

**Project 3**  
**CSC 9010 - Computer Vision**  
**Grading: 150 points**  
**Due Date: April. 9, 2015**

---

**Description:** In this project, we will be reinforcing the concepts of image classification in Matlab using different classification methods that we learned in class. The application that we will apply our techniques on is the in the area of OCR or optical character recognition. To add some challenge to the program, you will be given several CAPTCHAs (an acronym for “Completely Automated Public Turing test to tell Computers and Humans Apart”) and asked to classify the numbers in the image.

**Part 1 - Segment out the Numbers (30 points)**

For the first part of this project, be given several CAPTCHAs and will need to isolate each character. In order to perform this task, you should look at the morphological operations provided in Matlab. Specifically, you can utilize the bwmorph operations and look at erosion and dilation. The bwlabel method can give you the connected components of the image. For example, given the CAPTCHAs in Figure 1, you can isolate the characters as seen in Figure 2.

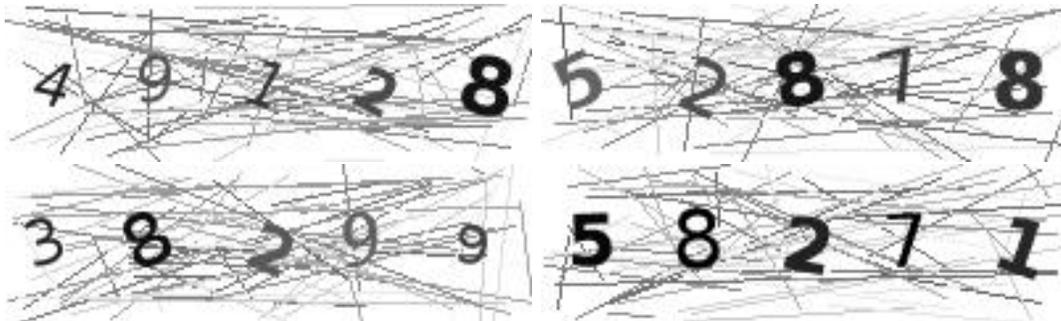


Figure 1: CSC 9010 Computer Vision - Test CAPTCHAs for OCR

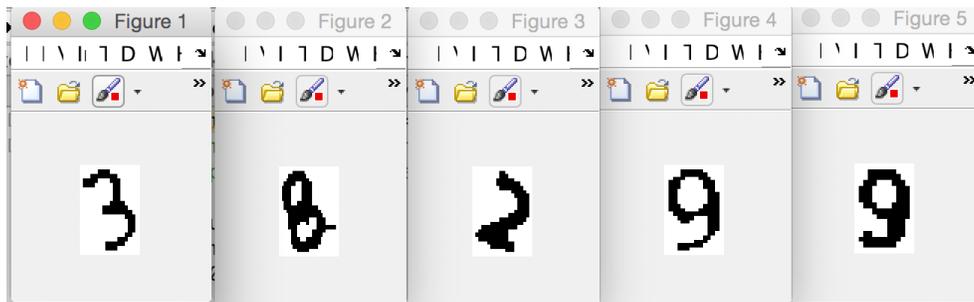


Figure 2: CSC 9010 Computer Vision - Segmented CAPTCHA using bwmorph operations

**Part 2 - Build a Training Set (30 points)** You will need to build a training set of characters that you can feed into a classification method in order to account for the possible variations that you might see in the CAPTCHAs. I provide 10 images of basic digits that you can transform

in different ways to obtain the training set. See Figure 3. If you see the CAPTCHAs they have multiple rotations and line widths. To create a training set, rotate, dilate, and erode the training images in order to get a good training sample. For example, I created 15 variations of each digit with different rotations (-10, -5, 0, 5, 10) with an erosion, no morph, and dilation for each case.



Figure 3: CSC 9010 Computer Vision - Training set of characters.

**Part 3 - KNN classification (30 points)** For each segmented image, perform a KNN classification for  $k=1,5,9$  using Euclidian distance in the pixel space to classify each character. Put the results in your PDF write up.

**Part 4 - SVM classification (30 points)** Use the “fitsvm” method to construct the SVMStruct object that can classify each character. Use the “predict” method with the SVMStruct to classify each character as a digit. Remember that there is no multi-class SVM built into Matlab and thus you will need to create multiple SVMStructs, one for each digit. Put your results in the PDF write up.

**Part 5 - Random Forest classification (30 points)** Use the “TreeBagger” method with 100 trees to create an ensemble classifier. Then you can again use the “predict” method with the random forest training object to perform a multi-class digit segmentation. See Figure 4 for an example result.

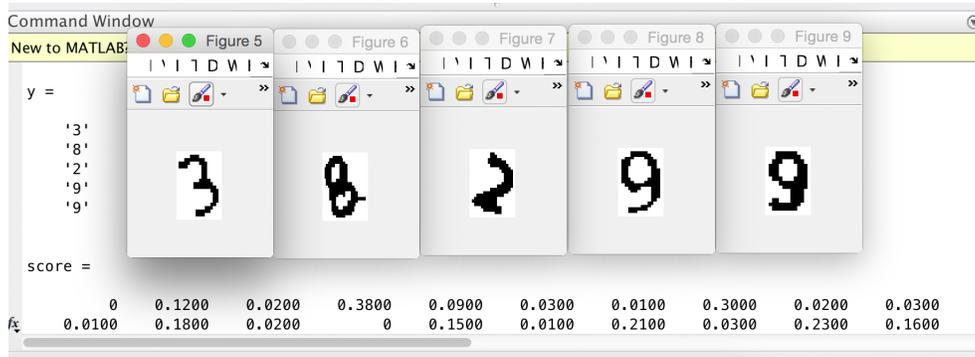


Figure 4: CSC 9010 Computer Vision - Result of a Random Forest classification on one CAPTCHA

**Part 6 - PDF write up (30 points)** Create a 3-5 page write up that summarizes the interesting parts of your program and includes results. Include any problems or insights that you encountered.

**Deliverables:** Submit on Blackboard.