

# Mcahine Vision HW3 Report

資工三 110590004 林奕廷

## Dependencies

```
python = ">=3.9,<4"
opencv-python = "^4.9.0.80"
alive-progress = "^3.1.5"
matplotlib = "^3.8.3" # optional, for visualing the histogram
```

## Run

```
python 110590004_hw3.py
```

## Question 1

### Grayscaleing and Binarization

- Use  $(0.3 \times R) + (0.59 \times G) + (0.11 \times B)$  to convert the RGB image to grayscale image.
- Implement Triangle algorithm to binarize the grayscale image.
  - Apply smoothing in the histogram to get a better threshold.
  - If the matplotlib is installed, the histogram and threshold will be shown in **debug** folder.
  - Thresholds: img\_1: 237, img\_2: 242, img\_3: 241, img\_4: 234.

## Part 1

### N-Connected Distance Transform

- Implement the 4-connected and 8-connected distance transform.
- Using the following formula to calculate the distance:

$$f^0[i, j] = f[i, j]$$

$$f^m[i, j] = f^0[i, j] + \min(f^{m-1}[u, v])$$

where  $(u, v)$  is n-neighbors of  $(i, j)$

## Part 2

### Skeletonization

1. Start with the smallest number  $h = 1$  in the distance transform.
2. Iteratively remove the points with height  $h$  that are not the local maximum of 4-neighbors.
3. If removing the point would leads to a connectivity lose, then keep the point.
4. Increase the height  $h$  and repeat the process until the height is larger than the maximum height in the distance transform.
5. Use the structure element to do thinning on the skeleton. The structure element is defined as:

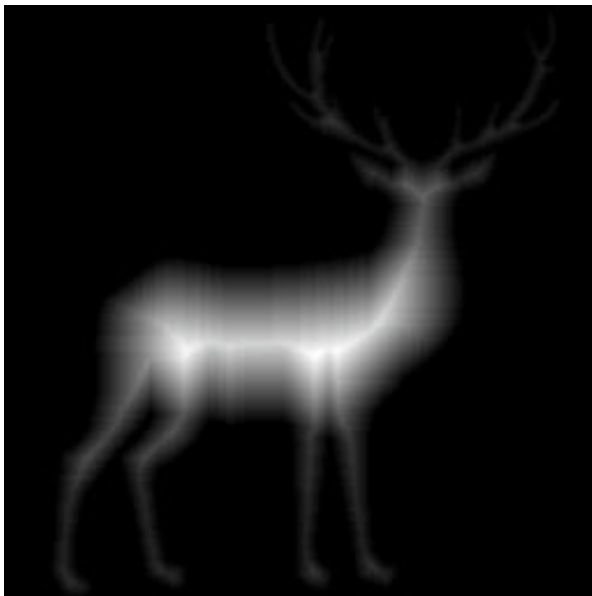
0	0	0
	1	
1	1	1

	0	0
1	1	0
	1	

- At each iteration, the image is first thinned by the left hand structuring element, and then by the right hand one, and then with the remaining six  $90^\circ$  rotations of the two elements.

## Result

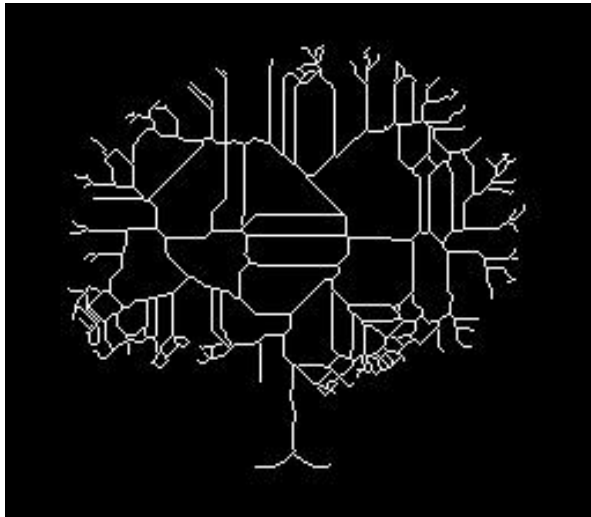
### 4-Connected Distance Transform



## 8-Connected Distance Transform



## Skeletonization



Machine  
Vision

