



# 西安交通大学

School of Computer Science and Technology

Name:	MD MAHFUZUR RAHMAN
Student ID:	3123999081
Course Name:	Digital Image Processing
Instructor:	梁毅军
Homework:	<a href="#">01</a>
Submission Date:	24.10.23

### Source Code:

```
byte[,] rotation(byte[,] f,float x0,float y0,float x1,float y1,int  
ow,int oh,float a){
```

```
    int w = f.GetLength(0);  
    int h = f.GetLength(1);
```

```
    double[,] T1 = new double[3,3];  
    T1[0,0] = 1;  T1[1,0] = 0;  T1[2,0] = 0;  
    T1[0,1] = 0;  T1[1,1] = 1;  T1[2,1] = 0;  
    T1[0,2] = -x0; T1[1,2] = -y0; T1[2,2] = 1;
```

```
    double[,] R = new double[3,3];  
    R[0,0] = Cos(a);  R[1,0] = Sin(a);  R[2,0] = 0;  
    R[0,1] = -Sin(a); R[1,1] = Cos(a);  R[2,1] = 0;  
    R[0,2] = 0;  R[1,2] = 0;  R[2,2] = 1;
```

```
    double[,] T2 = new double[3,3];  
    T2[0,0] = 1;  T2[1,0] = 0;  T2[2,0] = 0;  
    T2[0,1] = 0;  T2[1,1] = 1;  T2[2,1] = 0;  
    T2[0,2] = x1;  T2[1,2] = y1;  T2[2,2] = 1;
```

```
    double[,] Temp = multi(T2,R);  
    double[,] T = multi(Temp,T1);
```

```
    double[,] T_inv = new double[3,3];  
    T_inv = invert(T);
```

```
    byte[,] output = new byte[ow,oh];
```

```
    for (int y=0; y<oh; y++)  
        for(int x=0; x<ow; x++){  
            double[,] temp = new double[3,1];  
            temp[0,0] = x; temp[1,0] = y; temp[2,0] = 1;  
            temp = multi(T_inv,temp);  
            double x_pre = temp[0,0];  
            double y_pre = temp[1,0];
```

```

        output[x,y] = bilInter(f,x_pre,y_pre);
    }
    return output;
}

```

```

double[,] multi(double[,] f1,double[,] f2)
{
    int rows1 = f1.GetLength(0);
    int cols1 = f1.GetLength(1);
    int cols2 = f2.GetLength(1);

    double[,] g = new double[rows1,cols2];

    for (int i = 0; i < rows1; i++)
        for (int j = 0; j < cols2; j++)
            for(int k = 0; k < cols1; k++){
                g[i,j] += (double)(f1[i,k]*f2[k,j]);}

    return g;
}

```

```

double[,] invert(double[,] f)
{
    int n = f.GetLength(0);
    double[,] augmentedMatrix = new double[n,2*n];
    for(int i = 0; i < n; i++){
        for(int j = 0; j < n; j++){
            augmentedMatrix[i,j] = f[i,j];
        }
        augmentedMatrix[i,i + n] = 1;
    }

    /*for(int i = 0; i < n; i++){
        for(int j = n; j < 2*n; j++){
            if(i == j-n){
                augmentedMatrix[i,j] = 1;
            }
        }
    }
    */
}

```

```

        }else{
            augmentedMatrix[i,j] = 0;
        }
    }
}*/

for(int i = 0; i<n; i++){

    double pivot = augmentedMatrix[i,i];
    for(int j = 0; j < 2*n; j++){
        augmentedMatrix[i,j] /= pivot;}

    for(int j = 0; j < n; j++){
        if (j != i) {
            double factor = augmentedMatrix[j,i];
            for (int k = 0; k < 2 * n; k++) {
                augmentedMatrix[j,k] -= factor *
augmentedMatrix[i,k];}
            }
        }
    }

    double[,] inverse = new double[n,n];
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) {
            inverse[i,j] = augmentedMatrix[i,j + n];
        }
    }

    return inverse;
}

```

```

byte bilInter(byte[,],double x,double y)
{
    int i = (int)x;
    int j = (int)y;

```

```

int w = f.GetLength(0);
int h = f.GetLength(1);

if (i<0||j<0||i>=w-1||j>=h-1)
    return 0;

double a = x-i;
double b = y-j;

double fxj = f[i,j]*(1-a)+f[i+1,j]*a;
double fxj1 = f[i,j+1]*(1-a)+f[i+1,j+1]*a;

double fxy = fxj*(1-b)+fxj1*b;

return (byte)fxy;
}

void main(){
    byte[,] f = LoadImg();
    byte[,] output = rotation(f, 30,128,128,128,256,256,30);
    ShowImg("f",f);
    ShowImg("Output",output);
}

```

**Input Image:**



**Output Image:**



## Using Different value:

