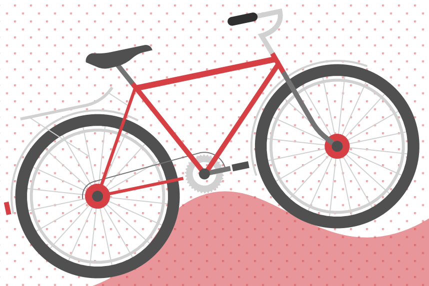
PHYSICS 301

Advanced Signal and Image Processing

ACTIVITY

03 Principal Components Analysis

for compression



Rene L. Principe Jr.

2015-04622

Dr. Maricor N. Soriano

objectives

Derive eigenvectors and their eigenvalues from image and spectral data

Convert a digital camera into a spectral images using principal components analysis

key take-aways

- Eigenvectors/eigenfaces are analogous to how a bunch of weighted sines and cosines reconstruct signals through linear superposition.
- Spectral metrics quantify facial reconstruction similarity but outliers in the dataset tend to go around these metrics.
- PCA can compress large datasets into small representative eigenvectors, effectively reducing its dimensionality.

SOURCE CODE

- Physics-301/Activity 03 - PCA for Compression .ipynb at main · reneprincipejr/Physics-301 (github.com)
- https://drive.google.com/file/d/1WPMBsni7iw4YS_oNa47vaFy5YixpR_sG/view?usp=sharing

Background

Principal Components Analysis is a useful tool to **represent a high-dimensional data into a few basis vectors** which satisfies the eigenvalues equation given by

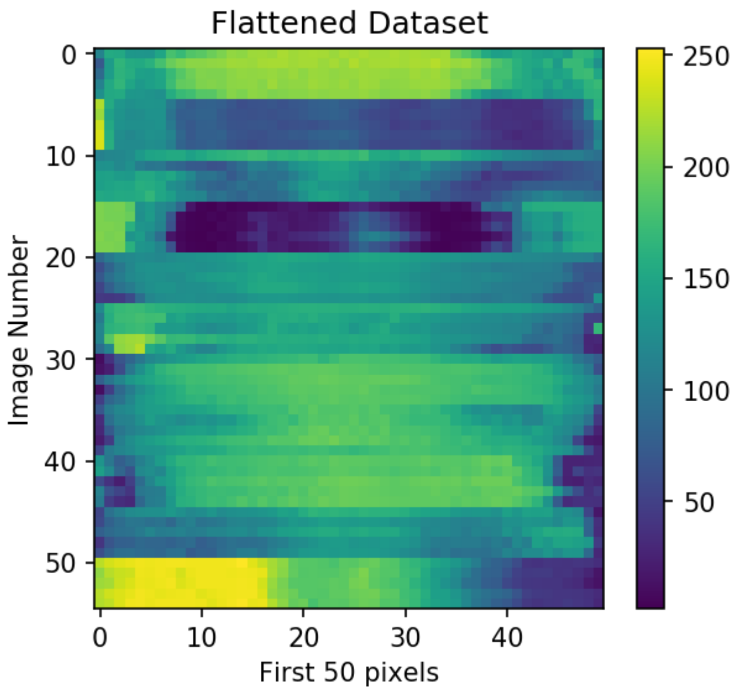
$$R\Phi_i = \lambda_i \Phi_i.$$

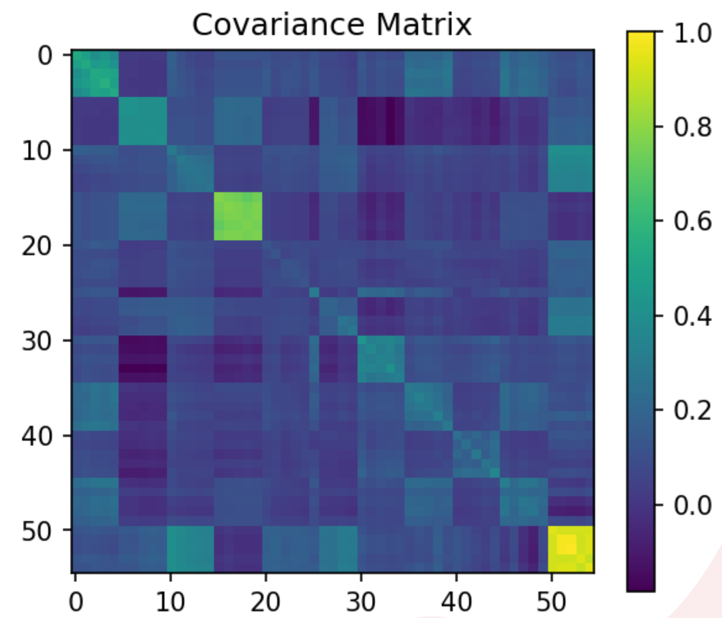
The **quality** of compression is determined by how large the cumulative sum of the first few eigenvalues can **represent the variance of the entire dataset** [1].

In this activity, the objective is to **derive eigenvectors and their eigenvalues from database of face pictures** and **qualitatively and quantitatively assess the face reconstructions attempts**.

Dataset Mosaicking

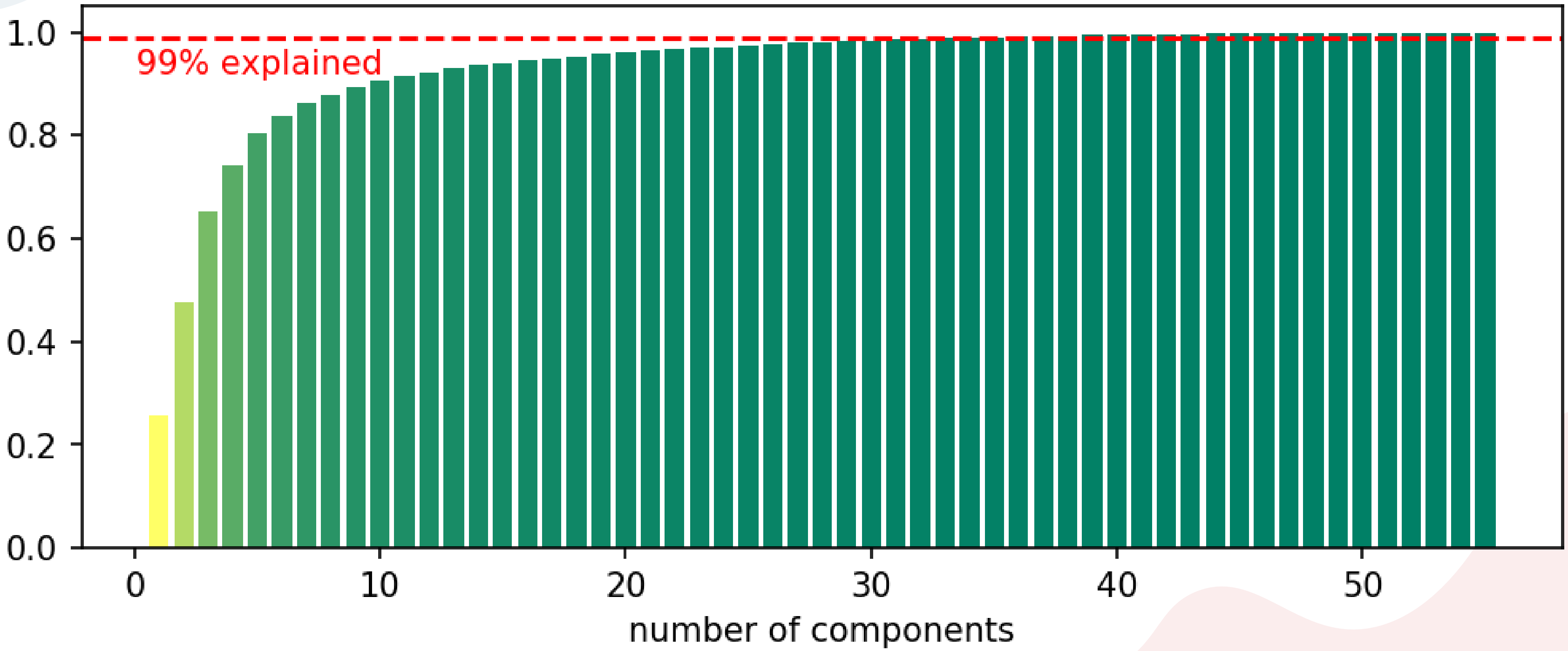
11 sets of 5 face pictures were compiled and mosaicked into a 55x2500 concatenated image. This flattened dataset and its covariance matrix are shown below. The sets are **distinguishable** given the separability on every **five-index interval along y**. Then, we use the *sklearn.decomposition* in Python to facilitate the Principal Components Analysis [2].





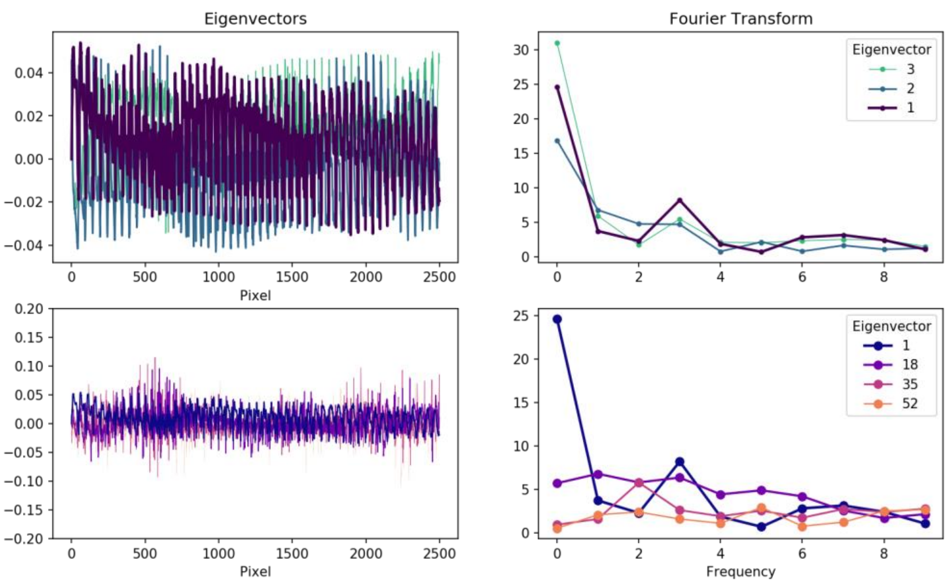
Percent Explained

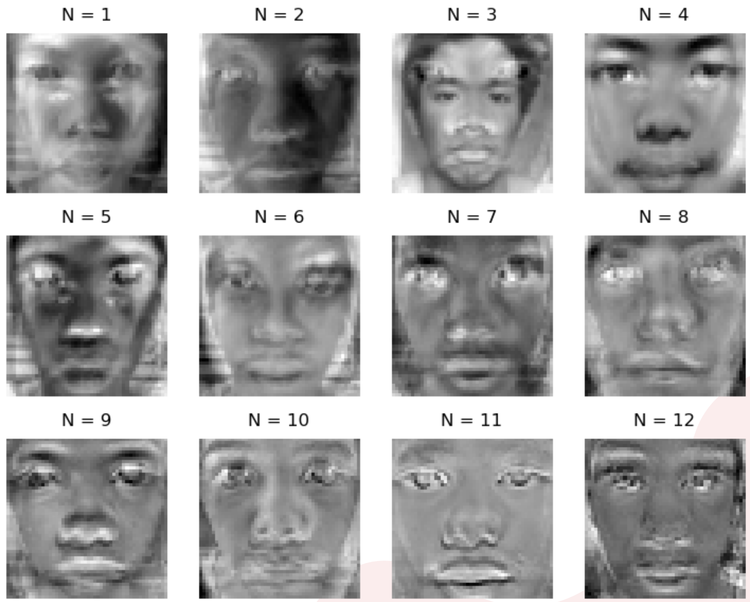
Taking the cumulative sum of the normalized eigenvalues, we can see that at around 30 principal components can represent 99% of the variance in the face database. Therefore, **a set of 2500 datapoints can be represented by 30 principal components with 1% error** in representation, hence the "**compression**".



Fourier Transform

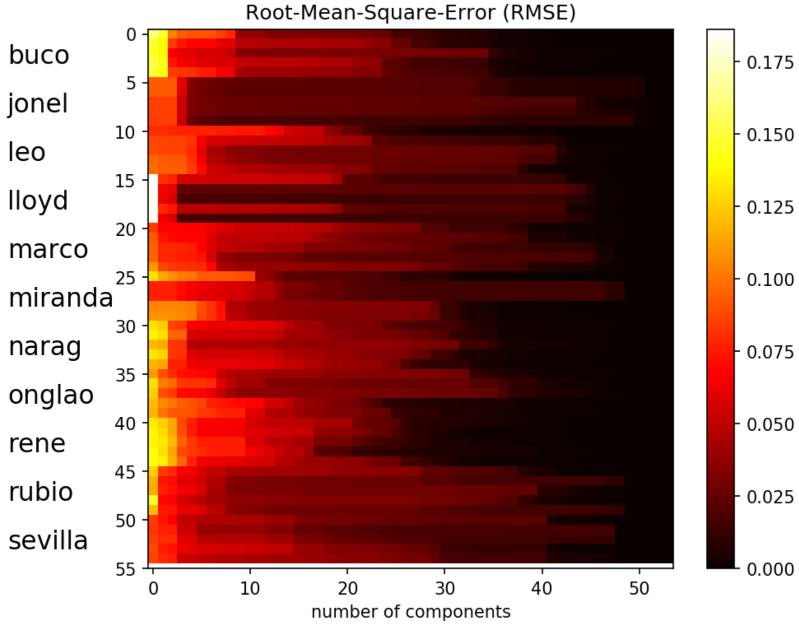
In optics, **any signal can be represented as superposition of sines and cosines of varying frequency**. I find this analogous to PCA's eigenvectors which constitutes the facial reconstruction. Taking the Fourier transform reveals that indeed, **each eigenvector had varying frequencies** [3]. Visualizing these eigenvectors into eigenfaces, combinations of these pseudo-images constitute facial reconstruction as it is just a linear superposition of these eigenfaces weighted by the calculated eigenvalues.

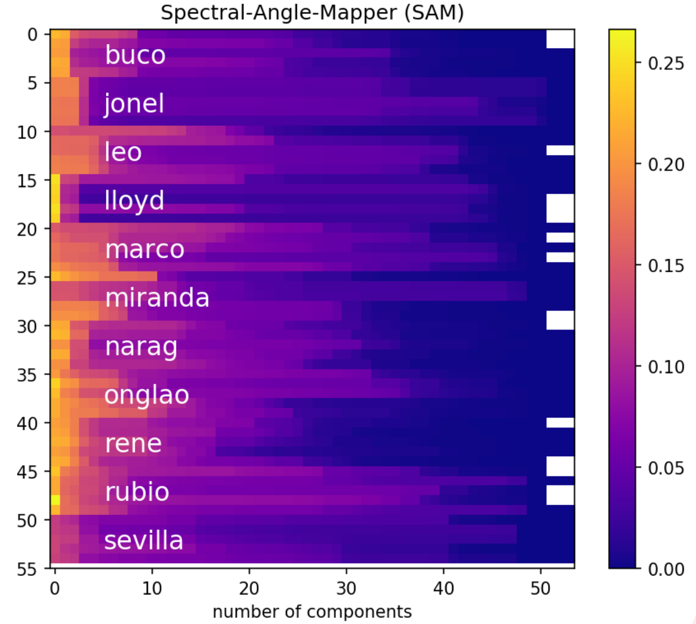




Accuracy metrics

We employed RMSE and SAM metrics which measures the residual error and structure shape similarity between the actual $\hat{r}$ and reconstructed r faces, respectively [4]. As shown below, using more eigenfaces allows us to reconstruct better as shown by the declining errors.



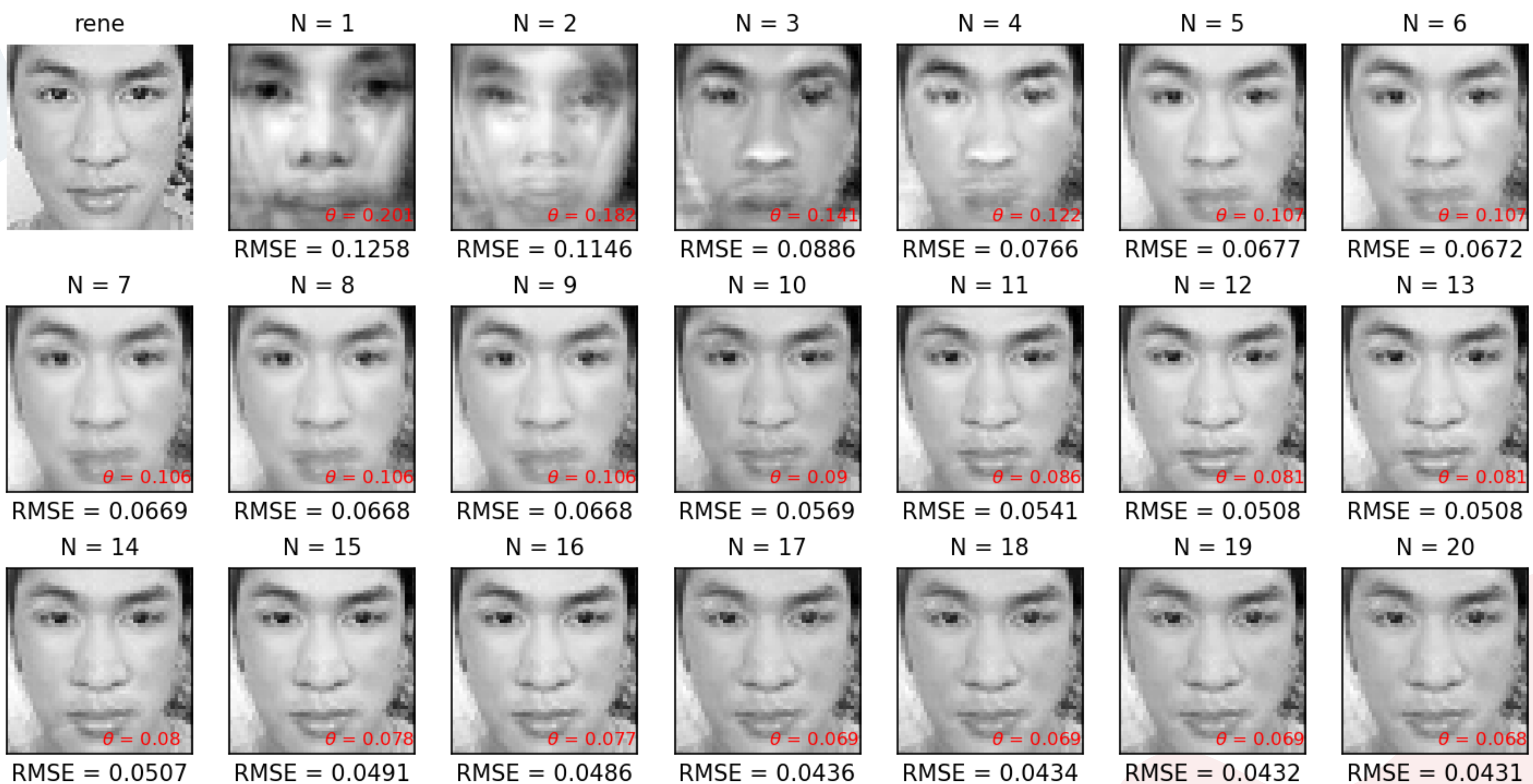


$$RMSE = \sqrt{\frac{\sum_N (r(N) - \hat{r}(N))}{n}}$$

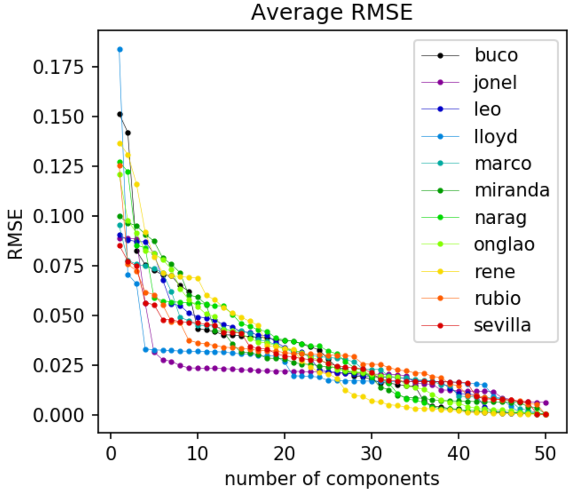
$$SAM = \cos^{-1}\left(\frac{\sum_N r(N) * \hat{r}(N)}{\sqrt{\sum_N r(N)}\sqrt{\sum_N \hat{r}(N)}}\right)$$

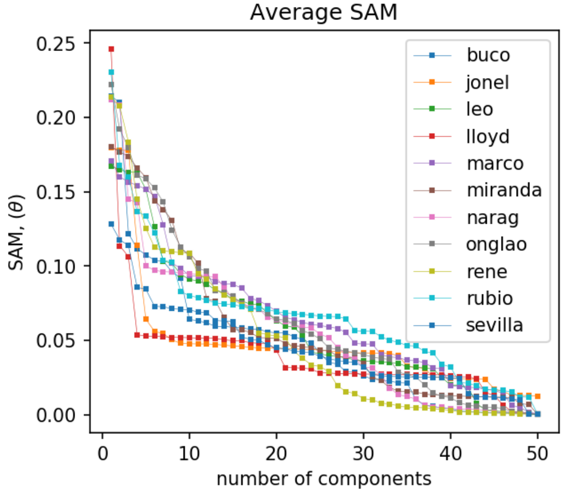
Reconstruction

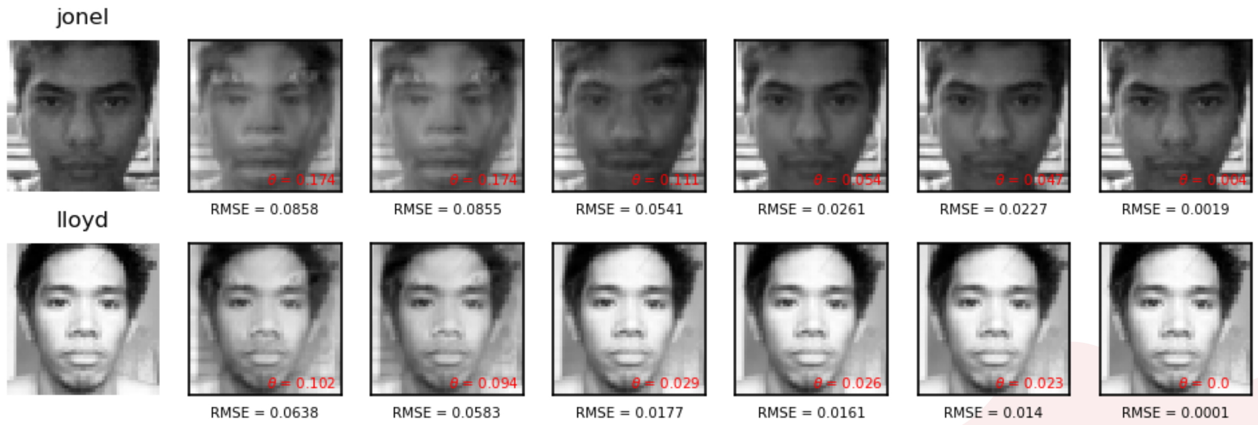
Shown here is the reconstruction of my face using increasing number of principal components. **At N = 5, the reconstruction is already recognizable with RMSE = 0.0677 and SAM = 0.107**.



Averaging the 5 facial image reconstruction of the 11 sets, the disparity in their reconstruction errors trend are mainly attributed to the quality of capture. **Jonel** and **Lloyd's** error values were low, but they are considered as **outliers** because unlike the rest of the dataset, their images contain captures of their entire head, hence **unaligned with the rest of the data**.



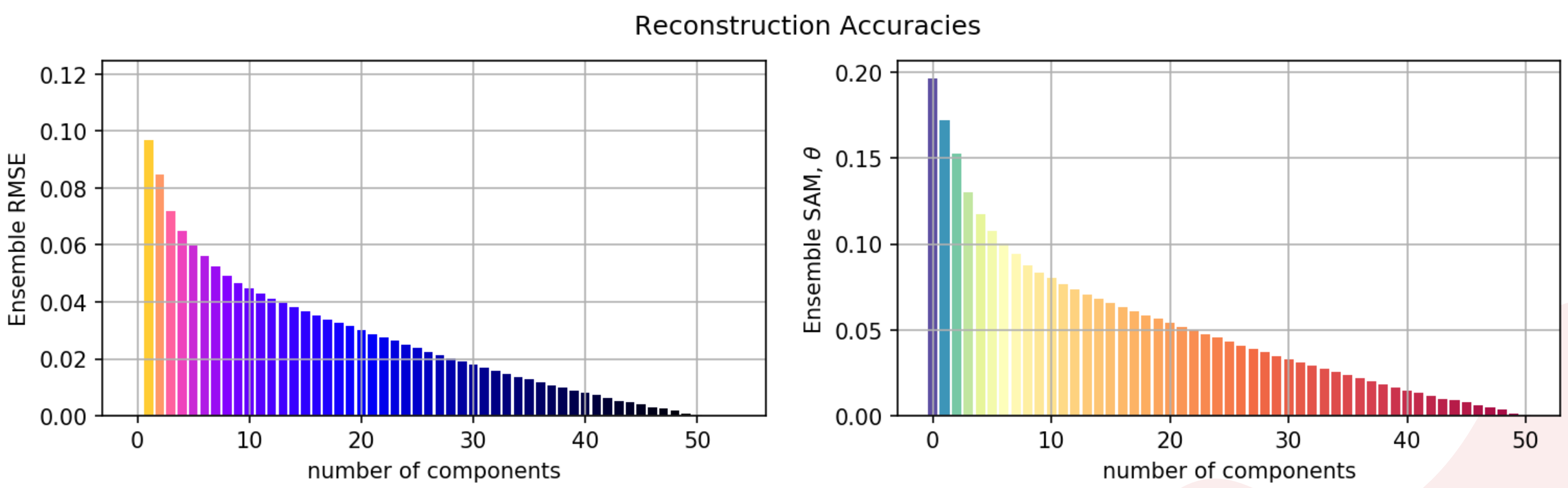


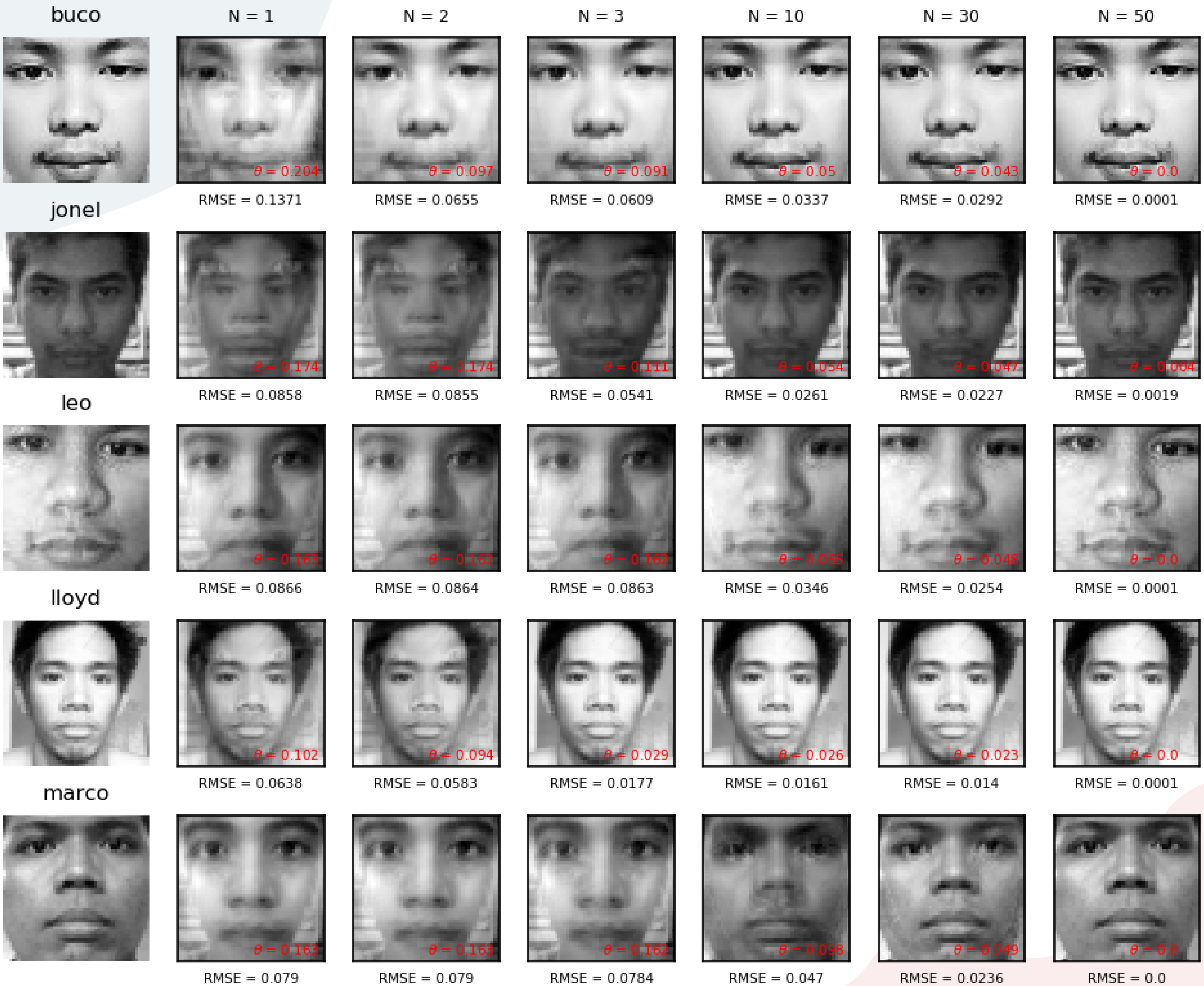


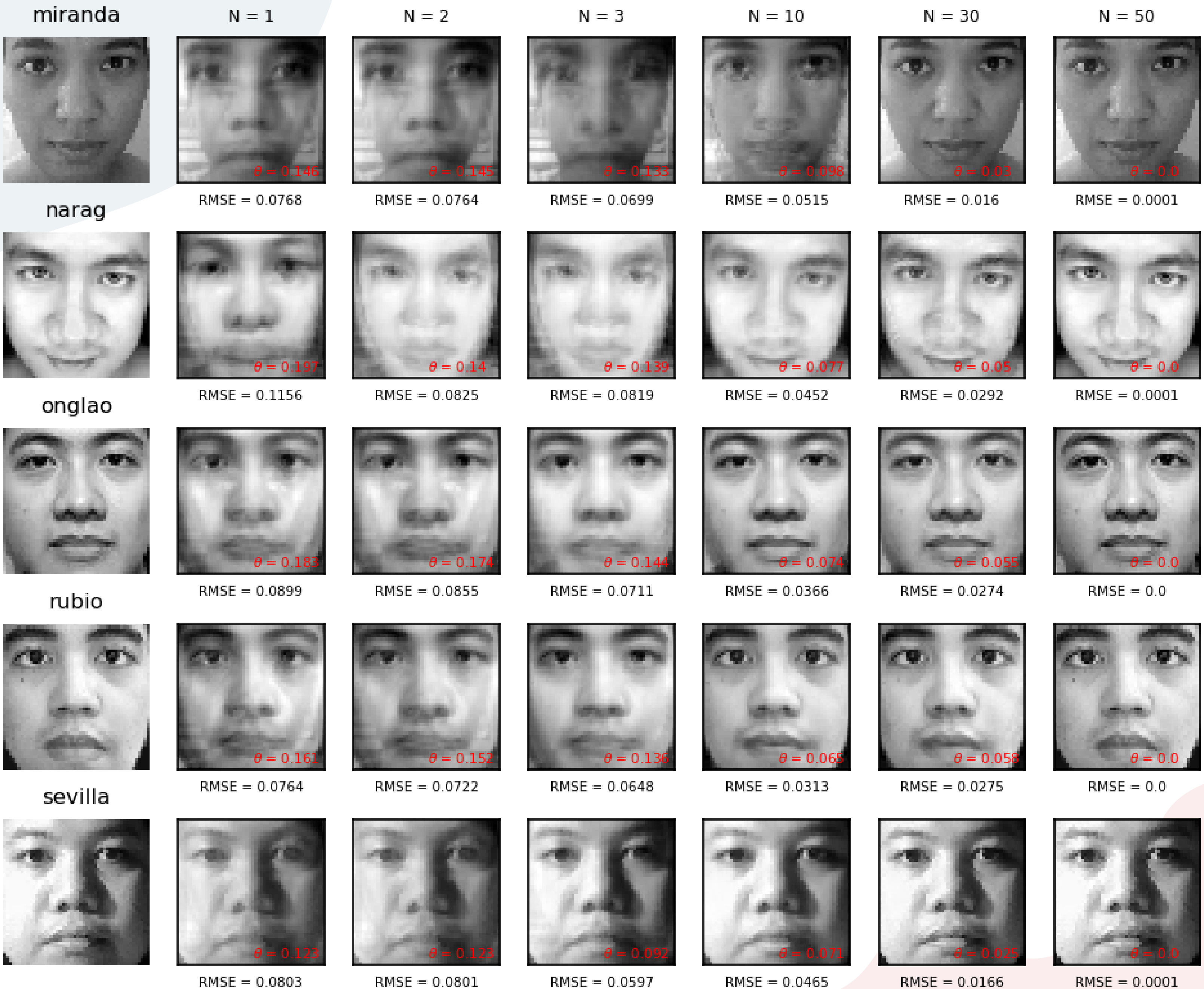
Conclusions

Overall, we demonstrated how **PCA can compress large datasets into small representative eigenvectors,** and we've shown how sufficient the reconstruction were using RMSE and SAM metrics.

Real-world data are high-dimensional in nature and in fact, compression techniques like these are useful to optimize the usage of resources need to transfer information.




buco
N = 1
N = 2
N = 3
N = 10
N = 30
N = 50
θ = 0.204
θ = 0.097
θ = 0.091
θ = 0.05
θ = 0.043
θ = 0.0
RMSE = 0.1371
RMSE = 0.0655
RMSE = 0.0609
RMSE = 0.0337
RMSE = 0.0292
RMSE = 0.0001
jonel
θ = 0.174
θ = 0.174
θ = 0.111
θ = 0.054
θ = 0.047
θ = 0.004
RMSE = 0.0858
RMSE = 0.0855
RMSE = 0.0541
RMSE = 0.0261
RMSE = 0.0227
RMSE = 0.0019
leo
θ = 0.163
θ = 0.162
θ = 0.162
θ = 0.065
θ = 0.048
θ = 0.0
RMSE = 0.0866
RMSE = 0.0864
RMSE = 0.0863
RMSE = 0.0346
RMSE = 0.0254
RMSE = 0.0001
lloyd
θ = 0.102
θ = 0.094
θ = 0.029
θ = 0.026
θ = 0.023
θ = 0.0
RMSE = 0.0638
RMSE = 0.0583
RMSE = 0.0177
RMSE = 0.0161
RMSE = 0.014
RMSE = 0.0001
marco
θ = 0.163
θ = 0.163
θ = 0.162
θ = 0.098
θ = 0.049
θ = 0.0
RMSE = 0.079
RMSE = 0.079
RMSE = 0.0784
RMSE = 0.047
RMSE = 0.0236
RMSE = 0.0


miranda
N = 1
N = 2
N = 3
N = 10
N = 30
N = 50
θ = 0.146
θ = 0.145
θ = 0.133
θ = 0.098
θ = 0.03
θ = 0.0
RMSE = 0.0768
RMSE = 0.0764
RMSE = 0.0699
RMSE = 0.0515
RMSE = 0.016
RMSE = 0.0001
narag
θ = 0.197
θ = 0.14
θ = 0.139
θ = 0.077
θ = 0.05
θ = 0.0
RMSE = 0.1156
RMSE = 0.0825
RMSE = 0.0819
RMSE = 0.0452
RMSE = 0.0292
RMSE = 0.0001
onglao
θ = 0.183
θ = 0.174
θ = 0.144
θ = 0.074
θ = 0.055
θ = 0.0
RMSE = 0.0899
RMSE = 0.0855
RMSE = 0.0711
RMSE = 0.0366
RMSE = 0.0274
RMSE = 0.0
rubio
θ = 0.161
θ = 0.152
θ = 0.136
θ = 0.065
θ = 0.058
θ = 0.0
RMSE = 0.0764
RMSE = 0.0722
RMSE = 0.0648
RMSE = 0.0313
RMSE = 0.0275
RMSE = 0.0
sevilla
θ = 0.123
θ = 0.123
θ = 0.092
θ = 0.071
θ = 0.025
θ = 0.0
RMSE = 0.0803
RMSE = 0.0801
RMSE = 0.0597
RMSE = 0.0465
RMSE = 0.0166
RMSE = 0.0001

reflection

In my undergraduate skill building training, Dr. Soriano has guided me step-by-step in understanding how Principal Components Analysis works and so, I'd like to give her credit for my smooth execution of this activity. In fact, before I used machine learning, PCA is a state-of-the-art regression technique which I used in reconstructing spectra (which I'll do in Activity 4). This activity one of the most exciting because we got to work with actual images. I'd like to thank my classmates for providing the face dataset that was used in this activity. More than the technical skillset satisfaction, I'm elated with the realization as to how everything is very much related to each other. Imagine reconstructing a face with just 5 inputs! Out of excitement, I think I went beyond as I measured the eigenvector frequencies, quantified the reconstruction accuracies, discussed the outlier results, and provided sufficient visualization.

In this activity, I'd give myself a score of **100/100.**

references

- [1] M. Soriano, Physics 301 – Principal Components Analysis for Compression, (2022).
- [2] sklearn.decomposition.PCA — scikit-learn 1.1.1 documentation
- [3] Discrete Fourier Transform (numpy.fft) — NumPy v1.22 Manual
- [4] P. E. Dennison, K. Q. Halligan, and D. A. Roberts, A comparison of error metrics and constraints for multiple endmember spectral mixture analysis and spectral angle mapper, Remote Sens. Environ. 93, 359 (2004).