

Learning the 2-D Topology of Images

Nicolas Le Roux, Yoshua Bengio, Pascal Lamblin,
Marc Joliveau, Balázs Kégl

October 19th, 2007

- 1 Introduction
 - The task
 - The idea
- 2 Experiments and Results
 - Settings
 - Embeddings
 - Reconstructing Images
- 3 Conclusion and Future Work

Introduction

- We usually oppose generic and domain-specific (with prior knowledge) learning algorithms
- There are lots of techniques that are developed especially for learning on images (convolutions...)
- Is the prior that a data set consist in images a strong one?
- Or can we learn it from a limited number of samples?

The Task

- We have a set of images coming from some distribution
- But without any information on the ordering of the pixels
- The pixels have been randomly permuted (same permutation for all images)
- We want to recover the original image
- Or at least an image on which we can apply image-specific learning algorithms
- Is it possible without testing every permutation?
- How many samples do we need?

Correlation and Pseudo-Distance Between Pixels

- In natural images, the values of two neighboring pixels are usually very strongly correlated across different images
- Further away pixels are more independent from each other
- Correlation across images in the data set (or its absolute value) can be seen as a similarity measure
- If we assume it to be the value of a Gaussian kernel:

$$|\rho_{ij}| = K(x_i, x_j) = e^{-\frac{1}{2}\|x_i - x_j\|^2},$$

we have a proxy for the distance between pixels i and j :

$$D_{ij} = \sqrt{-\log |\rho_{ij}|}$$

- We can apply several manifold learning techniques to assign every pixel a coordinate on a 2-D plane, using this pseudo-distance measure

Some Details

- On some datasets (like MNIST), some pixels have a really low variance, leading to the manifold folding on itself
- We removed low-variance pixels

Manifold learning techniques we used:

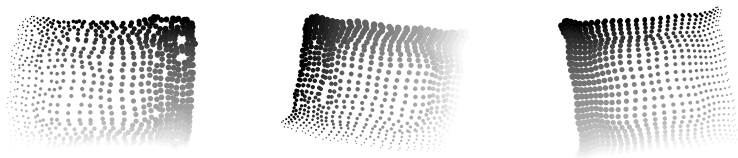
- Isomap
- LLE
- MDS
- MVU

Data sets

We used several image (or image-like) data sets

- MNIST (28×28)
- NORB with background (27×27)
- Spectrograms from blues music

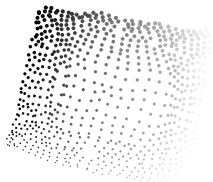
Isomap on NORB



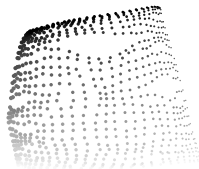
(a) 250 training images (b) 500 training images (c) 2000 training images

Figure: Embedding discovered by Isomap on the NORB dataset, with different number of training images.

Different Algorithms on NORB



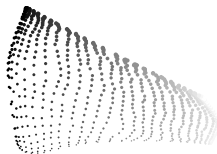
(a) Isomap embedding



(b) LLE embedding



(c) MDS embedding



(d) MVU embedding

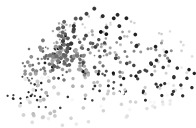
Different Algorithms on MNIST



(a) Isomap embedding



(b) LLE embedding



(c) MDS embedding



(d) MVU embedding

Retrieving the Original Pixel Coordinates

- The embedding we get can be arbitrarily scaled (in both dimensions), rotated, and translated
- In order to compare the original and reconstructed coordinates, we find the linear transformation that minimizes the distance between true and reconstructed coordinates [Please imagine cute figure here]
- Measured RMSE on NORB: 1.16 pixel with 250 examples (1.05 with 2000)

Cute image



Mapping Images Back to a Grid

- Optional: scale and rotate embedding to make it square
- Convolve into a regular grid image:
 - For each pixel in the grid, we do a weighted average of the values of neighboring points
 - The weights depend on the distance

MNIST Examples

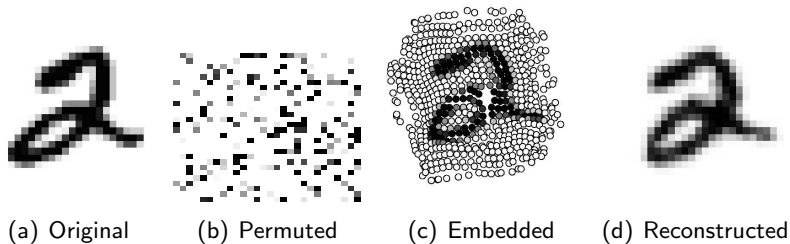


Figure: Example of the process of transforming an MNIST image

Conclusion

- We can recover the 2-D topology of images
- We only need a small number of examples to do that
- Then, image-specific algorithms are applicable

Future Work

- Apply image-specific learning algorithms on reconstructed images
- Try the algorithm on data sets where we don't know there are images
- In particular, try a generalized version in more than 2 dimensions
- Try different pseudo-distances functions (closer to the real distance)
- Try different ways to reconstruct images
- Recover the permutation itself

Anniversaire

Happy Birthday Hugo!

The End

Questions?