Notes
Computer Vision

Basil R. Yap

2018 January

1 Week 1

Gaussian Filter: $G(x,y) = \frac{1}{2\pi\sigma^2}e^{-\frac{x^2+y^2}{2\sigma^2}}$

Edge Detection:

- Prewitt Operator: [[1,1,1]]
- Sobel operator: [[1,2,1]]

2 Week 2

histogram equalization: increase contrast

1. compute cdf at k (normalized)
2. multiply by L-1 (L, number of levels; 256), then floor
3. this is new intensity

KNN Distances:

1. L2 Distance (Euclidean distance): $\sqrt{\sum(I_1 - I_2)^2}$
2. L1 Distance (Manhattan distance): $\sum|I_1 - I_2|$

L1 is diamond shaped, L2 is circular

KNN computationally expensive and unable to differentiate position and intensity shift.

3 Week 3

for given image len x width:

- input x = D x 1 vector (flatten image)
- weight W = k x D array
- bias b = k x 1 vector

- score s = k x 1 vector

s = Wx + b

s = Wx where W=[W,b], x=[x,1]

- x=(D+1) x 1 vector
- W=k x (D+1) array
- s=k x 1 vector

Softmax classifier
probability of class m can be calculated with $softmax(f) = \frac{e^{f_m}}{\sum_{j=1}^{K} e^{f_j}}$
cross-entropy loss for ith training sample: $L_i = -\log \frac{e^{f_{yi}}}{\sum_{j=1}^{K} e^{fi}}$
assumption of cnn:

- locality of pixel dependencies
- stationary of image statistics
- translation invariance
- same filter for whole image