Signals and Systems

Dr. Saeed Akhavan

Computer Assignment 2

Deadline: 16th of Azar at 17:00

University of Tehran

Department of Electrical and Computer Engineering

Problem Description

According to Wikipedia, Computer vision is an interdisciplinary scientific field that deals with how computers can gain high-level understanding from digital images or videos. Although recently computer vision has grown closer to the field of computer science, its original, fundamental ideas were based on signals and systems concepts.

Consider a simple example: license plate recognition. The picture that a camera "sees", is actually a series of zeros and ones that are abstracted to form a 3 by m by n array, in which the amount of redness, greenness and blueness of each pixel from the image are represented by a vector of length 3. This array of values needs a structured pipeline to extract the alphabet symbols from this mess of numbers. In this assignment, we are going to design a basic pipeline to achieve this goal, using a correlation-based approach.

Conceptual Questions

1. Define the concept of correlation as simple as you can, avoiding the mathematical expressions.
2. Formulate correlation for two-dimensional maps (arrays). Simplify the formula as much as possible.
3. How one might use correlation in recognizing one object/symbol from a set of pre-defined object/symbols?

University of Tehran
Department of Electrical and Computer Engineering

Pipeline Implementation

Our goal is to write a script that opens a plate image file upon running and extracts its number and displays it in MATLAB console.

First provide the following functions:

- `plateArrC = openPlate()`

  This function opens a dialog window that asks the direction to plate image file. Upon selecting the file, it loads the contents of that file in the form of a 3d array. (Hint: you can open the dialog box using the MATLAB's uigetfile function.)

- `imgBw = rgb2Bw(imgColored)`

  Not all information in an image depend on the color extent, so in order to reduce the computation complexity, one can convert the colored image into a grayscale image, reducing the size of data to one-third. This function does so, using a linear combination of r, g and b values for each pixel with the following formula:

  ```
  g = 0.299 * R + 0.587 * G + 0.114 * B
  ```

  This formula closely represents the average person's relative perception of the brightness of red, green, and blue light. Now that the image is in grayscale, one can set a threshold for values in each pixel and reduce its content to zero and one, if the value was lower or higher than the threshold, respectively. To set this threshold value you can use many methods, or even do it manually for each image. However, we recommend the Otsu method, which has a MATLAB implementation in Image Processing Toolbox. Use this threshold level and create a binary map from your original image.

- `imgClean = CleanBw(imgNoisy, cleaningParams)`

  The binary map might seem a little noisy, as it has small patches of zeros and ones that contain no information, like the image on the fig.1-a. To clean this binary map, one can find the size of connected patches and omits the smaller ones, or use the methods called “eroding and dilating”. Choose one of these models and implement the desired function. You are allowed to use MATLAB’s pre-defined functions in the scope of this function.

  The desired output of this function is something similar to fig.1-b.

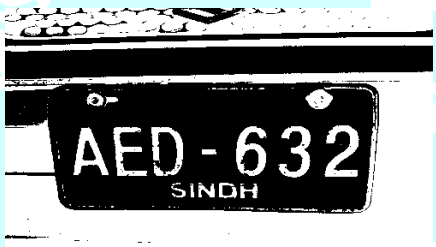

Figure2 -a: Noisy BW map

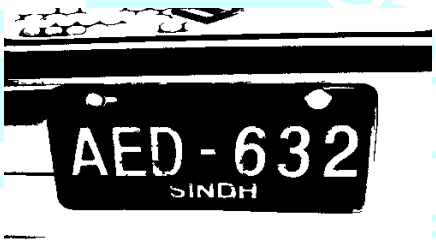

Figure1 -b: Denoised BW map

  This function can be used both as a mean of omitting noisy patches and larger patches that might be significant in our problem only by changing the cleaningParams. The identity of cleaningParams is based on the algorithm that you choose and is totally up to you.

- `mapSet = loadmaps(mapDir)`

  This function iterates through mapDir directory, loading each image in that directory along with that file’s name. The output argument, mapSet, may be a 2d cell array, a map (which is MATLAB equivalent of python’s dictionary) or any other data structure that you prefer.

- `rho = mapCorr(map1, mapSet)`

  This function will iterates through maps stored in mapSet and returns the correlation of each of these maps with the map1 that is specified as another input argument.

University of Tehran
Department of Electrical and Computer Engineering

Then use these functions and complete the following script:

```
close all

clear

clc

mapSet = loadmaps('Map Set');

plateArrC = openPlate();

plateArrM = rgb2Bw(plateArrC);

%% Clean the image, extracting patches that contain only
the alphabet on the plate.

[L, Ne] = bwlabel(imgClean);

License = [];

for iPatch = 1:Ne

      [r, c] = find(L == n);

      patchCrop = imgClean(min(r):max(r), min(c):max(c));

      patchCrop = imresize(patchCrop, [42,24])

      rho = mapCorr(patchCrop, mapSet)

      license = [license, <Name of map with highest
correlation>];
end

disp(license)
```

Signals and Systems

Dr. Saeed Akhavan

Computer Assignment 2

Deadline: 16th of Azar at 17:00

University of Tehran
Department of Electrical and Computer Engineering

General Notes about Homework!

- Ask your questions about this homework from me!
- Cheating is strongly prohibited in this course. You may cooperate with your fellow classmates but avoid duplicating each other's solutions.
- Refer to course's general rules for further details.