y_1)}{(x_2-x_1)(y_2-y_1)} f(Q_{12}) + \frac{(x-x_1)(y-y_1)}{(x_2-x_1)(y_2-y_1)} f(Q_{22})$$

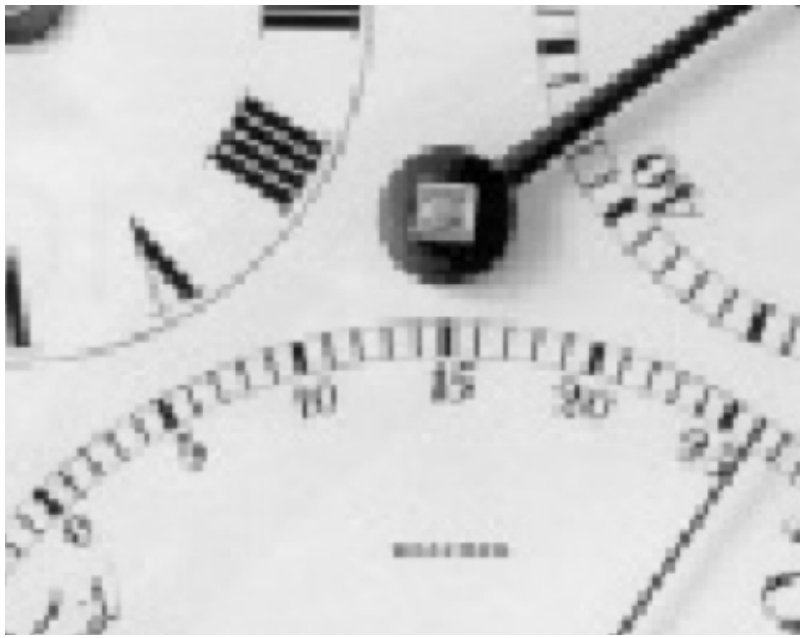
```
sum.val[0]=(y2-y)*(m1.val[0]*(x2-x)+m2.val[0]*(x-x1))+(y-y1)*(m3.val[0]*(x2-x)+m4.val[0]*(x-x1));
sum.val[1]=(y2-y)*(m1.val[1]*(x2-x)+m2.val[1]*(x-x1))+(y-y1)*(m3.val[1]*(x2-x)+m4.val[1]*(x-x1));
sum.val[2]=(y2-y)*(m1.val[2]*(x2-x)+m2.val[2]*(x-x1))+(y-y1)*(m3.val[2]*(x2-x)+m4.val[2]*(x-x1));
newimage.at<Vec3b>(X,Y)[0]=sum.val[0]/((x2-x1)*(y2-y1));
newimage.at<Vec3b>(X,Y)[1]=sum.val[1]/((x2-x1)*(y2-y1));
newimage.at<Vec3b>(X,Y)[2]=sum.val[2]/((x2-x1)*(y2-y1));
```

Contrast

nearest

bilinar

Contrast



nearest

bilinar

Arithmetic Operations

Arithmetic Operations

$$s(x,y) = f(x,y) + g(x,y)$$

$$d(x,y) = f(x,y) - g(x,y)$$

$$p(x,y) = f(x,y) \times g(x,y)$$

$$v(x,y) = f(x,y) \div g(x,y)$$

Arithmetic Operations

$$s(x,y) = f(x,y) + g(x,y)$$

Reduce the noise content

Arithmetic Operations

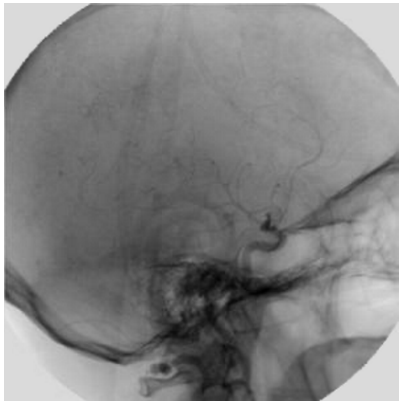
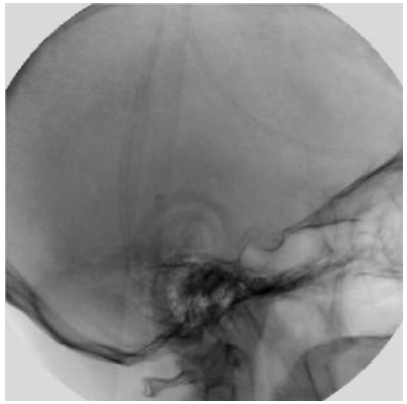
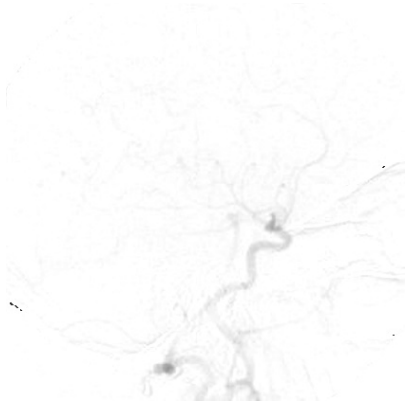
$$\bar{g}(x,y) = \frac{1}{K}\sum_{i=0}^{K} g_i(x,y)$$

Arithmetic Operations

$$d(x, y) = f(x, y) - g(x, y)$$

Enhance differences between images

Arithmetic Operations

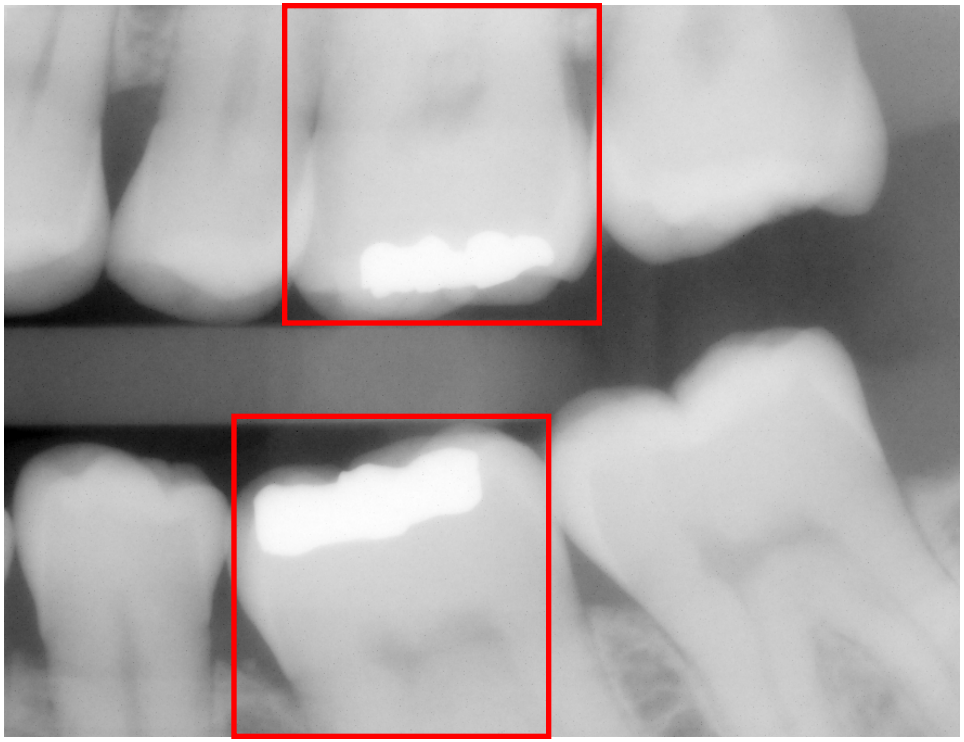




Arithmetic Operations

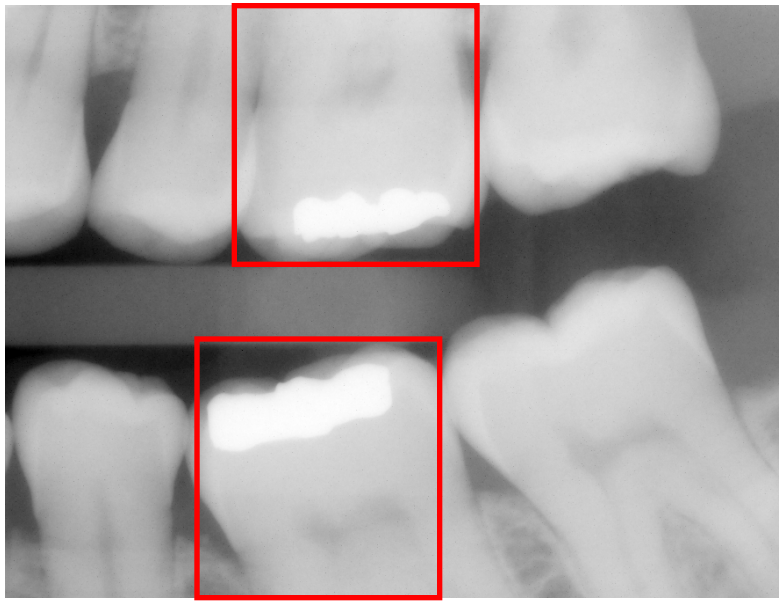
$$p(x,y) = f(x,y) \times g(x,y)$$

Region of interest

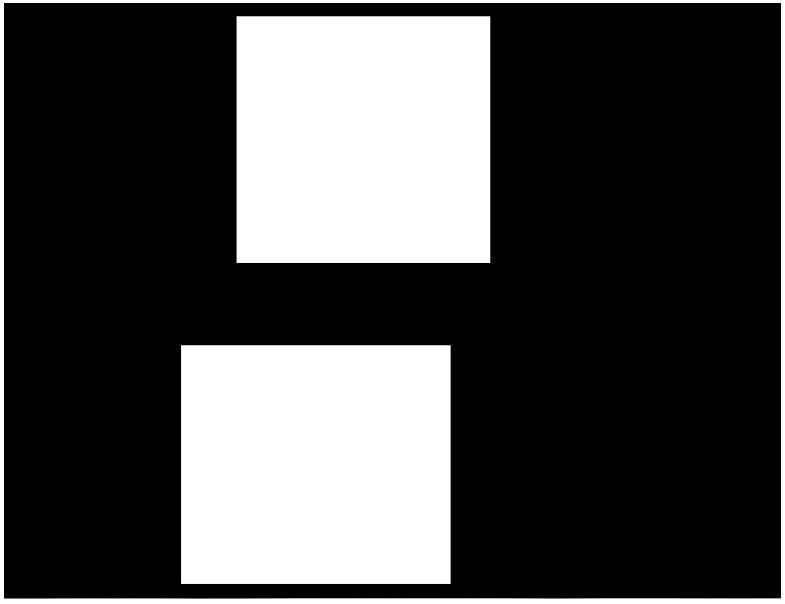
Arithmetic Operations



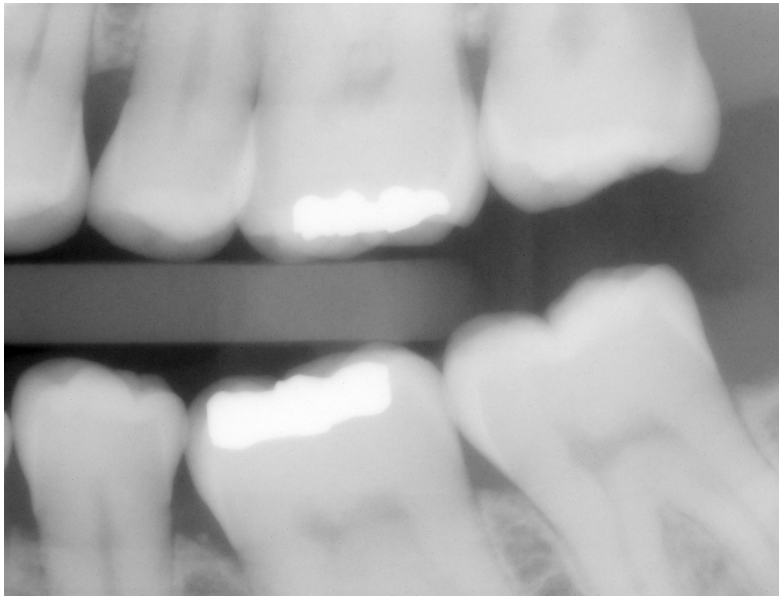
Arithmetic Operations



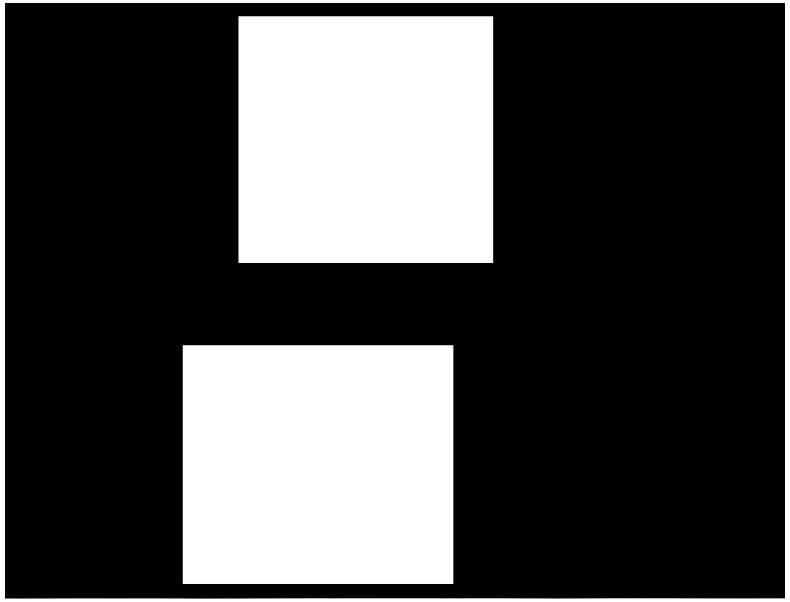
X



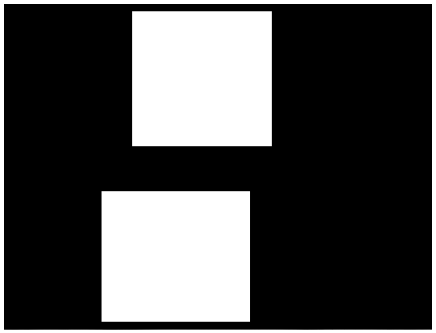
Fault&Experience



X



Arithmetic Operations

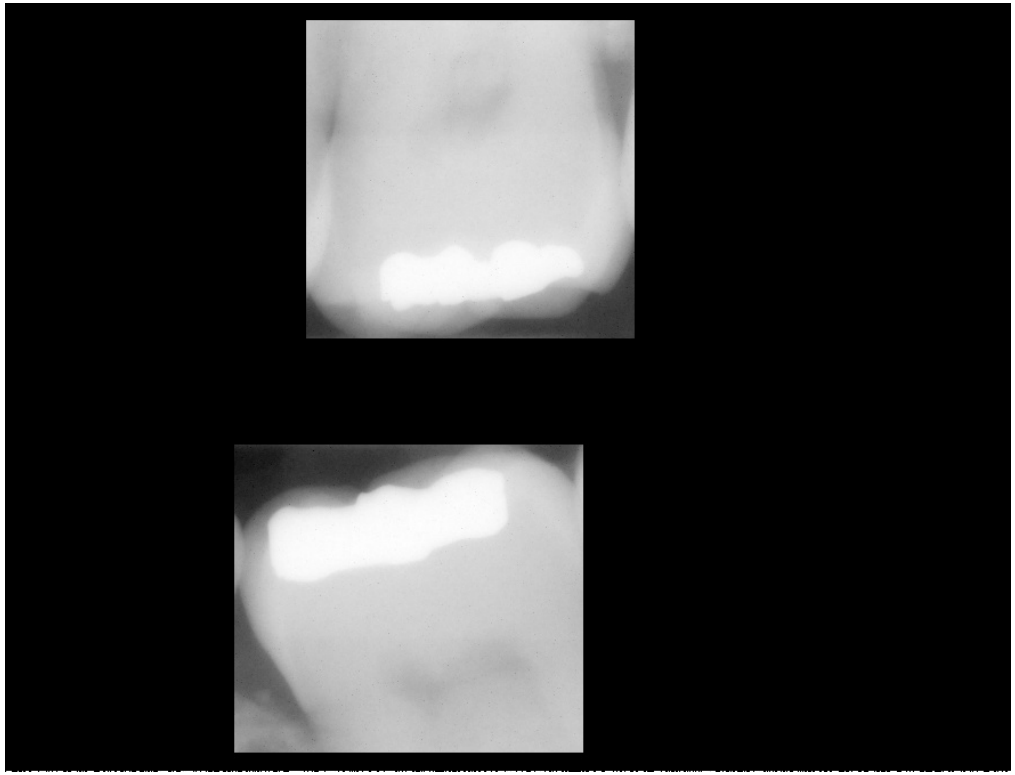


0 ÷255→ 0

255 ÷255→ 1

```
Average.val[0] = shadow.at<Vec3b>(i,j)[0]/255;
Average.val[1] = shadow.at<Vec3b>(i,j)[1]/255;
Average.val[2] = shadow.at<Vec3b>(i,j)[2]/255;
```

Result



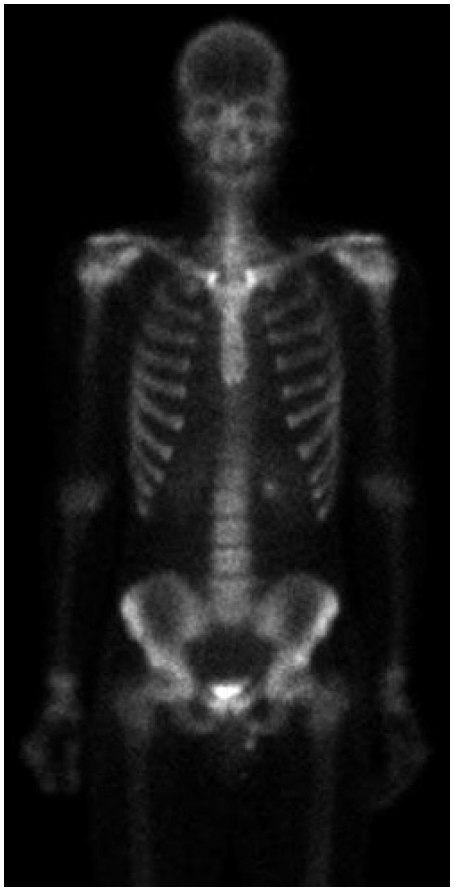
Set Operations

Set Operations

$$A = \{(x, y, z)\}$$

$$A^c = \{(x, y, K - z) | (x, y, z) \in A\}$$

Set Operations



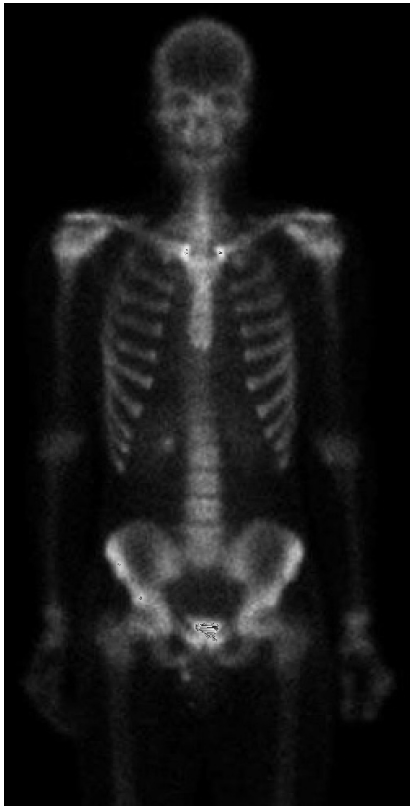
A

Set Operations

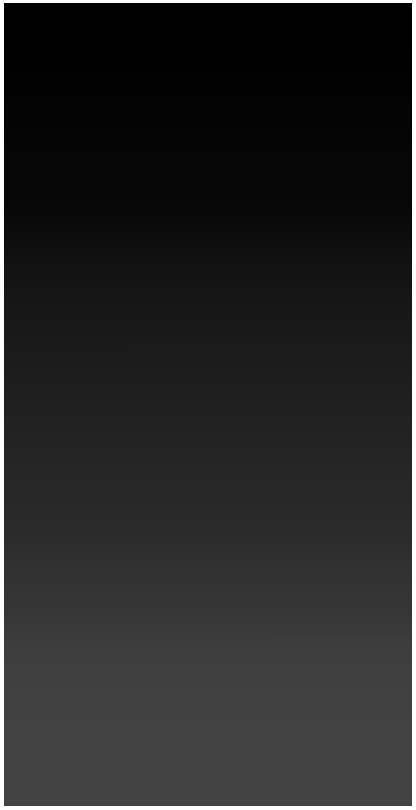
$$A^c = \{(x, y, K - z) | (x, y, z) \in A \}$$

```
dst.at<Vec3b>(i,j)[0] =255-inimage.at<Vec3b>(i,j)[0];
dst.at<Vec3b>(i,j)[1] =255-inimage.at<Vec3b>(i,j)[1];
dst.at<Vec3b>(i,j)[2] =255-inimage.at<Vec3b>(i,j)[2];
```

Set Operations



$$A \xrightarrow{3\overline{\sum f(x,y)}} B$$



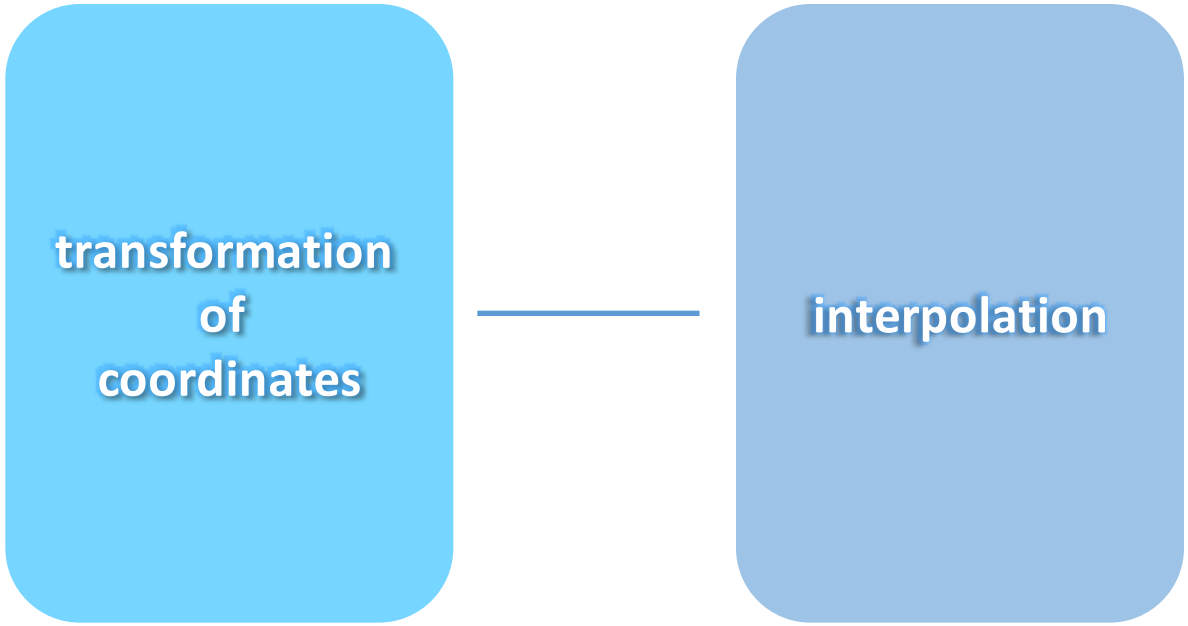
Result

$$A \cup B = \{\max_{z}(a,b) \,|a \in A, b \in B\}$$

```
if (dst.at<Vec3b>(k,l)[0]>sumimage.at<Vec3b>(k,l)[0])
   fin.at<Vec3b>(k,l)[0]=inimage.at<Vec3b>(k,l)[0];
else fin.at<Vec3b>(k,l)[0]=sum.val[0];
if (dst.at<Vec3b>(k,l)[1]>sumimage.at<Vec3b>(k,l)[1])
   fin.at<Vec3b>(k,l)[1]=inimage.at<Vec3b>(k,l)[1];
else fin.at<Vec3b>(k,l)[1]=sum.val[1];
if (dst.at<Vec3b>(k,l)[2]>sumimage.at<Vec3b>(k,l)[2])
   fin.at<Vec3b>(k,l)[2]=inimage.at<Vec3b>(k,l)[2];
else fin.at<Vec3b>(k,l)[2]=sum.val[2];
```

Geometric Spatial Transformations

Geometric Spatial Transformations



Geometric Spatial Transformations

$$(x, y) = T\{(v, w)\}$$

Rotation

$$T=\begin{bmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \longrightarrow \begin{cases} x = v\ cos\theta\ - w\ sin\theta \\ y = v\ sin\theta + w\cos\theta \end{cases}$$

Rotation

Nearest neighbor interpolation

Bilinear interpolation

Bicubic interpolation

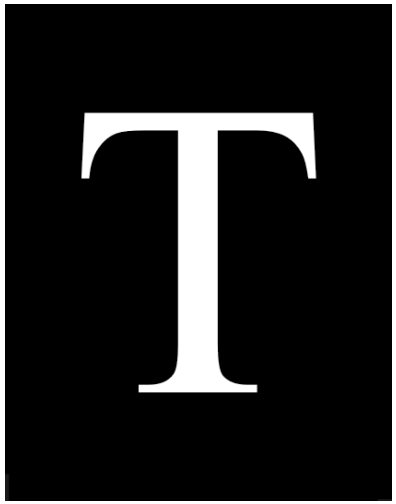
Contrast

```
m1=reimage.at<Vec3b>(x1,y1);
m2=reimage.at<Vec3b>(x2,y1);
m3=reimage.at<Vec3b>(x1,y2);
m4=reimage.at<Vec3b>(x2,y2);
```

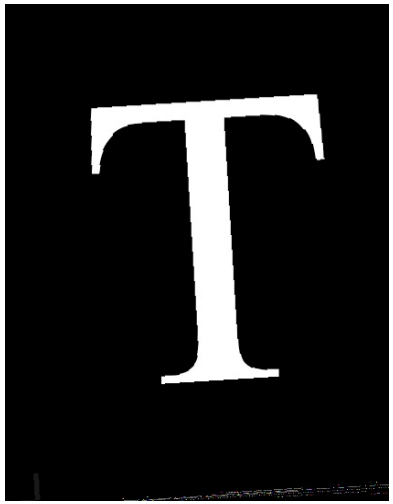
```
m1=inimage.at<Vec3b>((int)(x1*cos(201)-y1*sin(201)),(int)(x1*sin(201)+y1*cos(201)));
m2=inimage.at<Vec3b>((int)(x2*cos(201)-y1*sin(201)),(int)(x2*sin(201)+y1*cos(201)));
m3=inimage.at<Vec3b>((int)(x1*cos(201)-y2*sin(201)),(int)(x1*sin(201)+y2*cos(201)));
m4=inimage.at<Vec3b>((int)(x2*cos(201)-y2*sin(201)),(int)(x2*sin(201)+y2*cos(201)));
```

Bilinear interpolation

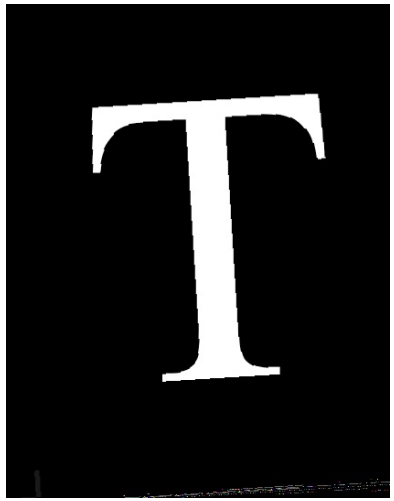
Result



initial

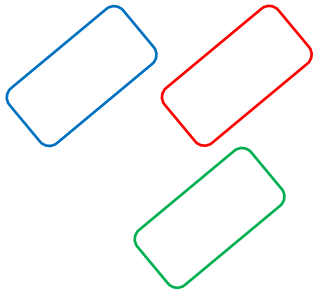


nearest



bilinear

Plan



Thank you