

# Project Guideline: Handwritten Digit Recognition using C++ and OpenCV

## Objective:

The objective of this assignment is to utilize the C++ programming language along with the OpenCV library to perform handwritten digit recognition using the MNIST dataset. The task involves downloading and installing the OpenCV library, downloading the MNIST database, and design or adapt an algorithm to recognize the digits in the database. Finally, you are required to produce a comprehensive assignment report.

## Step 1: Installing OpenCV

1. Visit the official OpenCV website (<https://opencv.org/get-started/>) and download the latest version of the library compatible with your operating system.
2. Follow the installation instructions provided on the website to install OpenCV on your machine.
3. Verify the installation by compiling and running a simple OpenCV program.

## Step 2: Downloading the MNIST Database

1. Visit the MNIST database website <http://yann.lecun.com/exdb/mnist/> download:  
[train-images-idx3-ubyte.gz](#): training set images (9912422 bytes)  
[train-labels-idx1-ubyte.gz](#): training set labels (28881 bytes)  
[t10k-images-idx3-ubyte.gz](#): test set images (1648877 bytes)  
[t10k-labels-idx1-ubyte.gz](#): test set labels (4542 bytes)
2. Extract the contents of the downloaded files to obtain the training and testing images along with their corresponding labels.

## Step 3: Implementing Handwritten Digit Recognition

1. Create a new C++ project in your preferred Integrated Development Environment (IDE).
2. Include the necessary header files to work with OpenCV and other required libraries.
3. Write a function to load the MNIST dataset into memory.
4. Design, implement or adapt the algorithm using the MNIST training set images.
5. Write a function to recognize handwritten using the newly implemented algorithm. This function should take a test image as input and output the recognized digit.
6. Test the digit recognition algorithm using the MNIST test images and evaluate its accuracy.

Here, we highly recommend you to use one of the classical machine learning methods:

Methods	Fundamentals	OpenCV example
K Nearest Neighbors (KNN)	<a href="#">Tutorial</a>	<a href="#">Opencv tutorial</a>
Support Vector Machine (SVM)	<a href="#">Tutorial</a>	<a href="#">Opencv tutorial</a>
K-means clustering (Kmeans)	<a href="#">Tutorial</a>	<a href="#">Opencv tutorial</a>
Decision Trees	<a href="#">Tutorial</a>	<a href="#">Opencv tutorial</a> : chapter

		Decision Trees
Deep Neural Network (DNN)	<a href="#">Tutorial of multilayer perceptron</a>	<a href="#">Opencv tutorial</a> : chapter Neural Networks

#### Step 4: Writing the Project Report

Create a brief report documenting your approach, code implementation details, experimental results, and any observations or conclusions you made during the process. Additionally, include relevant visualizations, such as sample images from the MNIST dataset.

**Your report must be no longer than 4 pages in total (in A4 size).** You can refer to the following steps to help organize your report:

1. Start by briefly introducing the problem of handwritten digit recognition and its significance.
2. **List the commands and operations to reproduce your project based on your submitted code packages.** This section is **IMPORTANT** since it may affect your scores in the coding section if your illustration is not clear enough.
3. Present your code implementation, highlight important sections and explain any complex logic or algorithms used.
4. Present the overall results, e.g. conclude your experimental results in a quantitative or qualitative way.
5. Analyze your results, and conclude the report with a summary of your findings.

**Note:** Make sure to adhere to proper coding practices, such as modularization, commenting, and following naming conventions. Additionally, provide clear explanations and justifications for your design choices throughout the report.

#### Grading Criteria:

Report (40%): The report file should be submitted in a **.pdf** form.

Code (60%): The codes of your project implementation. Make sure the codes you packed are complete enough to reproduce your project implementation.

#### Submission:

Pack your **codes** and **report** in a zipped folder and name it as **proj\_12XXXXXXX.zip** (replace 12XXXXXXX with your student ID).

Good luck with your project!