Applications of PCA Denoising in Image Classification

Andrew Duenner
Alex Kimn

April 20, 2018

Preferred Presentation Date

May 17, 2018

Programming Libraries

- Language: Python 3
- Libraries: OpenCV, Numpy, pyTorch, PyBM3D

Algorithm

We will implement principal component analysis as a method to reduce the dimension of an image represented as a matrix. Our primary goal is to reduce the noise in an image represented by a matrix $A \in \mathbb{R}^{m,m}$. PCA allows us to pick the subspace with dimension $k \ll m$ that it most representative of the variance in the image matrix. Two promising methods for efficiently calculating PCA for very small k are the NIPALS method [3] and power iteration. For calculating many principal components, these methods can be susceptible to loss of orthogonality and an alternative method must be used. The most common method for calculating a large number of principal components involves calculating singular values [5] [1]. Calculating singular values with high accuracy requires two steps: bidiagonalization and computing the SVD of the bidiagonal matrix [5]. For bidiagonalization we will implement Golub-Kahan Bidiagonalization [5] and for computation of the singular values we will explore divide and conquer methods as well as methods using givens rotations [1]. In our analysis we will comment on which of our implementations are most favorable to cheaply calculating a sufficient number of principal components to denoise an image. Once the PCA transform is calculated, the inverse transformation is applied to obtain the final denoised image.

Competing Algorithms

- **Control:** Apply no denoising filter

- **LPG-PCA** [7]: Two-stage PCA with local pixel grouping prior to the second stage
- **Block Matching** [2]: Extracts similar image fragments into a 3D-cylinders, each of which is run through a filter and then aggregated back to the original 2D-shape
- **Auto-Encoder** [6]: Train a robust representation from the corrupted images using encoder/decoder pair; use encoder output to classify *time permitting*

Benchmark Problems

- **MNIST** 70,000 28x28 black and white images of digits; 10-way classification
- **CIFAR-10** 32x32 color images of everyday objects; 10-way classification problem
- **UMD Faces** > 300,000 relatively large images of human faces; > 8,000-way classification problem; we will likely use a reduced version with only the most represented people

Evaluation

We will evaluate the output of these various algorithms in two ways. Firstly, we will compare the denoised image to the original and use the norms of the residual as a metric to evaluate the algorithms' performance. However, since the original images themselves may be noisy, in order to evaluate the quality of the denoising with respect to the end application (classification), we will utilize each of the algorithms as a pre-processing filter for a CNN (Convolutional Neural Network) trained on each of the benchmark problems. Here, we will be use the accuracy increase of the classifier to evaluate the effectiveness of the denoising algorithms. This will be done across different severities and types (Gaussian, contrast, etc.) of noise to see how robust of a classifier can be created with each algorithm.

References

[1] Alan Kaylor Cline and Inderjit S Dhillon. Computation of the singular value decomposition. 2006.

[2] Kostadin Dabov, Alessandro Foi, Vladimir Katkovnik, and Karen Egiazarian. Image denoising with block-matching and 3d filtering. In *Image Processing: Algorithms and Systems, Neural Networks, and Machine Learning*, volume 6064, page 606414. International Society for Optics and Photonics, 2006.

[3] Paul Geladi and Bruce R. Kowalski. Partial least-squares regression: a tutorial. *Analytica Chimica Acta*, 185:1 – 17, 1986.

[4] Tim Roughgarden and Gregory Valiant. Cs168: The modern algorithmic toolbox lecture 8: How pca works. 2016.

[5] Lloyd N Trefethen and David Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.

[6] Pascal Vincent, Hugo Larochelle, Isabelle Lajoie, Yoshua Bengio, and Pierre-Antoine Manzagol. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(Dec):3371–3408, 2010.

[7] Lei Zhang, Weisheng Dong, David Zhang, and Guangming Shi. Two-stage image denoising by principal component analysis with local pixel grouping. *Pattern Recognition*, 43(4):1531–1549, 2010.