Project Report:
A Visual Exploration of Manifold Learning for Images

Mattia Bruno Stellacci

June 24, 2022

Outline

Motivation

- Create a system to facilitate understanding dimensionality reduction for images
- *Why dimensionality Reduction?*
- *Why images?*
- Allow the user to visualize and manipulate raw data and lower-dimensional embeddings
- Allow users to compare different dimensionality reduction techniques

Goals

- Build a powerful, modern Visual Analytics platform in the browser
- Harness a `WebAssembly` port of `OpenCV` to incorporate image manipulation in the browser
- Incorporate multiple dimensionality reduction/manifold learning techniques to allow users to compare them conveniently.
- Create visual support for some of the notable properties of lower-dimensional embeddings: Clustering of similar instances,separability, Variance, Compactness.

Intended Users

- **Students**
- **Machine learning engineers/Data scientists**

Related Work

- Dimensionality Reduction extremely common for Data Visualization:
  - Many publications in the 00's focus on/describe the techniques themselves.
  - 2000: 'A global geometric framework for nonlinear dimensionality reduction'
  - 2010 Survey: "Data Visualization: Manifold Learning for Visualizing and Analyzing High-Dimensional Data"
  - Later papers make use of this technique to visualize their data, less focussed on the techniques *per se*
- Big Data: *Curse of Dimensionality*
- Machine Learning: New found interest in dimensionality reduction, the Manifold hypothesis:
  - Word Embeddings (word2vec)
  - Generative models (VAE)
- Couldn't find a work on a visual analytics system that combines dimensionality reduction/manifold learning for images with automatic image processing

Implementation

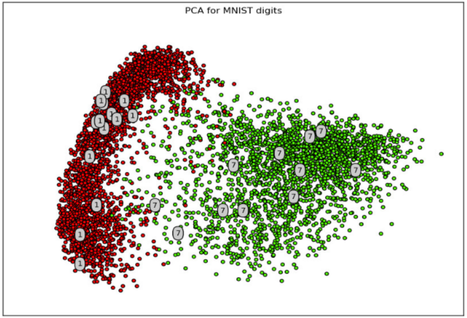
Reminder: Vision/Mockup

Digits to include

○ 0 ◉ 1 ○ 2 ○ 3 ○ 4
○ 5 ○ 6 ◉ 7 ○ 8 ○ 9

--Please choose an option--

- --Please choose an option--
- Principal Component Analysis
- Multidimensional Scaling
- Locally Linear Embedding
- Isomap Embedding
- T-distributed Stochastic Neighbor Embedding



Parameters/Stats of the current embedding

[Space to display the digits selected in the above graph. Either by selecting individual Points or a region]

Technologies I chose:

- **Dataset:** MNIST Handwritten Digits
- **Frontend:** `ReactJS`
- **Charting Library:** `react-vis` (RIP)
- **Backend/Embedding:**
  - Python/`bottle`
  - `scikit-learn`
- **Image Processing:** `OpenCV`

Dimensionality Reduction techniques

Dimensionality reduction techniques I included:

- **Principal Component Analysis**
- **Isomap embedding**
- **Locally Linear Embedding**

Didn't make the cut:

- **TSNE**
- **MDS**
- **Variational Autoencoder**

Insights I gained/Challenges I encountered:

- **Diversity:** One size does not fit all.
- **Explainability** What do the results mean?
- **Comparability** Random Sampling from huge Dataset might not be the best idea.
- **Interpretebality:** Sampling and Inverse transform.
- **Specificity**: Focus on a single case.
- **Bidirectionality:** Harness the true power of interactivity and aggregation

Demo

Conclusion