CAP 4410: Computer Vision Project 1: Transformations

Due Date: Thurs, Sept 16, 2021 **11:59 pm**
Submission Type: Upload onto Canvas

Problem Statement: In this assignment you use write code in Python 3 to create many transformed versions of an image. This way of creating transformed version of an image is a process, often called image augmentation, is used to increase the amount of training data for current deep learning-based approaches to object recognition (see page 275 of your textbook, Fig 5.28).

I_T = Transform (I), with the constraint that the size of the transformed image must be same as the input image.

Type of transformations to be implemented:

Let the input image be W x H sized.

As studied in class, there are two ways to output the transformed image:

(A) choose the portion of the output plane where the output image is mapped. This is done by the inverse_warp function studied in class. Note the output image could be different from W x H in this case.

(B) choose W x H portion of the output plane with the top-left corner remaining at (0, 0), i.e. we choose the portion of the output plane that coincides with the input image (the inverse_warp2 function).

- Translation X % shift, Y % shift, rest will be black. X and Y shifts are user specified, % is based on the total number of rows and cols in the image. Positive percentages are shifts to the right (or down), and negative ones are to the left (or up). Use method (A) to specify the output image.
- Rotate theta degrees. Theta is the angle of 2D rotation specified in degrees. Positive values denote counterclockwise rotation and negative values denote clockwise rotation. Use method (B) to specify the output image.
- Scale % the image. Greater than 100% denotes expansion, and less than 100% is contraction. Use method (B) to specify the output image.
- Affine, A, is a user specified vector of 6 parameters a_00, a_11, a_01, a_10, t_x, t_y. Use method (B) to specify the output image.
- Projective, H, is a user specified vector of 8 parameters h_00, h_11, h_12, h_10, h_11, h_12, h_21, h_22, (h_23=1) . Use method (B) to specify the output image.
- Contrast and brightness modulation of the L-channel of the input image input using "a" and "b". The output image will be the same size as the input and in RGB format.
- Gamma correction of the L-channel of the input image input using "a" and "b". The output image will be the same size as the input and in RGB format.
- Histogram equalization of the L-channel of the input image input using "a" and "b". The

output image will be the same size as the input and in RGB format.

- Compute the mean image and the standard deviation image of a collection of images.
- Batch normalize an image given a collection of images.

Use inverse warping to implement these transformations, otherwise you will run unto with gaps and holes in the final image.

Submission Requirements:
Please upload a ZIP file containing the following files:

1. All code files according to the REAME.md that comes with the skeleton code.
2. A 2-Page technical report containing the following sections :
    a. Pseudo-code for each of the transformations implemented, along specification of any assumptions about input and output.
    b. Show examples of each of the transformations for 2 images from the dataset provided.
3. Your code will be tested on different test inputs and graded based on the progress of your approach on these test inputs. A demo session could be required if there is problems with the code you submit.

Grading:

Each assignment will be graded out of 100:

1. Code (out of 15): quality of coding, readability, understandability (comments, variable names, etc.)

2. Correctness (out of 20): Will compare function outputs with different images and parameters and test against instructors code outputs.

3. Submission Requirements Met (out of 15): Can you follow the submission guidelines? Did you turn in your deliverables as instructed with proper naming conventions and packaging requirements as the instructor asked? Did you use the function prototypes correctly so that it works the way it was supposed to work? Basically, can you follow directions. At a real job, you have strict requirements and must adhere to them. Failure to do so can be disastrous.

4. Report (out of 50): In order to get top grades on the reports, you must have professional IEEE or CVPR formatted reports which are dual column, include good example images, mathematical formulas, etc. You can write them in any word processor but for the best results, check out Overleaf.com which uses LaTeX (lay-tech) but you do not need to use it. Your report must be in the zip submission as a PDF, report.pdf. Make sure to spend time on this, it is worth half your grade and should be taken seriously.

Solutions to your **programming assignments** have to be self-sufficient and **not dependent on other computer vision code, such OpenCV vision package**. You may use packages for display graphics or mathematics packages, such as for linear algebra (numpy) or graphs or optimization.

All reuse of code has to be clearly acknowledged in the source code, any README files, and also in the report. Failure to do so will be considered plagiarism.