Assignment A2: Pixels

CS 4640
Fall 2021

Assigned: 7 September 2021

Due: 16 September 2021

For this problem, submit through Canvas the A2 report PDF as well as any required Matlab functions or data stuctures used to address the problems.

Some notes:

- No scripts

- Functions must have CS4640 prefix

- Indent headers correctly (5 spaces indented lines) and put required info

- Do not exceed 72 characters per source line

1. Produce a 141x8 defect array, *defects*, where each column corresponds to a specific defect as enumerated in the Student Project 1 description (StudentProjectCola.pdf). The eighth column indicates that there was no bottle in the middle of the image. Save it as 'defects.mat' and submit through Canvas.

2. Develop a program called *CS4640_inspect_Coke* with header given below which inspects all the images in a directory and for each image applies all available defect detection functions to return a report on the defects of each Coke bottle image. This function should call the cap defect and no bottle defect functions defined in problems 4 and 5.

```
function defects = CS4640_inspect_Coke(dir_name)
```

```
% CS4640_inspect_Coke - determine defects by inspecting Coke bottle image
% On input:
%      dir_name (string): name (or path) of directory
% On output:
%      defects (nx8 array): defects matrix; n is the number of images
%      in the directory; the 7 columns represent specific defects:
%       col 1: bottle under-filled or empty
%       col 2: bottle over-filled
%       col 3: bottle has missing label
%       col 4: bottle has white label
%       col 5: bottle label is not straight
%       col 6: cap is missing
%       col 7: bottle is deformed
%       col 8: no middle bottle
% Call:
%      defects = CS4640_inspect_Coke('bottle_images');
% Author:
%      <Your name>
%      UU
%      Fall 2021
%
```

3. Reproduce Figure 3.6 from the text using the function:

$$gt(\sigma) = \frac{255ln(1 + (e^\sigma - 1)g)}{ln(1 + (e^\sigma - 1)255)}$$

where $g \in \{0, 1, \ldots, 255\}$, $\sigma \in \{0.0001, 0.1, 1, 2, 5\}$. Use the Matlab *legend* function to show which plot corresponds to which σ. Compute this log transform on the cameraman image, and show the original image and the transformed image with $\sigma = 1$. Discuss what specific details in the image are made more obvious in the transformed image.

4. Use color histogramming (Matlb function *CS6640_hist_color*) to identity the bottle caps. First create a template color histogram, ch_cap, using an 5x5 subwindow from image001, and set the number of bins to 4. Save this template to a file called ch_cap.mat. Write a function, *CS4640_cap* with header given below, which uses the template as follows:

- At every pixel in the test image, compute the color histogram of an 5x5 subwindow centered at each possible pixel (i.e., where an 5x5 fits).

- Compute the KL divergence (*CS6640_KL_div*) of the template histogram and the subwindow histogram.

- Threshold the divergence to label red pixels.

Discuss issues you dealt with in developing this, as well as ideas for improvement. How can a reasonable threshold be determined for this? Report the performance results on the bottle dataset (i.e., correctness on the 141 images).

```
function b = CS4640_cap(im,ch_cap,num_bins,m,n,thresh)
% CS4640_cap - locate bottle cap in inspection image
% On input:
%      im (MxNx3 array): color image
%      ch_cap (num_bins^3 x 1 vector): color histogram template
%      num_bins (int): number of bins for each color in color histogram
%      m (int): number of rows in subwindow
%      n (number of columns in subwindow
%      thresh (float): threshold for KL divergence match
% On output:
%      b (Boolean): 1 if cap; else 0
% Author:
%      <Your name>
%      UU
%      Fall 2021
%
```

5. Develop a Matlab function, *CS4640_no_bottle* with header below that uses histogramming to determine if there is no bottle present in the middle of the test image. Explain your method and any issues with it. Report the performance on the inspection dataset.

```
function b = CS4640_no_bottle(im)
% CS4640_no_bottle - determine if no bottle im middle of image
% On input:
%      im (MxNx3 array): input image
% On output:
%      b (Boolean): 1 if no bottle in middle; else 0
% Call:
%      b = CS4640_no_bottle(bot1);
```

```
% Author:
%     <Your name>
%     UU
%      Fall 2021
%
```