Georgia Institute of Technology
George W. Woodruff School of Mechanical Engineering
ME6406 Machine Vision (Fall 2021)
Assignment #4: (ME6406A Due **Tuesday Dec. 7th 2021, 23:59pm EDT**)

1. Pose Estimation [1]

A calibration block with one circular pocket on its top face is shown in Figure 1(a). The *object coordinate system* $X_oY_oZ_o$ is defined at one of its lower corners and the axes are aligned with the three orthogonal edges of the block as shown in Fig 1(b). An image of the component is captured by a camera located at an unknown position and unknown orientation. The image is similar to Fig. 1(b)

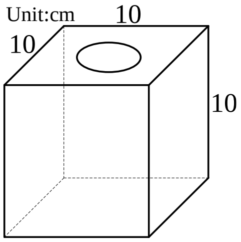

Fig 1(a)

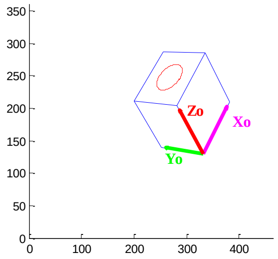

Fig 1(b)

f [cm]	dx' [cm]	dy' [cm]	C_x [pixel]	C_y [pixel]	N_{cx}	N_{cy}	N_{cz}
0.79	6×10^{-4}	6×10^{-4}	376	240	–0.4226	–0.6943	–0.5826

a. Use Hough Transform to identify the 9 straight edges in '*block.png*'. You may use MATLAB built-in HT algorithm. Find the 9 normal vectors $\mathbf{n}=(af,bf,c)^T/|(af,bf,c)^T|$ as defined in [1]. (Note: for equation of line $au+bv+c=0$ as defined in [1] on 2D image plane, (u,v) are the physical coordinates measured from the center of the image plane, hence if you obtain coefficients associated with pixel coordinates, make sure you do the appropriate conversion to get the real 'a', 'b' and 'c')

b. From circle ellipse correspondence, the normal vector from the circle center is found in $[N_{cx}\ N_{cy}\ N_{cz}]^T$. Along with the results obtained from 1a, carry out pose estimation that utilizes *line correspondence* and *orthogonal constraint*. Show value of [**R**] and **T**.

Suggested functions: hough.m, houghpeaks.m, houghlines.m. Nonimal value of T_z is 120 cm

2. Artifical Neural Network (ANN) based on back-propagation (BP) learning

Design a BP-ANN to recognize O and T on a binary 4×4 square grid.

a. Define your ANN structure by specifying the number of inputs, hidden layers, layer nodes, and outputs. Provide a schematic of your chosen ANN structure.

b. Derive the weight update rule for your ANN, assuming a unipolar sigmoid function for each processing element.

c. Train with the data in Fig 2(a). Show the convergence curve (mean squared error vs. number of epoch). Save the weights of nodes in a ".mat" file.

d. Test the data in Fig 2(b) by reading the weights of nodes in your ".mat" file. Show the output values and results.

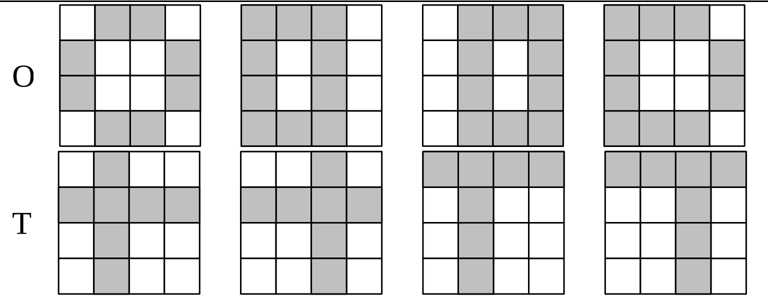

Fig 2(a). Training data

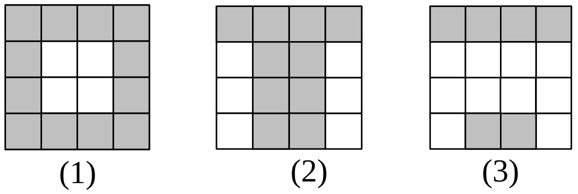

Fig 2(b). Test data

3. Color

I. Artificial Color Contrast (ACC): Consider two color samples [225, 88, 96], [149, 135, 134] respectively representing target and noise in Fig 3, which are to be spatially separated in color space using DoG as discussed in class so that classification can be more easily performed:

$$h_i(x,y) = G_{\sigma c} * f_j(x,y) - G_{\sigma s} * f_k(x,y)$$

where $f_j(x,y)$ with (j=1, 2, 3) corresponds to *RGB* component images respectively; and $f_k(x,y)$ are some linear combinations of *RGB* component images to be designed.

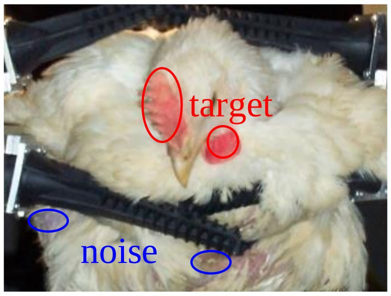

Fig 3. Chicken

a. Derive the following equations with $f_k(x,y)$ equal to $+(R-G)$ and $+(R+G-B)$:

$$\begin{array}{ll} h_1(x,y) = DoG * R + G_{\sigma s} * G & h_a(x,y) = DoG * R + G_{\sigma s} * [B-G] \\ h_2(x,y) = DoG * G + G_{\sigma s} * [2G-R] & h_b(x,y) = DoG * G + G_{\sigma s} * [B-R] \\ h_3(x,y) = DoG * B + G_{\sigma s} * [B-(R-G)] & h_c(x,y) = DoG * B + G_{\sigma s} * [2B-(R+G)] \end{array}$$

b. Perform the ACC transformation (σ_c=1, σ_s=10) on sample color patterns (100×100 each) with the following combinations: 1-2-3, 1-2-c, 1-b-3, 1-b-c, a-2-3, a-2-c, a-b-3, a-b-c

II. Color-based Image Segmentation: Color is important information. Same objects commonly have their domain color. L-a-b color system is the color-opponent space with the L lightness dimension and a-b color-opponent dimensions. The color-based image segmentation can be performed by applying the clustering method on the points in a-b domain with the post process. Follow the steps to segment the target of the RGB image '*Chicken.jpg*'.

Step 1. Transfer pixels from RGB to Lab color system.

Step 2. Apply k-means clustering on data in a-b domain with cluster number (k=3).

Step 3. Erode the segment image to filter out small fragments.

III. Principle component analysis (PCA): Use the RGB image '*Chicken.jpg*' for the following.

a. Determine the covariance matrix of data.

b. Derive the components (eigenvectors) with eigenvalues arranged in a descending order.

c. Obtain the maximum and minimum values of three component matrices. Show these three matrices (images) with linear mapping from the minimum and maximum values to the range of (0-255).

Reference:

[1] Qiang Ji, Mauro Costa, Robert Haralick, and Linda Shapiro, "An Integrated Linear Technique for Pose Estimation from Different Features," *International Journal of Pattern Recognition and Artificial Intelligence*, Vol. 13, No. 5, 1999

[2] Simpson, P. (1992). Foundations of neural networks, Chapter 1 in *Artificial Neural Networks*, Lau, C. & Sanchez-Sinencio, E. (Eds.), pp. 2-13, IEEE Press, New York, NY

Other handouts posted in Canvas: 00_ANN2.pdf and 00_ANN03-Example.pdf