Student Name: Yi Qian Goh
Course Name: CSEN140 Machine Learning and Data Mining
Date: 1 November 2024
Project Title: 3D MNIST Handwritten Digits Classification and Visualization

Project Summary/Introduction

This project aims to develop a machine learning pipeline for classifying 3D representations of handwritten digits based on a modified version of the MNIST dataset. The transformation into 3D point clouds introduces a novel aspect to the classic 2D dataset. Point clouds enhance the classification challenge and offer a rich visualization potential. I will explore the effectiveness of different machine learning and deep learning models in classifying this voxelized data, as well as experiment with 3D visualizations to enable better interpretability and presentation of classification results. This project aims to create a robust model for 3D digit classification and utilize computer vision techniques to produce intuitive 3D visualizations of each digit.

Description of the Dataset

This dataset is gracefully provided by David De La Iglesia Castro on Kaggle. This data set is a 3D-augmented version of the MNIST handwritten digit dataset, created by voxelizing 2D images into 3D point clouds. The original images are transformed into 3D data through the following process:

- Each 28x28 grayscale MNIST image was converted into a 16x16x16 voxel grid to represent the digit in 3D space.
- The data is stored in an HDF5 format with two main arrays:
  - `X_train`: 10,000 training samples with each digit represented as a 4096-D vector from the voxelized 3D point cloud.
  - `y_train`: 10,000 corresponding labels.
  - `X_test`: 2,000 test samples.
  - `y_test`: 2,000 corresponding labels.

- Additional transformations include random rotations and noise applied to point clouds to augment the dataset.

Proposed Approach

I. Data Preprocessing and Exploration
   - Normalize the 3D data points for consistency.
   - Visualize and analyze sample point clouds using 3D plotting tools to understand digit shape patterns.

II. Model Selection and Training
   - Baseline Models: Experiment with variations in neural network architecture (e.g., Convolution Neural Networks (CNN) adapted for voxel data or using 3D convolution layers) to improve accuracy.

    - Since CNN is a more advanced deep learning model, it may be challenging to implement with my beginner level ML knowledge. Thus, an alternative model I can implement for classification is K-Nearest Neighbors (KNN) to establish baseline performance.

III. Evaluation and Metrics
    - Evaluate models using accuracy and loss.
    - Plot the accuracy and loss of the train and test datasets
    - (If time persists) Use cross-validation and hyperparameter tuning to optimize model performance.

IV. 3D Visualization of Results
    - Generate 3D visualizations of correctly and incorrectly classified digits to explore model decision boundaries.
    - Visualize model confidence on 3D data using color-coded point clouds, emphasizing the model's confidence levels in various regions of each digit.

Software I Will Use

- Programming Language: Python
- Libraries:
    - Data Processing: NumPy, pandas, h5py (for HDF5 data handling)
    - 3D Visualization: Matplotlib or Plotly for interactive visualizations
    - Machine Learning:
        - TensorFlow/Keras and PyTorch for the 3D CNN implementations
        - Scikit-learn for alternative KNN implementation
- Development Environment: Jupyter Notebook

Plan of Work

I. Week 1: Data Preprocessing and Exploration
    A. Load and normalize the data from HDF5 files.
    B. Visualize samples of 3D point clouds to establish intuition for the dataset and its complexities.

II. Week 2: Baseline Model Development
    A. Implement and evaluate CNN baseline models using Tensorflow/Keras and Pytorch
    B. Document initial findings and baseline metrics

III. Week 3: Train, Test, and Model Fine Tuning
    A. Develop and train models for 3D point cloud classification.
    B. Evaluate model performance on test data set
    C. Experiment with variations (e.g., KNN)

IV. Week 4: Visualization and Analysis of Results
    A. Create 3D visualizations of model predictions, particularly focusing on misclassifications.
    B. Document final results, graphs of accuracy, insights derived from the project, and prepare for the final report.

References

Castro De La Iglesia, D. (2019). *3D MNIST*. Kaggle. https://www.kaggle.com/datasets/daavoo/3d-mnist

Keita, Z. (2023, November 14). *An Introduction to Convolutional Neural Networks: A Comprehensive Guide to CNNs in Deep Learning*. Datacamp. https://www.datacamp.com/tutorial/introduction-to-convolutional-neural-networks-cnns

“In the realm of civil engineering, time is money. The judicious use of public datasets can significantly accelerate the due diligence process, providing project managers with crucial information about the project site’s boundaries, topography, habitats, hydrology, soils, and zoning regulations. Armed with this knowledge, engineering firms can formulate efficient strategies and approximate estimates, ultimately leading to smoother project execution and increased overall efficiency. Embracing the power of public datasets is more than a technological advancement; it’s a smart business move that can elevate the success of civil engineering projects in today’s dynamic landscape.” (Civil Engineering Datasets for Assessing Project Feasibility)

RF Signal Data
90%Val_Acc_3D_Points_Classification
ModelNet40 - Princeton 3D Object Dataset
- PointNet for 3D Object Classification II [PyTorch]
- 90%Val_Acc_3D_Points_Classification

3D MNIST
- 3D-MNIST | basic CNN | Adorable visualisations

Project title
• Data set
• Project idea
• The approach you will use
• Software you will use
• References.
• Teammate (if any).
• Timeline.

C.