# **CVI620/DPS920**

# **Project- Milestone 2- Dataset and Evaluation**

| Total Mark: | 10 marks (20% of the total project grade)   * 7 out of 10: Blackboard submission * 3 out of 10: Lab demo |
| --- | --- |
| Submission file(s): | * Training.csv * Test.csv * Milestone2.cpp * Project\_milestone2.docx |

Please complete this milestone with your **group**. Only one submission per group please.

This milestone is designed to help you in implementing your project. Not all the steps may be relevant to your project. Therefore, it would be beneficial to talk to your instructor about your specific project.

1. Add this declaration to your file:

We, ------------ (mention your names), declare that the attached assignment is our own work in accordance with the Seneca Academic Policy. We have not copied any part of this assignment, manually or electronically, from any other source including web sites, unless specified as references. We have not distributed our work to other students.

1. Gather a dataset of images or videos containing samples you can work with- the more, the better. These samples should contain positive and negative samples, easy and difficult samples, etc. depending on your project. Possible methods are:

* Take the images or record the videos yourself. (Be mindful of other’s privacy and do not violate it.)
* Use Google to find and download images. (Use those labelled for noncommercial reuse)
* Search for datasets available online. Many researchers share their datasets online.

Prepare a .csv file (named **data.csv**) to contain a list of your data (one row per sample)

Example 1: Reading credit card number- a dataset of images of credit cards, at different distances, lightings, angles, tilts, on top of other objects, etc.

Example 2: Detecting drowsy drivers- a dataset of video sequences containing drowsy and non-drowsy people, possibly with various skin color, and at different lighting

1. Ground Truth (gt)

Gather the output you expect your algorithm to produce, if it was working perfectly. You might need to write code providing you with a GUI to label your data. Add a column (or columns) to data.csv file to include the ground truth (or filename for ground truth) for each data sample.

Example 1: Each row of data.csv contains: i) filename for an image in the dataset mentioned in (2), and ii) the credit card number shown in the image

Example 2: Each row of data.csv contains: i) filename for a video sequence in the dataset mentioned in (2), and ii) a flag to specify whether the person in the video was drowsy or not (e.g. 0 or 1). Alternatively, if each video sequence contains one or more times the driver looks drowsy, instead of the flag, we could save a list of the starting time(s) in seconds of drowsiness. This list will be empty if the driver does not appear drowsy in the whole video sequence.

1. Training vs. Test

Write a function named **splitTestTraining** to **randomly** select 70% (up to 80%) of your dataset and save them in a folder named **training**. Move the rest to another folder named **test**. Do not use the test set for optimizing your code. Keep it for performance evaluation only, at the end of your project. In the same code, split the csv file into two csv files for the training and test sets, i.e. **training.csv** and **test.csv**. When done, add title rows to explain the contents of your csv files.

1. Evaluation

Implement a function, named **evalFun**, to calculate the evaluation measures for your algorithm, e.g. precision, recall, F1, PR curves, etc. The output is the performance measure value (or multiple measures). The input to this function is a list of output values (what your method found), and a list of ground truth values (what it should have found if it was working perfectly). Consult with your professor if you are unsure about the evaluation method.

If you are using a different evaluation measure, explain your method here and implement it in **evalFun.**

1. Test measures

Write code, named **milestone2.cpp** to use above function and csv files. Test your function with the ground truth. You should get perfect performance. Test it with random values. This is the baseline performance based on chance. Your project, when done, should do better than chance!

1. Specify what each member has done towards the completion of this work:

|  | Name | Task(s) |
| --- | --- | --- |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |