CS 320 Computer Vision Homework #2
Due November 9, 2009

You may work with a partner on this homework; submit one copy of your results and source code with both names.

Use techniques that we have learned to determine how much money is shown in an image. You may assume that the only "money" consists of US pennies, nickels, dimes and quarters. Each image is taken under fluorescent lights on a white sheet of paper. Every image includes the same pencil, for scale. (The first image shows the pencil next to a ruler, but has no coins in the image – it does not count). Images are available at the website http://cs.hiram.edu/~walkerel/cs320/ColorCoins .

You may NOT assume that every object in the image is a coin.

While there are many potential techniques you can use, here are some starting points. Feel free to experiment with the interactive tools in ImageJ before writing your code.

1. Determine a threshold to separate the coins from the background. Since coins are different colors, you might want to try multiple thresholds, or different thresholds for different colors. Then, isolate the individual coins with a connected components algorithm in the binarized image.
2. Use a convolution kernel of your choice to detect edges in the image. Use a Hough transform to find likely circles (or ovals) in the image.

Whether you choose option 1 (region-based) or 2 (edge-based) or a combination of both, you should generate a modified image with the "objects" highlighted in some way. You could do this by outlining them or highlighting them with pixels of a contrasting color so it looks like the image has been drawn on. (Image A)

Once you have found your objects, use features of each object (e.g. geometric features, color features) to recognize the pencil and/or coins. Your next output should indicate which objects are coins. (Image B)

Finally, use a combination of at least two features to classify your coins as pennies, nickels, dimes and quarters. Your final output should mark the coins in the image by their types, and should also output the total amount of money in the image. (Image C as well as a text result).

Test your system on 10 images and report the results of the tests (false positives, false negatives, correct classifications, correct rejections). Write a paragraph evaluating the performance of your system and explaining your results. Include (at least) the three image results noted above for the 10 test images. (Of the 10 images used for testing at the end, no more than 5 should have been used for testing while developing your algorithms. If you have time, you are encouraged to test with more than 10 images).

Your final submission should be a complete report of how your system was developed, as well as documented source code for an ImageJ plugin that starts with an image, creates the three images noted A, B, and C above, and outputs a number (the value of the coins in the image, excluding any non-coins).

Please include data to justify any "magic numbers" in your code, and a careful formal evaluation of your system for part 5.

Each team will also give a short presentation of their system and results in class on November 9 or November 11.