low-level vision

Image Processing

Convolution

- use convolution to filter out some kind of signals.
  - moving average filter: low-pass filter
  - Gaussian filter: low-pass filter
    - when σ is close to $+\infty$ all high-frequence parts are filtered, and the signal becomes a constant.
    - when σ is close to 0, no filtering at all

Edge detectors

Definition of an edge

- An edge is defined as a region in the image where there is a “significant” change in the pixel intensity values (or having high contrast) along one direction in the image, and almost no changes in the pixel intensity values (or low contrast) along its orthogonal direction.

Criteria for edge detection

- TP: True Positive, FP: False Positive, FN: False Negative, TN: True Negative
- Precision = TP / (TP+FP). Precision measures how many assertations are really useful(or truth) among all those answers proposed by the algorithm.
- Recall = TP / (TP+FN). Recall measures how many true targets are found among all targets waiting to be identified.
- Good localization: minimize the distance between the detected edge and the ground truth edge

Method

1. Compute image gradient to find edge, however noises can be overwhelming -> use Gaussian filter to smooth the image. In practice, convolve with the derivative of a Guassian filter.
2. Non-maximal suppression: remove all pixels that are not the local maximum in the gradient direction. In practice
   - check two pixels along the gradient direction, if their gradient(use bilinear interpolation) is greater than the current pixel, then it is a local maximum.
   - simply, just compare the pixel value with its neighboring pixels, namely pixels in the 3x3 window, now interpolation is not needed.
3. Hysteresis thresholding: apply a threshold on the gradient magnitude to eliminate weak edges.
   - define maxVal = 0.3 x avg magnitude of the pixels that pass NMS
   - define minVal = 0.1 x avg magnitude of the pixels that pass NMS
   - pixels with gradient magnitude greater than maxVal are reserved
   - pixels with gradient magnitude less than minVal are discarded
4. Edge Linking
   - If the current pixel is not an edge, check the next one.
   - If it is an edge, check the two pixels in the direction of the edge (ie, perpendicular to the gradient direction). If either of them (or both)
     - have the direction in the same bin as the central pixel
     - gradient magnitude is greater than minVal
     - they are the maximum compared to their neighbors (NMS for these pixels),
       then you can mark these pixels as an edge pixel
   - Loop until there are no changes in the image once the image stops changing, you've got your canny edges! That's it! You're done!

Corner detectors

Definition of a corner

The key property of a corner: In the region around a corner, image gradient has two or more dominant directions

Method

1. Compute the image derivatives in the x and y directions. ($I_x,\ I_y$)
2. Compute the square of the derivatives. (I_x^2, I_y^2, I_xI_y)
3. Apply the window function. (g(I_x^2), g(I_y^2), g(I_xI_y))
   - Rectangle window
     - translation-invariant
     - rotation-variant
   - Gaussian filter
     - translation-invariant
     - rotation-invariant(in theory, in practice, rotation-variant due to discretization of the image plot)
4. Corner response function: $\theta = g(I_x^2) + g(I_y^2) - g(I_xI_y)^2 - \alpha[g(I_x^2) + g(I_y^2)]$
5. Non-maximum suppression

invariant and equivariant

$X \in V$, $f : V \to V$ is a function, $T : V \to V$ is a transformation operation.

- f is equivariant: $T[f(x)] = f(T[x])$
  - eg. corner detection
- f is invariant: $f(x) = f(T[x])$
  - eg. classification

Line fitting

Least Square Method

- Data: $(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)$
- Line equation: $y_i - mx_i - b = 0$
- Find (m, b) to minimize $E = \sum_{i=1}^{n}(y_i - mx_i - b)^2$
- The best fit line is given by $y = mx + b$
- In practice, define $E = ||Y - XB||^2$, where B is the vector of (m, b), and we try to minimize E
  - $B = (X^TX)^{-1}X^TY$
  - Limitation: Fails completely for vertical lines

SVD

- Line equation: $ax + by = d$
- Energy landscape: $E(a, b) = \sum_{i=1}^{n}(ax_i + by_i - d)^2$
- In practice, we try to minimize $||Ah||^2$, where $h = (a, b, d)$
  - we will fall to trivial solution $h = (0, 0, 0)$ if no constraint is given. So we constrain $||h|| = 1$.
  - So we do SVD on A, $A = UDV$, and try to place h in the most squashed direction, which is the direction of the smallest singular value.
  - h is the last column of V. (Consider the sigularity values are places in a descending order from top to bottom).
- Robust to small noises but sensitive to outliers.

RANSAC

Sample some points to give a condidate line, and examine how many points agree with this line(inlier). Choose the one with the most inliers.

1. Randomly select k points from the data set. (k=2 for a line)
2. Compute an estimation of the transformation.(Line in this case)
3. Choose a threshold ϵ. Any point lies within ϵ distance from the estimated line is considered an inlier.
4. Retain the one with the most inliers.
   - if two candidates have the same number of inliers, choose the one with the smallest energy. (Square error in this case)
5. Recalculate the transformation using all the inliers.
6. The recalculation may change the inliers, so repeat the process until convergence(inliers do not change anymore).

However, RANSAC cannot help for problem with high rate of outliers.