Name: John Stephen Gutam

CS 682 Computer Vision Exam

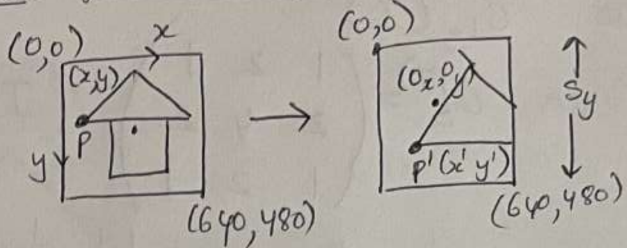
G Number: G01413212

4/5/2024

Netid: jgutam

1. Coordinate transformation between pixels of the same point:

Let 'P' (x,y) be the point on the image 1 and $P'(x' y')$ be the transformed point on the image 2 in 2D imaging system.



Original Image

Transformed image.

Image size (original) = $(640, 480)$
$x \quad y$

The new origin in the transformed image $= (O_x, O_y) = \left(\frac{640}{2}, \frac{480}{2}\right) = (320, 240)$

- During upsampling, each pixel in cropped image is upsampled to 2 x 2 block of pixels in the original image.
- Therefore pixel coordinates given by

$$x' = x_s + O_x \qquad \text{where } x_s = S_x x \quad \text{and } (O_x, O_y) = (320, 240)$$

$$y' = y_s + O_y \qquad y_s = S_y y$$

In the matrix form we can represent as

$$x' = \begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = K \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

The 'k' translation can be given as $\underbrace{\begin{bmatrix} S_x & S_\theta & O_x \\ 0 & S_y & O_y \\ 0 & 0 & 1 \end{bmatrix}}_{K}$ where $S_\theta \approx 0$

2. Let the input image be 'I'.

By applying the Gaussian filter on the image, ~~it~~ let represent as $G_s(I)$

Given $G_s(I) = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} (I)$

Now apply the horizontal derivative

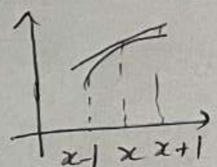
horizontal derivative

$$f_x = \frac{\partial}{\partial x}(G_s)(I) = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} \begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}$$

$$f_x = \frac{\partial}{\partial x}(G_s)(I) = \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} = M$$

Applying the derivative means convolving with $\begin{pmatrix} -1 \\ 0 \\ 1 \end{pmatrix}$.

$\rightarrow \frac{df(x)}{dx} = \lim \frac{f(x+\Delta x) - f(x)}{\Delta x}$

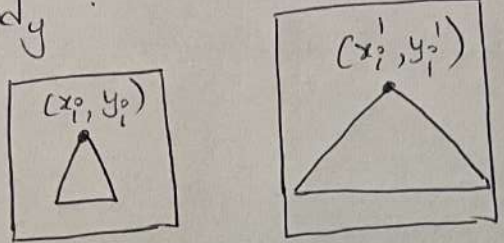


-1	0	1

The above matrix 'M' is similar to the sobel filter S_x^T where $S_x = \begin{pmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{pmatrix}$

Therefore taking the horizontal derivative I_x of the Gaussian-filter image, can be approximated to sobel filter.

3. The given two images undergone a scaling and translation transformation.
Parameterized by translation $[t_x, t_y]$ and scaling s_x and s_y.



(a) The equation relating two image coordinates $[x_i, y_i]^T$ and $[x_i', y_i']^T$ are

$$\begin{bmatrix} x_i' \\ y_i' \end{bmatrix} = \underbrace{\underbrace{\begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix}} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \underbrace{\begin{bmatrix} t_x \\ t_y \end{bmatrix}}}_{4 \text{ unknown parameter}}$$

(b) We need minimum of 2 points as we have 4 unknown parameters.

(c) Linear constraint in the form of $Ax = b$ for 4 unknown parameters

$$\begin{bmatrix} x_i' \\ y_i' \end{bmatrix} = \begin{bmatrix} s_x & 0 \\ 0 & s_y \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} t_x \\ t_y \end{bmatrix} \Rightarrow \underbrace{\begin{bmatrix} & \cdots & & \\ x_i & y_i & 0 & 0 & 1 & 0 \\ 0 & 0 & x_i & y_i & 0 & 1 \\ & \cdots & & \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} s_x \\ s_y \\ t_x \\ t_y \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} \cdots \\ x_i' \\ y_i' \\ \cdots \end{bmatrix}}_{b}$$

So the above equation is represented $Ax = b$.

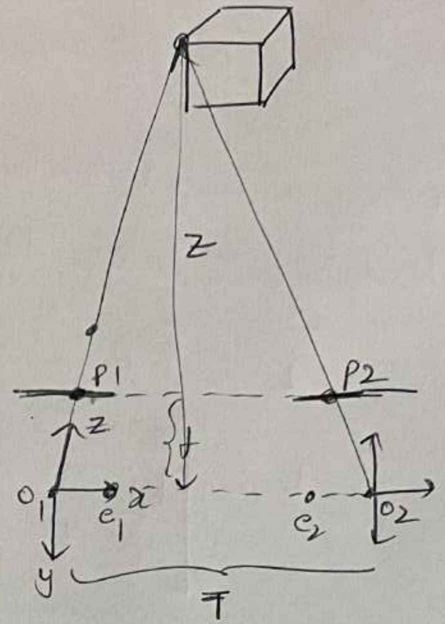
(d) The main idea behind RANSAC is to iteratively select random samples, estimate the model parameters, and evaluate the model's fitness to the data.

Pseudo code:

1. Randomly select minimal subset of points
2. Hypothesis a model
3. compute error function
4. Select points consistent with model
5. Repeat hypothesis and verify loop.

(e)

4. (a) Given optical axis of the camera is z,
y axis pointing down and x-axis is pointing to the right.



As the camera moves in
x-z plane, y = 0

$$\hat{T} = \begin{bmatrix} 0 & -t_z & t_y \\ t_z & 0 & -t_x \\ -t_y & t_x & 0 \end{bmatrix}$$

$$\Rightarrow \hat{T} = \begin{bmatrix} 0 & -t_z & 0 \\ t_z & 0 & -t_x \\ 0 & t_x & 0 \end{bmatrix} \longrightarrow ① \qquad \boxed{t_y = 0}$$

$$\hat{T}x_1 = T \times X_1$$

$$\Rightarrow x_2^T \underbrace{\hat{T} R}_{E} x_1 = 0$$

$$\boxed{E = \hat{T} R} \longrightarrow ③$$

Also given the camera is rotated around y-axis which is yaw

$$R_y(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \longrightarrow ②$$

By combining ① and ②, ③ Equations, Essential matrix is given by

$$E = \hat{T} R = \begin{bmatrix} 0 & -t_z & 0 \\ t_z & 0 & -t_x \\ 0 & t_x & 0 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}$$

(b) The minimal number of corresponding points needed ~~8~~ 4.
usually it takes 8 points using 8 point algorithm but as the camera moving along the x-z plane and can rotate along y-axis.

(c) planar essential matrix decomposed to rotation and translation is given by

$$E = \hat{T} R$$

$$E = \underbrace{\begin{bmatrix} 0 & -t_z & 0 \\ t_z & 0 & -t_x \\ 0 & t_x & 0 \end{bmatrix}}_{\substack{\text{Camera moved} \\ \text{along } xz \text{ plane.}}} \underbrace{\begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}}_{\substack{\text{Rotation around} \\ y\text{-axis}}}$$

5.(a) Depth of field refers to the range of distance that appears acceptably sharp. It varies depending on camera type, aperture and focusing distance.

The concept of focus relates to the sharpness or clarity of objects within the depth of field.

How to control the depth of field?

→ changing the aperature size affects depth of field.

- A smaller aperture increases the range in which the object is approximately in focus.
- But small aperature reduces amount of light - need to increase exposure.

→ Varying the aperture

- Large aperture = small DoF
- small aperture = large DoF

Focusing tradeoffs

- A smaller aperture increases the range in which the object is approximately in focus, the image is getting darker and sharper - need long exposure time.
- Large aperture, shallow depth of field, small exposure time.

(b) Gaussian smoothing

A particular case of averaging

- The coefficients are samples of a 1D Gaussian.

- Give more weight at the central pixel and less weight to the neighbors.
- The further away the neighbors, the smaller the weight.

$$g(x) = \frac{1}{\sqrt{2\pi}\,\sigma} e^{-x^2/2\sigma^2}$$

standard deviation σ : determines extent of smoothing.

- Gaussian filters remove 'high-frequency' components from the image. (low pass filter)
- Gaussian filter reduces salt-and-pepper noise

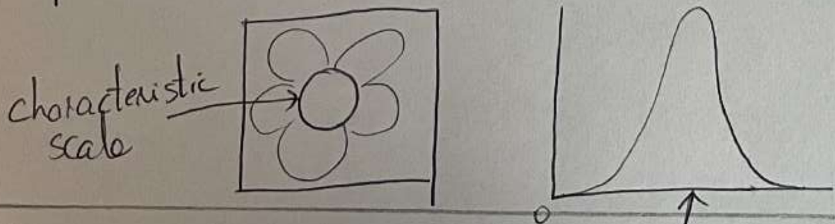
Median Filter

- Replace each pixel with the Median value of all the pixels in the neighborhood.
- Median filter does not spread the noise
- Can remove spike noise
- Expensive to run.

Gaussian smoothing can be used when preserving overall image structure is acceptable. Median filtering can be used when removing impulse noise and preserving sharp edges.

(c) Characteristic scale of image location:

The characteristic scale of a blob is defined as the scale that produces peak of Laplacian response in the blob ~~&~~ center



characteric scale of image location can be computed by scale-space and scale selection techniques.

(d) Non-maximum suppression technique is used to retain only the most prominent and non-overlapping detections while suppressing or eliminating redundant detections.

↳ If the corner response value is not the maximum among its neighbour, suppress the detection by discarding it. otherwise, retain the detection as a valid corner.

↳ To eliminate redundant detections group the same corners.

(e) Seperable filters:

↳ whose response can be computed as a sequence of two or more 1D convolution operations.

- Factors into product of two 1D Gaussians

Sepearability of the Gaussian Filter

$$G_\sigma(x,y) = \frac{1}{2\pi\sigma^2} \exp^{-x^2+y^2/2\sigma^2}$$

$$= \left(\frac{1}{\sqrt{2\pi}\sigma} \exp^{-x^2/2\sigma^2}\right)\left(\frac{1}{\sqrt{2\pi}\sigma} \exp^{-y^2/2\sigma^2}\right)$$

- 2D Gaussian can be expressed as the product of two functions, one a function of x and the other a function y.
- In this case the two functions are the identical 1D.

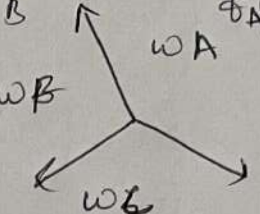
why is seperability improve the computational efficiency?

- The complexity of filtering an $n \times n$ image with an $m \times m$ kernel is $O(n^2 m^2)$
- what if the kernel is seperable?

$$O(n^2 2m)$$

6. (a) Given a multi class logistic regression for the classes A, B, C with the weights

$\phi_B(x) = w_B^t x + b$ $\qquad$ $\phi_A(x) = w_A^t x + b$



$$\begin{cases} w_A = [1, -4, 7]^T \\ w_B = [2, -3, 6]^T \\ w_C = [7, 9, -2]^T \end{cases} \qquad \phi(x) = [-2, 1, 3] \qquad \phi_C(x) = w_C^t x + b$$

sigmoid function $\sigma(w^t x) = \dfrac{1}{1 - e^{(-w^t x)}}$

$P_w(y_i = A / x_i) = \sigma(w^t x + b)$

$P_w(y_i = \overset{\&C}{B} / x_i) = 1 - \sigma(w^t x + b)$

- Apply softmax function to convert these scores to probabilities:

$$\text{Softmax}(\phi_A, \phi_B, \phi_C) = \left(\frac{e^{\phi_A}}{\sum_{\text{A}} e^{\phi_j}}, \frac{e^{\phi_B}}{\sum_j e^{\phi_j}}, \frac{e^{\phi_C}}{\sum_j e^{\phi_j}} \right)$$

$\boxed{\phi_A = \underline{\underline{w^t x}}}$

$\phi_A = 1 * (-2) - 4 * 1 + 7 * 3 = 15$

$\phi_B = 2 \times (-2) + (-3)(1) + (6) \times 3 = 11$

$\phi_C = 7 \times (-2) + 9(1) + (-2)(3) = 11$

$\phi_A = \underline{\underline{15}}$, which means it belongs A classification.

(b) But given that the new point is classified into B label.

Given $\alpha = 1$

$w_i = w_j + \alpha f(x_i)$ (correct)

$w_j = w_j - \alpha f(x_i)$ (incorrect)

For the example $f(x) = [-2, 1, 3]$ weights for w_A and w_B are updated as they wrongly classified this point.

$$w_A = \begin{bmatrix} 1 \\ -4 \\ 7 \end{bmatrix} + (1)\begin{bmatrix} -2 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} -1 \\ -3 \\ 10 \end{bmatrix}$$ $\boxed{w_i = w_j + \alpha f(x_i) \text{ is used}}$

$$w_B = \begin{bmatrix} 2 \\ -3 \\ 6 \end{bmatrix} + (1)\begin{bmatrix} -2 \\ 1 \\ 3 \end{bmatrix} = \begin{bmatrix} 0 \\ -2 \\ 9 \end{bmatrix}$$ $\boxed{w_j = w_j - \alpha f(x_i) \text{ is used}}$

$$w_C = \begin{bmatrix} 7 \\ 9 \\ -2 \end{bmatrix}$$ It remains same as it is not disturbed or classified with new point.

(d) Advantages of support vector machine

1. Effective in high dimensional space:
2. Robust to overfitting
3. Flexibility through kernel methods.

7. (a) Technique I will implement is

1. convert the image to grayscale
2. Apply the blob ~~detect~~ segmentation technique to separate the cells from the background
3. Feature like dark dots/cells are extracted
4. cell counting.

(b) To process the video data from a camera perched above a belt.

1. Capture image convert to grayscale
2. will apply the ML model which has been trained with vegetable detection model.
3. Through the object detection and recognition technique will classify the object.
4. Based on the respectic price of each ~~pro~~ object, the total price is calculated and bill can be generated to the customer.