

Lecture 6 - IP - notes

Jannick Drews

October 29, 2018

1 Pre-lecture

Preparation: Read section 5.2.2 and 5.3 (Edge detection)

2 Lecture

3 Sobel

Based on First-order derivative

- Noise reduction
Median/Mean filter
- Edge enhancement
Calculate candidates for the edges
- Edge localization
Decide which edge candidates to keep

$$g_x \approx f(x+1, y) - f(x-1, y)$$

$$g_y \approx f(x, y+1) - f(x, y-1)$$

Gradient vector: $\vec{g} = [g_x, g_y]$

Magnitude: $g_m = \sqrt{g_x^2 + g_y^2} \approx |g_x| + |g_y|$

4 Canny

Based on groups of edges

- Noise reduction
2d Gaussian used for smoothing
- Edge enhancement
Magnitude of gradient vector
- Edge localization
Thinning edges using non-maximal suppression;
No sobel effect, just a single-pixel edge found in the input-image. (Finds direction of the gradient; $\tan^{-1}(\frac{g_y}{g_x})$, then if the neighbouring pixels go in same direction, the one with the smallest magnitude is suppressed(0), we're trying to find a local maxima on the direction of the gradient.)

We end up with the thresholding dilemma again, this can be fixed with Hysteresis Thresholding.

5 Hysteresis Thresholding

Check slides

(Basically matches the thresholding with the upper-threshold and sees if there's any pixels alike to support the edge, else suppress it.) Conclusion:

Pros:

One pixel wide edges

Edges are grouped together (often good for segmentation)

Robust against noise!

Cons:

Complicated to understand and implement

Slow

Used a lot!

6 Important notes

(Pre-processing)

$Edge \rightarrow ROI \rightarrow Feature \rightarrow fill \rightarrow segment$

Edges:

Shi-Tomasi method

Harris algorithm

Scale-space corner detectors(Old SoTA)

Term: **Keypoint**, refer to corners and regions, a point in the image where it doesn't change no matter how much you edit.

Getting magnitude and orientation (HOG), SIFT, SURF (basic ones)

Feature correspondence, ascertaining which parts of one image correspond to which parts of another image. (Homography mapping (MACHINE-LEARNING)).

7 Important

Make a edge detection algorithm

8 Mini-project

It's all about making your own function.