

Project Report for

**CE4041 – Artificial Intelligence**

Classifier implementation for the MNIST handwritten digits database

Revision Timestamp:27/10/2024 17:58:58

# Project Team

|  |  |
| --- | --- |
| Student Name | Student Id |
| Qinyuan | 20137095 |
| Eamon Moloney | 8457077 |
| Ibrahim Saana Aminu | 25381993 |
| Des Powell | 9513833 |
| Terence Coffey | 15223124 |

Contents

Project Team [2](#__RefHeading__652_755868819)

Introduction [4](#__RefHeading__654_755868819)

Approach [4](#__RefHeading__656_755868819)

Investigate Solution Space [4](#__RefHeading__658_755868819)

Design Decisions [5](#__RefHeading__660_755868819)

Results [5](#__RefHeading__662_755868819)

Accuracy [5](#__RefHeading__664_755868819)

Lessons learned and changes for future Projects [6](#__RefHeading__666_755868819)

Conclusion [7](#__RefHeading__668_755868819)

References [8](#__RefHeading__670_755868819)

# Introduction\*\*

The work implemented in this project creates a Classifier for the Modified National Institute of Standards and Technology (MNIST) handwritten digits database using the Keras Neural Network library and coding in Python. The classifier is designed to achieve a testing accuracy of at least 99 percent. The MNIST[1] dataset contains 60,000 28 x 28 grey scale images of handwritten digits and their associated labels for training and 10,000 images for testing. The classifier is a 10-way classifier with 10 possible digits (0-9) and 10 possible digit classes. Each bit in the image (8 x 8) has 8 bit giving a total of 784 inputs to the classifier. This represents just one MNIST input image. The output is a sparse-encoded 10 element binary vector representing the digits. The MNIST dataset is prebuilt into the Keras dataset.

# Approach

**Team Work:**

An in-person meeting was used as the basis for deciding on the understanding of the project requirements and the approach to adopt in the code implementations. Based on these meetings, the group unanimously adopted Google colab notebook to allow for effective collaboration with input at various levels by the team members. The final codes were run centrally on one group member’s computer with each member also running the codes individually to scan for error and necessary adjustments where necessary. The final project report was created, shared and edited by all team members via OneDrive. The content of the final submission was agreed upon by all members.

# Investigate Solution Space

. We observed that there are many ways of achieving a performance level of 99 percent. Generative AI such as Gemini and ChatGP allowed us to quickly create a prototypes to see the test accuracy results for our choices. An entry in Wikipedia [1] for MNIST database provides a table which showed the error rate for different types of Neural Network models. The range of workable solutions include Convoluted Neural Networks(CNNs) , K-Nearest Neighbours, Linear and non-linear classifiers.

We decided to test the test accuracy based on a number of parameters. These were

|  |  |
| --- | --- |
|  | number of Conv2D layers, first parameter(32,64,...), kernel size, activation function) |
|  | number of drop out layers, drop out percentage. |
|  | validation split 0.1 to. 25 (on training data) plus a setting which dynamically varies this percentage between epocs |
|  | earlystop instead of running to final epic, it stops once the algo determines no additional progress of sufficient quantity can be made. |
|  |  |

# Design Decisions

Report review

# Results

In other s

Fig 5, Shows files transferred(video and photo) files from Raspberry Pi to a host.

## Accuracy

Provide the classification accuracy on the test and taining sets.

The training set should be split into a training and a validation portion. Want accuracy cor testing, validation and training set.

*By comparing the cassification accuracy of the 3 sets sets, estimate if the classifier is over fitted and why.*

Discuss the Project especially

* How you choose to tackle it
* What design decisions you made
* What the results are like
* What you might do better/differently next time you had to tackle a similar project
* If plots are called for they should be in your code and in your report.
* Marks for neat well designed code with appropriate level of comments
* neat logically laid out and informative reports.
* Provide classification accuracy for the training and test data. The test data should be split in the ration 70 to 80 and the baance for validation.

# Future outlook

In thi

I selected an industrial

The Raspberry Pi Camara module 2 replaced the original Camera Module(V1 Series) in April 2016 [8]. The original Camera Module V1 was

|  |  |  |  |
| --- | --- | --- | --- |
| Comparison of Camera sensor [11] | | | |
| Feature | Option 1 : OV5647 Model | Option 2: Sony IMX219 | Industrial camera [6] |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table 1 Camera Sensors comparison

This upgrade provides more than an improvement in resolution and image quality[10]. Improvementsking the day aowever this needs to be dynamic.

# Conclusion

Omni Vision(V1) OV5647 and Sony the end of 2028. Overall, both camera modules are excellent general purpose camera modules for taking still pictures and video.

# References

[1] https://en.m.wikipedia.org/wiki/MNIST\_database

[2]

[3]

[4]