

HW2

翁正朗 PB22000246 DIP

1

马赫带和同时对比度都反映了人类视觉系统在处理边缘和对比度时的一种共同现象：**边缘增强**。

- **马赫带**：在亮度渐变的区域附近，人眼会感知到比实际更明显的明暗边界。这种现象是由于视觉系统在边缘处增强对比度，以帮助我们更清晰地识别物体的轮廓。
- **同时对比度**：一个颜色或亮度区域会受到其邻近区域的影响，使得同一颜色或亮度在不同背景下看起来不同。这种现象也是视觉系统通过对比度增强来强调边界和差异。

2. 视觉系统的非线性处理：

- 视觉系统并不是简单地记录光强，而是通过复杂的神经处理来增强对比度和边缘，以提高视觉清晰度和物体识别能力。

3. 背景影响：

- 在这两种现象中，背景或相邻区域对中心区域的感知有显著影响，突出显示了人类视觉系统的相对性。

这些现象揭示了视觉系统如何通过增强边缘和对比来优化信息处理，以便在复杂环境中更有效地识别和理解视觉信息。

2

1. 赫尔曼栅格错觉：

- 在黑色方块的交叉点处，人们会看到灰色的斑点。

2. 佩奇错觉：

- 两条相同长度的线段，因箭头方向不同而显得一长一短。

3. 埃姆斯房间：

- 一个特制的房间，人在里面看起来会有不同的大小。

4. 纳克尔立方体：

- 一个二维图形，看起来像是一个立方体，可以在不同的方向上翻转。

3

欧氏距离

| | 0 | 1 | 2 | 3 | 4 |
|---|-------------|------------|---|------------|-------------|
| 0 | $2\sqrt{2}$ | $\sqrt{5}$ | 2 | $\sqrt{5}$ | $2\sqrt{2}$ |
| 1 | $\sqrt{5}$ | $\sqrt{2}$ | 1 | $\sqrt{2}$ | $\sqrt{5}$ |
| 2 | 2 | 1 | 0 | 1 | 2 |
| 3 | $\sqrt{5}$ | $\sqrt{2}$ | 1 | $\sqrt{2}$ | $\sqrt{5}$ |
| 4 | $2\sqrt{2}$ | $\sqrt{5}$ | 2 | $\sqrt{5}$ | $2\sqrt{2}$ |

街区距离（曼哈顿距离）

| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | 4 | 3 | 2 | 3 | 4 |
| 1 | 3 | 2 | 1 | 2 | 3 |
| 2 | 2 | 1 | 0 | 1 | 2 |
| 3 | 3 | 2 | 1 | 2 | 3 |
| 4 | 4 | 3 | 2 | 3 | 4 |

棋盘距离（切比雪夫距离）

| | 0 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|---|
| 0 | 2 | 2 | 2 | 2 | 2 |
| 1 | 2 | 1 | 1 | 1 | 2 |
| 2 | 2 | 1 | 0 | 1 | 2 |
| 3 | 2 | 1 | 1 | 1 | 2 |
| 4 | 2 | 2 | 2 | 2 | 2 |

4

```

1  import cv2
2  import numpy as np
3
4  # 读取灰度图像
5  image = cv2.imread('lena.bmp', cv2.IMREAD_GRAYSCALE)
6
7  # 检查图像是否成功加载
8  if image is None:
9      print("图像加载失败")
10 else:
11     # 定义区域的左上角坐标和大小
12     x, y = 200, 200
13     width, height = 10, 10
14
15     # 读取指定区域的像素值

```

```

16     region = image[y:y+height, x:x+width]
17
18     # 打印像素值
19     print("10x10 区域的像素值: ")
20     print(region)

```

```

1  [[ 78  72  82  78  74  73  71  67  64  74]
2   [ 69  68  80  91  87  84  69  68  77  98]
3   [ 78  83  84  83  64  70  77  87  87  98]
4   [ 86  72  74  70  65  65  97  90  78  72]
5   [ 77  67  60  62  66  86  99  90  73  59]
6   [ 66  58  64  70  81  91  90  79  60  53]
7   [ 63  74  70  88  98  94  94  85  62  59]
8   [ 74  83  83 103  93 105 102  66  58  51]
9   [ 83  95  90  97  96 104  81  69  54  51]
10  [ 78  90 100  99 100  90  64  67  57  54]]

```

5

```

1  import cv2
2  import numpy as np
3
4  # 读取灰度图像
5  image = cv2.imread('lena.bmp', cv2.IMREAD_GRAYSCALE)
6
7  # 检查图像是否成功加载
8  if image is None:
9      print("图像加载失败")
10 else:
11     # 将前256行的像素值设为255
12     image[:256, :] = 255
13
14     # 打印处理后的图像（显示图像）
15     cv2.imshow('Processed Image', image)
16     cv2.waitKey(0)
17     cv2.destroyAllWindows()
18     cv2.imwrite('half0_lena.bmp', image)
19

```



6

```

1  import cv2
2
3  def modify_bmp_palette(file_path):
4      with open(file_path, 'rb+') as f:
5          # BMP 文件头大小为 54 字节, 调色板从第 54 字节开始
6          f.seek(54)
7
8          # 修改调色板, 假设 BMP 是 8 位灰度图像
9          for i in range(256):
10             # 生成彩色映像表, 例如: 蓝色渐变
11             red = i
12             green = 128
13             blue = 255 - i
14             f.write(bytes([blue, green, red, 0])) # 写入 BGR0 格式
15
16     # 修改 BMP 文件
17     modify_bmp_palette('lena.bmp')
18
19     # 使用 OpenCV 打开并显示修改后的图像
20     image = cv2.imread('lena.bmp', cv2.IMREAD_UNCHANGED)
21     cv2.imshow('Modified Image', image)
22     cv2.waitKey(0)
23     cv2.destroyAllWindows()
24     cv2.imwrite('colored_lena.bmp', image)

```

