

作业5

1. 写出逆滤波和维纳滤波图象恢复的具体步骤。

• 逆滤波：

- $g(x,y)$ 傅里叶变换： $G(u,v)$, $h(x,y)$ 傅里叶变换： $H(u,v)$;
- 设计恢复转移函数: $M(u,v)$

$$M(u,v) = \begin{cases} 1/H(u,v) & \text{if } u^2 + v^2 < w_0^2 \\ 1 & \text{else} \end{cases}$$

或

$$M(u,v) = \begin{cases} k & \text{if } H(u,v) < d \\ 1/H(u,v) & \text{else} \end{cases}$$

- $\hat{F}(u,v) = G(u,v) * M(u,v)$
- $\hat{F}(u,v)$ 傅里叶逆变换得到 $\hat{f}(u,v)$

• 维纳滤波：

- $g(x,y)$ 傅里叶变换： $G(u,v)$, $h(x,y)$ 傅里叶变换： $H(u,v)$;
- 设计恢复转移函数: $M(u,v)$

$$H_w(u,v) = \frac{H(u,v)^*}{|H(u,v)|^2 + K}$$

- $\hat{F}(u,v) = G(u,v) * M(u,v)$
- $\hat{F}(u,v)$ 傅里叶逆变换得到 $\hat{f}(u,v)$

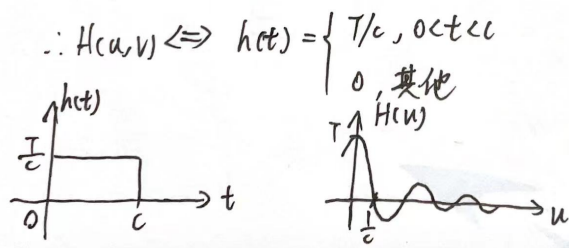
2. 推导水平匀速直线运动模糊的点扩展函数的数学公式并画出曲线。

设水平匀速直线运动为: $x = \frac{c}{T}t$, $y(t) = 0$

$$\therefore H(u, v) = \int_0^T e^{-j2\pi [ux(t) + vy(t)]} dt$$

$$\therefore H(u, v) = \int_0^T e^{-j2\pi uct/T} dt = \frac{T}{\pi uc} \sin(\pi uc) e^{-j\pi uc}$$

$$\therefore f(t) = \begin{cases} T/c, & |t| < \frac{c}{2} \\ 0, & |t| > \frac{c}{2} \end{cases} \Leftrightarrow T \frac{\sin(\pi uc)}{\pi uc}$$



3. 编程实现lema.bmp的任意角旋转

代码:

```
1 def rotate_image(image_path, angle):
2     img = Image.open(image_path).convert('L')
3
4     # 计算图像的中心点
5     width, height = img.size
6     xc = width / 2
7     yc = height / 2
8
9     # 将角度转换为弧度
10    angle_rad = math.radians(angle)
11    cos_theta = math.cos(angle_rad)
12    sin_theta = math.sin(angle_rad)
13
14    # 创建一个新的图像, 尺寸足够大以容纳旋转后的图像
15    new_width = int(width * abs(cos_theta) + height * abs(sin_theta))
16    new_height = int(width * abs(sin_theta) + height * abs(cos_theta))
17    rotated_img = Image.new('L', (new_width, new_height), color='black')
18
19    # 计算新图像的中心点
20    new_xc = new_width / 2
21    new_yc = new_height / 2
22
23    # 对每个像素进行旋转
24    for x in range(new_width):
25        for y in range(new_height):
26            # 将坐标转换为相对于中心点的坐标
27            x_rel = x - new_xc
28            y_rel = y - new_yc
29
30            # 应用旋转公式
```

```

31         x_rot = int(x_rel * cos_theta - y_rel * sin_theta + xc + 0.5)
32         y_rot = int(x_rel * sin_theta + y_rel * cos_theta + yc + 0.5)
33
34         # 检查旋转后的坐标是否在原图像范围内
35         if 0 <= x_rot < width and 0 <= y_rot < height:
36             rotated_img.putpixel((x, y), img.getpixel((x_rot, y_rot)))
37
38     return rotated_img
39
40 # 调用函数进行图像旋转
41 rotated_image = rotate_image("D:\\lab\\VSCodeProject\\dip\\imagedata\\lena.bmp", 45)
42
43 # 保存旋转后的图像
44 rotated_image.save("D:\\lab\\VSCodeProject\\dip\\hw5\\image_rotated2.bmp")
45

```

结果：

