# CAP 4410: Computer Vision Assignment 1

# Due Date: Friday, Sept 10, 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).

IT = 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:

* translation (t\_x, t\_y)
* rotate (theta)
* similarity (s, t\_x, t\_y)
* affine (a\_00, a\_01, a\_10, a\_11, t\_x, t\_y)
* projective (h\_00, h\_01, h\_02, h\_10, h\_11, h\_12, h\_20, h\_21, h\_22=1)
* subtract “mean image” each image.

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 your code files, including any helper files/dependencies.
2. A README file detailing how to run your code along with any compilation instructions.
3. A 2-Page technical report containing the following sections:
   1. Pseudo-code for each of the transformations implemented, along specification of any assumptions about input and output.
   2. Show examples of each of the transformations for 2 images from the dataset provided.
4. 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 will be scheduled to evaluate your implementation.

**Grading:**

Each assignment will be graded out of 100:

Code (out of 50) quality of coding, readability, understandability (comments, variable names, etc.)

Report (out of 50)

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.**