Big Data Project 1 Proposal

Larry Liu[1] and Oscar Scholin[2]

[1]Pomona College
[1]Pomona College

February 12, 2024

Building a better MNIST classifier based on differential geometry and graph theory

Our goal is to develop two deterministic algorithms that will solve high dimensional classification-based problems. In particular, we are going to use the MNIST handwritten digits dataset. We have three algorithms in mind:

1. A differential geometric approach based on our conversations with Prof. Gu
2. A graph theory network approach we have came up with
3. A geometric method approach that fits a spline on the contour and extracts features

Our code is hosted at `https://github.com/oscars47/BigData/`. Let's go through each idea.

1 Differential geometric approach

1.1 Centralizing Data

- Normalize pixel values p_i by thresholding to 0 or 1.
- Select pixels with value 1, compute average x and y coordinates.
- Shift all pixels by this average to centralize the data.

1.2 Applying Principal Component Analysis (PCA)

- Extract two principal eigenvectors to form rotation matrix P.
- Apply P^T to rotate data for alignment.
- Normalize data by finding bounding box and rescale:
  - Use polar coordinates and least squares for outlier exclusion.
  - Normalize vectors by their magnitude.
- Eigenvalue λ_2 indicates line-like structures (e.g., digit "1").

1.3 Special Handling for Certain Digits

- **Digit "6":**
  - Apply edge detection and pixel density analysis for splitting.
  - Use rotational and filtering techniques with non-square filters.
  - Post-process to clean residual parts.
- **Digit "9":** Similar to "6", adapted for "9".
- **Digit "5":** Use rectangular strip filters for analysis.

2 Graph theoretic

2.1 Preparing data

1. Normalize pixel values p_i by thresholding to 0 or 1.
2. Find the "bounding box" within the image
3. Divide the bounding box into a set number of regions

2.2 Building the network

1. Initialize an empty network
2. If there exists a non-0 pixel value within this box, place a node in the network at the "center of mass" of this box.
3. For each node in the network, if there exists a point in an adjacent box, connect this node to that node.

2.3 Classification

(a) Using graph similarity, we will compute the distance of each graph network to known targets for each of the digits that is indifferent to the rotation/scaling of the individual network.

See Figure 1 for an illustration of this process.

3 Geometric Method for MNIST Digit Recognition

Contour Extraction: The first stage involves applying edge detection algorithms, such as the Canny edge detector, to each 28x28 pixel grayscale image in the MNIST dataset. The goal is to accurately outline the contours of the digits, which are fundamental for subsequent geometric analysis. Post-processing steps like Gaussian blurring may be applied to reduce noise and improve contour detection accuracy.

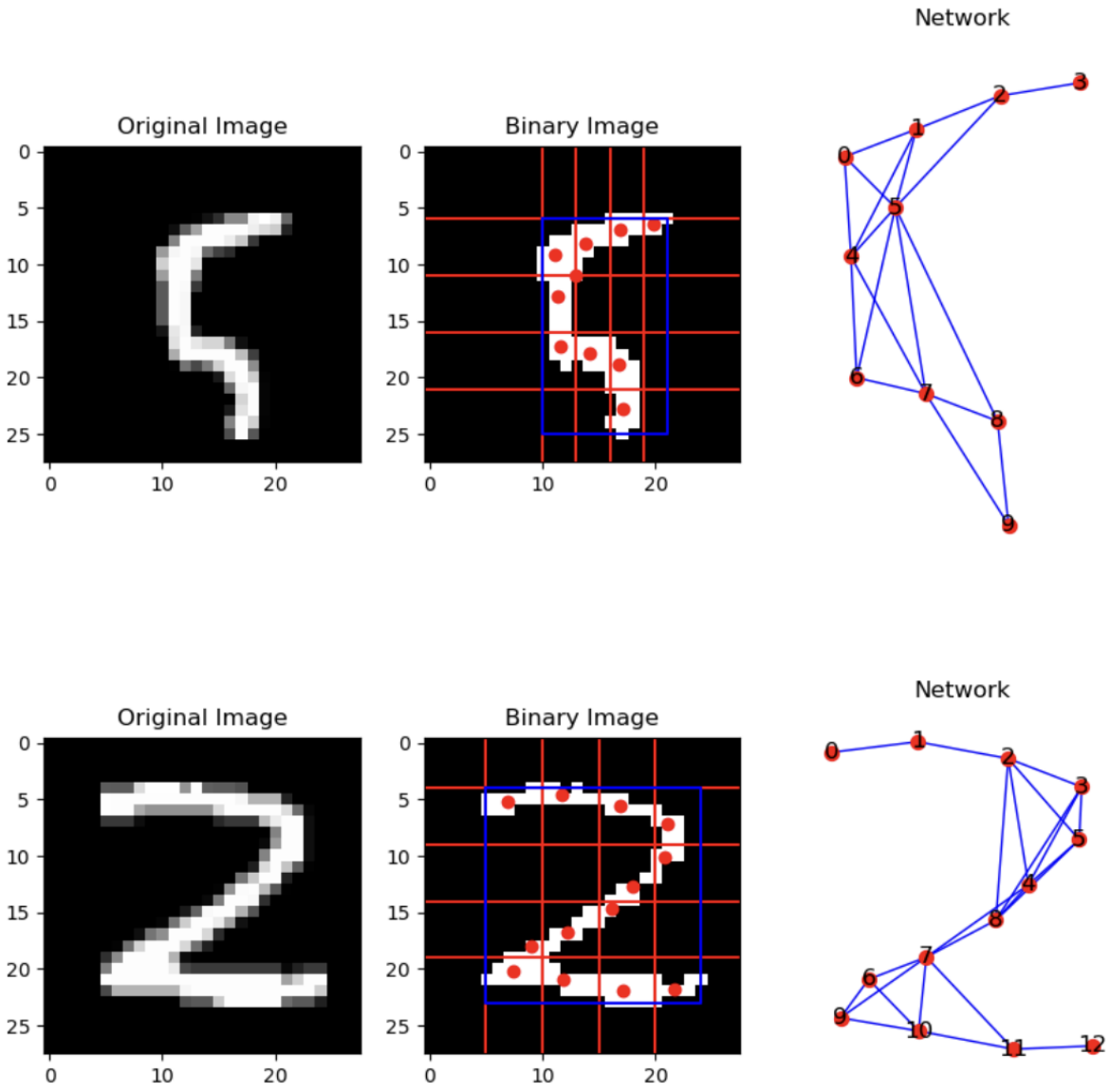

Figure 1: Sample of steps 2.1 and 2.2

Spline Fitting: Upon successful contour extraction, each contour is transformed into a continuous curve using spline fitting techniques. This step involves fitting a cubic spline or a Bézier curve to the contours. The spline representation smooths the jagged edges of the digit's contour and provides a mathematically convenient form for extracting geometric features. The fitting process must balance fidelity to the original contour with the smoothness of the curve to avoid overfitting to noise.

Feature Extraction: The spline representation of each digit allows for the extraction of various geometric features. These features include:

- *Curvature:* Measure the rate of change of the digit's contour direction, which helps in identifying loops and curves.

- *Inflection Points:* Points where the curvature changes sign, useful in distinguishing digits like 2 and 5.
- *Loop Detection:* Identify closed loops, critical for digits like 0, 6, 8, and 9.
- *Linearity:* Assess the extent to which parts of the digit follow a straight line, important for digits like 1 and 4.
- *Symmetry and Proportions:* Analyze the digit's symmetry and the proportions of its constituent parts.
- *Angles Between Lines:* Evaluate the angles formed by intersecting lines, particularly relevant for digits like 4 and 7.

Digit Classification: After feature extraction, a classification algorithm is applied to categorize each image into one of the ten digit classes. This stage can employ various techniques, ranging from simple rule-based classifiers (if the geometric features are distinct and well-separated) to more sophisticated machine learning models like Support Vector Machines (SVMs) or even neural networks.

Normalization and Testing for Robustness: It is essential to normalize the images in terms of size and orientation before feature extraction. The effectiveness and accuracy of the method will be tested against the MNIST dataset's diverse range of handwriting styles. Additionally, cross-validation techniques will be used to evaluate the model's performance and generalize its applicability.

Conclusion: This proposed geometric method, with its focus on the intrinsic shape and structure of each digit, represents a novel approach to digit recognition in the MNIST dataset. The methodology aims to be robust against variations in handwriting and could potentially complement existing pixel intensity-based recognition methods.