

In this assignment, you'll perform handwritten character classification.

The underlying data are drawn from the CEDAR dataset which was prepared for the U.S. Postal service using images of actual handwritten addresses from envelopes. More information about this dataset is available from <http://www.cedar.buffalo.edu/Databases/CDROM1>. I have extracted images from this dataset and placed them in several files available to you, namely,

- [http://www.cise.ufl.edu/~jnw/CAP6615sp11/Resources/train\\_upper.7z](http://www.cise.ufl.edu/~jnw/CAP6615sp11/Resources/train_upper.7z) which is a 7zip file containing 9,119 PBM images of uppercase handwritten letters in a directory called train\_upper
- [http://www.cise.ufl.edu/~jnw/CAP6615sp11/Resources/test\\_upper.7z](http://www.cise.ufl.edu/~jnw/CAP6615sp11/Resources/test_upper.7z) which contains 1,042 PBM images of uppercase handwritten letters in directory test\_upper

As the names of these files may suggest, you are to use the characters in train\_upper and test on the characters in test\_upper. More specific details will be provided below.

To make your job easier, you don't have to determine what features to use for this task if you don't want to. I have already calculated a number of features that are available in the following files:

- <http://www.cise.ufl.edu/~jnw/CAP6615sp11/Resources/TrainFeatures.mat>
- <http://www.cise.ufl.edu/~jnw/CAP6615sp11/Resources/TestFeatures.mat>

The features we are using are *edge histogram detector* (EHD) features. They are computed on the images using the following Matlab functions that I have made available to you:

- <http://www.cise.ufl.edu/~jnw/CAP6615sp11/Resources/HackEHDCharFeats.m>
- <http://www.cise.ufl.edu/~jnw/CAP6615sp11/Resources/ehdcharfeats.m>

As explained within the code, the features are based on edge masks associated with each pixel. The edge masks characterize directional edges in the character image as being either horizontal, diagonal, vertical, or antidiagonal as shown in the following Figure:

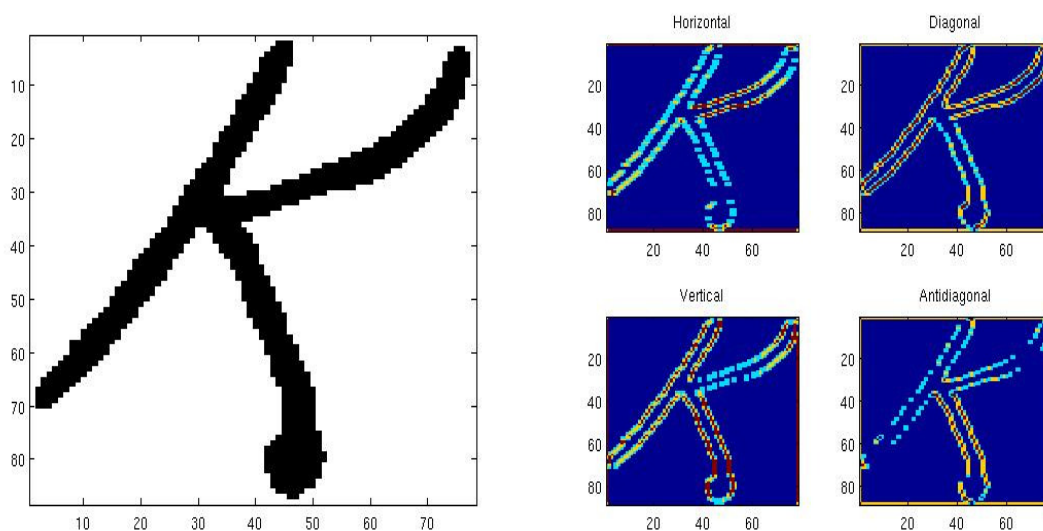


Figure 1. Character K and its corresponding edge mask images.

The character image sizes differ, however, each character images is divided into 16 regions labeled A through P as shown in Figure 2. Each set of 2X2 neighboring regions (such as {A,B,E,F} for examples) correspond to a single subimage of the original image. Thus, there are nine subimages. Each subimage

overlaps any of its neighboring subimages by  $\frac{1}{2}$ . Thus, for example, {A,B,E,F} and {B,C,F,G} both share regions B and F.

A	B	C	D
E	F	G	H
I	J	K	L
M	N	O	P

Figure 2. Regions from which subimages are constructed.

Each pixel is assigned at most one of these directions (or no direction at all) by first finding the direction with the maximal edge strength. If the edge strength of the direction with maximal strength exceeds a fixed threshold, it is assigned that direction. Otherwise, no direction is assigned.

Within a subimage, a normalized histogram of the directions assigned to the pixels in that subimage. Thus, there are nine (9) regions time four (4) directions, or 36 different features computed for a single image.

Each of the files TrainFeatures.mat and TestFeatures.mat contains three variables, X (an N-by-36 matrix of EHD features), d (an N-by-1 column vector of desired response, i.e., characters), and fnlist (an N-by-1 cell array of the file names associated with each image).

#### Your Task

1. Implement a classifier using some technique you've learned this semester, that map a character image's EHD features (or an image itself if you want to use other features) into its character label (a letter). Your classifier will be trained on images in train\_upper and tested on images in test\_upper.
2. You must implement and test this on either or both of the following sets of letters:
  - a. {M,N,O,P,Q,R,U,V,W}
  - b. All 26 letters in the Roman alphabet
3. You must investigate the impact of reducing the feature space using PCA. Consider testing numbers of features chosen on the sum of the largest eigenvalues (e.g., .9, .95, .99). Instead or in addition to using PCA, you can investigate the use of kernel PCA with some reasonable choices of numbers of eigenvectors to consider.
4. Write a report explaining what you did and what you conclude. You must produce a confusion matrix for your classifier and discuss what it shows.