# Computer Vision 2023 Project [SC]

# Handwritten Signature Identification and Verification

Authentication is an important factor to manage security. Signatures are widely used for personal identification and verification. Many documents like legal transaction and bank cheque requires signature verification. Signature-based verification of a multiple number of documents is a time-consuming and difficult task. Consequently, an explosive growth has been seen in biometric personal verification and authentication systems as traditional identity verification methods such as passwords, tokens, pins suffer from some fatal flaws and are incapable to satisfy security necessities.

In this project you will be tasked with identifying to whom a handwritten signature belongs to and whether it is real or forged through a multistage pipeline.

Project Objectives:

1. Apply Image preprocessing or Feature Extraction techniques where needed
2. Train a classification model to determine to whom a handwritten signature belongs to. You can use classical computer vision or regular deep learning for this stage
3. Train another model(s) to determine whether the signature is real or forged. In this stage you can either train a single Siamese model or multiple classical/regular deep learning models (A model for each person to determine whether a signature is real or forged).

4. **Part 2: added in Lab 9**: Signature object detection in document. Apply one object detection model (Faster RCNN or YOLO and so on) on the new detection dataset.
   **Minimum Requirements**:
   a. You have to use at least two different methods for the two stages (You cannot use classical methods for both step 2 and step 3 or regular deep learning for both).
   b. You cannot use pixels directly as features in case of classical computer vision. You have to use a feature extraction method first (e.g BoW using SIFT) and then use a classifier like SVM or logistic regression.
   c. In case of a team of 6. You will be tasked with applying two different techniques for step 3. In case of classical method you should use a different feature extraction method than stage 1 or a different deep learning architecture. [e.g if you used classical method for stage 1. In stage 2, you should apply two different techniques so that would be classical again and regular deep learning. In this case you should use a different feature extraction method than the one used in stage 1 (e.g BoW)]

# Dataset Description

**The dataset for part 1 of the project can be found** [here].

This dataset provided contains of 5 subfolders and each subfolder of these 5 contains train and test folders

- personA
  - Train – 40 images containing person A signatures + csv file containing signature verification labels
  - Test – 8 images containing person A signatures + csv file containing signature verification labels
- personB
  - Train – 40 images containing person B signatures + csv file
  - Test – 8 images containing person B signatures + csv file

- personC
    - Train – 40 images containing person C signatures +csv file
    - Test – 8 images containing person C signatures +csv file
- personD
    - Train – 40 images containing person D signatures + csv file
    - Test – 8 images containing person D signatures +csv file
- personE
    - Train – 40 images containing person E signatures + csv file
    - Test – 8 images containing person E signatures + csv file
- ➢ In stage 1 classification, you should train the model using the 200 training images (40*5) with 5 classes as output: PersonA, PersonB,PersonC, PersonD and PersonE. You should also report the test accuracy on the 40 test images (8*5). So, you should concatenate the test sets from the 5 folders in this stage
- ➢ In stage 2 classification, you can either create a separate verification model for each person and report the test accuracy on the 8 images for each model or create a single Siamese model and report the test accuracy on the 40 test images.

**Part 2 dataset can be found here**

This dataset consists of four folders:

- ‘TrainImages’ folder : contains the training images (660 images) for the detection task
- ‘TrainGroundTruth’ folder: contains the corresponding detection labels (660 files) for training detection task
    - Each image in ‘TrainImages’ folder has a corresponding text file in this folder with the same name as the image
    - The text file has (1 or more) rows. Each row represents the bounding box of a single signature.
    - Each row has 4 values: x1,y1,x2,y2
- ‘TestImages’ folder : contains the training images (115 images) for the detection task

- ‘TestGroundTruth’ folder: contains the corresponding detection labels (115 files) for testing the detection task

## Practical Exam Project Deliverables:

1. Apply Image classification on a signature to identify who it belongs to (Deliver Code)
2. Apply Image classification on the signatures of each person to identify who it belongs to (Deliver the code for 5 models in case of using classical/CNN or 1 model in case of Siamese)
3. **New:** Apply object detection on the signature detection dataset
4. If you trained the deep learning/Siamese models using a notebook, you must deliver the notebook with the output cell saved displaying the training logs. If you trained the model using IDE (i.e Pycharm). **You must deliver screenshots of the training process**.
5. **New:** create a test script for part 1 of the project where given an image of a signature, you print who it belongs to and whether it is fake or real.

6. A Report that includes description of:
   - Your data preparation process.
   - Brief description of the models and techniques used in each task.
   - Training and Testing times for each model.
   - Image Classification training and testing accuracy.
   - New: Object detection performance metrics for training and testing
   - Provide screenshots of the test sets classification with visualization.
   - New: Provide screenshots of the test sets detection with visualization.