Assignment 4: CS 215

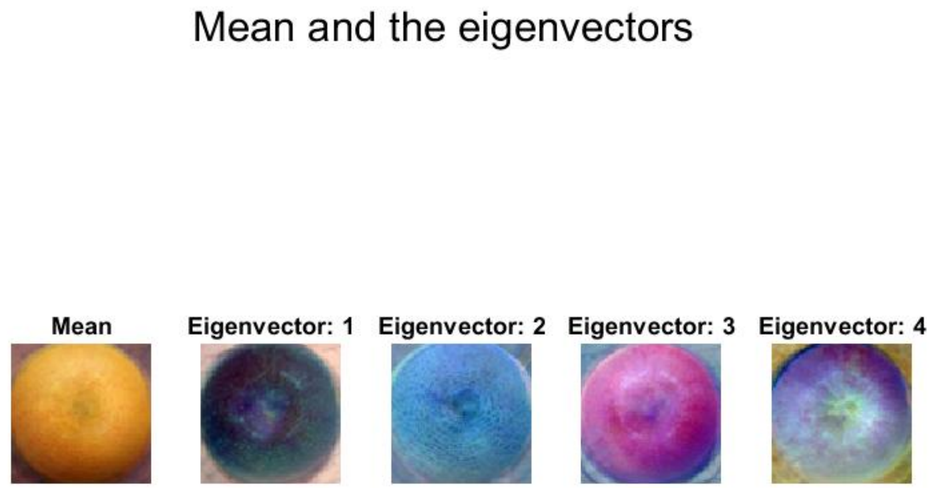
Devansh Jain	Harshit Varma
190100044	190100055

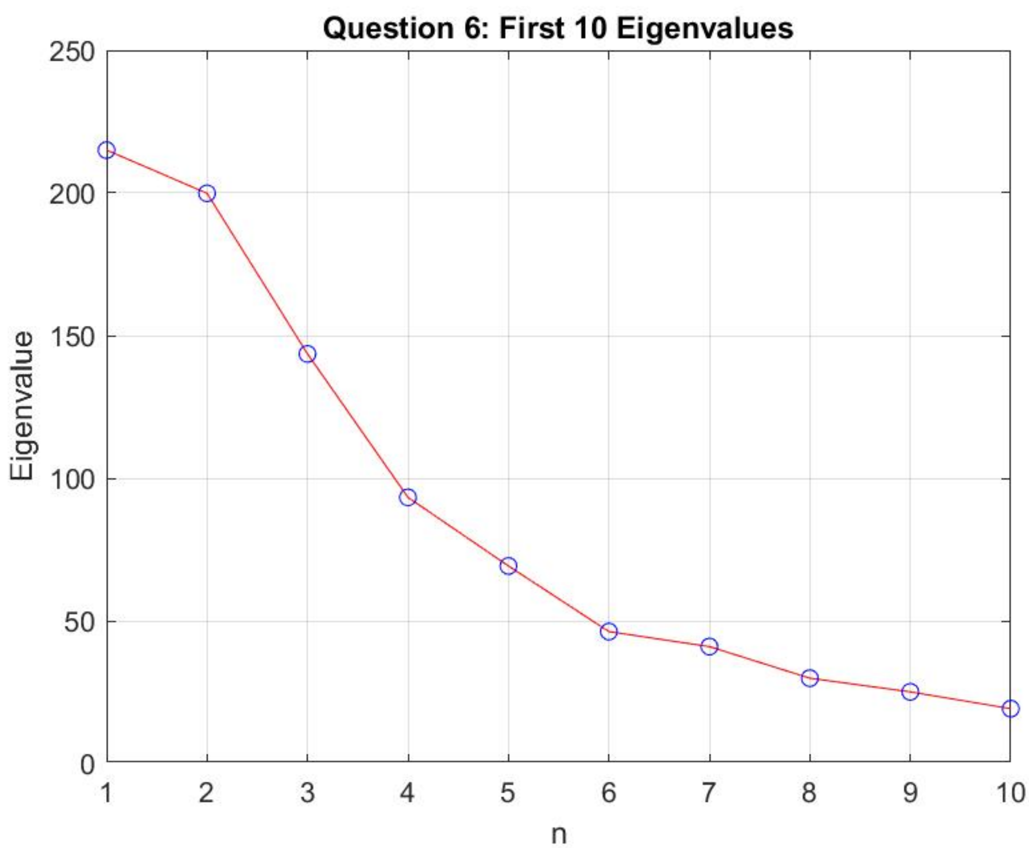
November 13, 2020

Question 6

Instructions for running the code:

1. Unzip and `cd` to `q6/code`, under this find the file named `q6.m`
2. Run the file, required plots will be generated and saved to the `q6/results` folder
   `q6i.jpg`: Mean and the 4 eigenvectors
   `q6ii.jpg`: Top 10 eigenvalues
   `comparison_i.jpg`: Comparison for i^{th} image
   `generated_fruits.jpg`: 3 generated images of the fruits


Mean and the eigenvectors
Mean
Eigenvector: 1
Eigenvector: 2
Eigenvector: 3
Eigenvector: 4


Question 6: First 10 Eigenvalues
Eigenvalue
n
0
50
100
150
200
250
1
2
3
4
5
6
7
8
9
10

Closest Representation:

Let $d = 80 \times 80 \times 3$
Let the sample mean of the original data be $\mu \in \mathbb{R}^d$, the sample covariance matrix of the original data be $C \in \mathbb{R}^{d\times d}$
Let the 4 chosen (normalized) eigenvectors of C be $e_1, e_2, e_3, e_4 \in \mathbb{R}^d$
Let the original images (uncentered) be $\{x_i\}_{i=1}^N$, in this case $N = 16$
Then, the projection of x_i onto the new dimensions will be given by $\sum_{j=1}^4((x_i - \mu)\cdot e_j)e_j$.
Now, since the projections are in a mean centered space, to come back into the uncentred space, we need to add μ to get the reconstruction of the original image:

$$\boxed{r_i = \mu + \sum_{j=1}^{4}((x_i - \mu)\cdot e_j)e_j}$$

Thus, the coefficients used in the linear combination are $\boxed{c_{ij} = (x_i - \mu)\cdot e_j}$, now we need to prove that these coefficients indeed minimize the frobenius norm, which is equivalent to minimizing the square of the frobenius norm.

For the i^{th} unrolled image, square of the frobinius norm is given by:

$$f_i^2 = ||x_i - r_i||^2$$

Note that x_i is **exactly** equal to $\mu + \sum_{j=1}^d((x_i - \mu)\cdot e_j)e_j$ (Sum of projections over **all** the eigenvectors of C), thus:

$$\begin{aligned}
f_i^2 &= \left|\left|\mu + \sum_{j=1}^{d}((x_i - \mu)\cdot e_j)e_j - (\mu + \sum_{j=1}^{4}((x_i - \mu)\cdot e_j)e_j)\right|\right|^2 \\
&= \left|\left|\sum_{j=1}^{d}((x_i - \mu)\cdot e_j)e_j - \sum_{j=1}^{4}((x_i - \mu)\cdot e_j)e_j\right|\right|^2 \\
&= \left|\left|\sum_{j=5}^{d}((x_i - \mu)\cdot e_j)e_j\right|\right|^2 = \left|\left|\sum_{j=5}^{d} c_{ij}e_j\right|\right|^2
\end{aligned}$$

Since any pair of e_js are orthogonal, and e_j are unit vectors,

$$f_i^2 = \left|\left|\sum_{j=5}^{d} c_{ij}e_j\right|\right|^2 = \sum_{j=5}^{d} c_{ij}^2 = \sum_{j=5}^{d}((x_i - \mu)\cdot e_j)^2$$

Thus we need to **minimize** $\sum_{j=5}^d((x_i - \mu)\cdot e_j)^2$
Now, the variance of the projected data which PCA **maximizes** is $\sum_{j=1}^4((x_i - \mu)\cdot e_j)^2$

Since,

$$x_i = \mu + \sum_{j=1}^{d}((x_i - \mu)\cdot e_j)e_j$$

$$(x_i - \mu)^2 = (\sum_{j=1}^{d}((x_i - \mu)\cdot e_j)e_j)^2$$

Since any pair of e_js are orthogonal, and e_j are unit vectors,

$$= \sum_{j=1}^{d}((x_i - \mu)\cdot e_j)^2 = \sum_{j=1}^{4}((x_i - \mu)\cdot e_j)^2 + \sum_{j=5}^{d}((x_i - \mu)\cdot e_j)^2$$

$$\boxed{(x_i - \mu)^2 = constant = \sum_{j=1}^{4}((x_i - \mu)\cdot e_j)^2 + \sum_{j=5}^{d}((x_i - \mu)\cdot e_j)^2}$$

Thus, **maximizing** the variance of the projected data: $\sum_{j=1}^{4}((x_i - \mu)\cdot e_j)^2$ is same as **minimizing** the frobenius norm of the difference: $\sum_{j=5}^{d}((x_i - \mu)\cdot e_j)^2 = f_i^2$.

Thus, in effect, PCA minimizes the frobenius norm of the difference between the projected and the actual image.
We follow the same way to reconstruct the images in our algorithm.
Let $E = [e_1\ e_2\ e_3\ e_4]^T \in \mathbb{R}^{d\times 4}$, let $X \in \mathbb{R}^{d\times 16}$ contain the mean centered data, then $A = X^T E \in \mathbb{R}^{16\times 4}$, where $A_{ij} = c_{ij}$, and thus the reconstructed data $\hat{X} = \mu + EA^T \in \mathbb{R}^{d\times 16}$, here μ is added to each column of EA^T.
In MATLAB:

```
% data is the mean centered data
% Q4 contains the 4 eigenvectors, coeffs contain the required coefficients
coeffs = data' * Q4; % 16x4
data_reconstructed = mu + Q4*coeffs';
```

We obtain the following reconstructions:

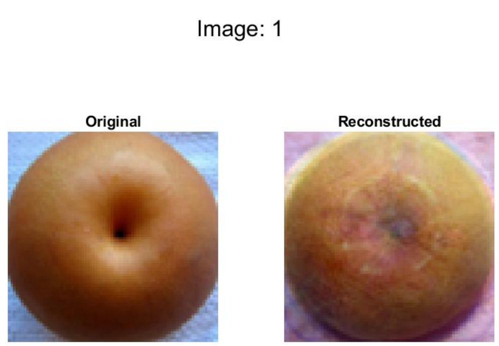


Image: 3

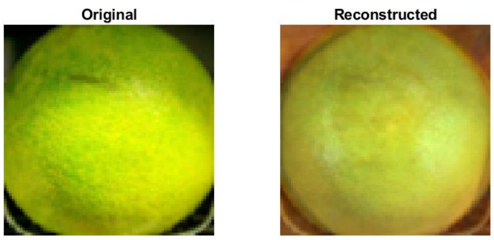


Image: 4

Image: 5

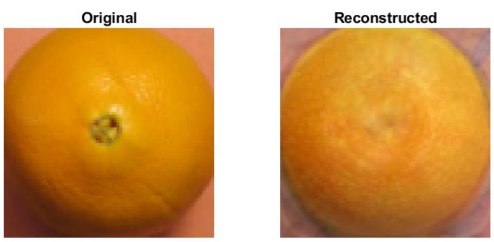


Image: 6

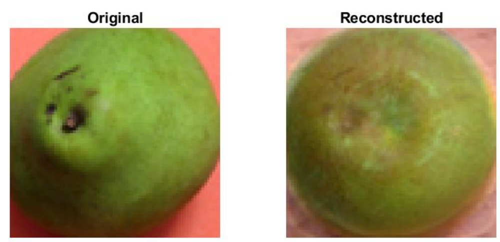


Image: 7

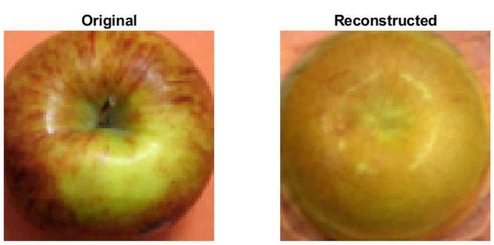


Image: 8

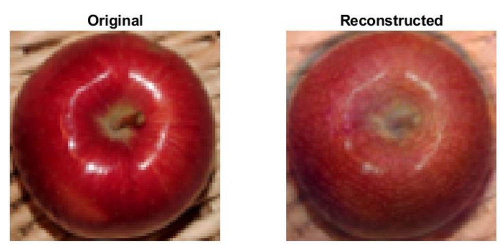


Image: 9

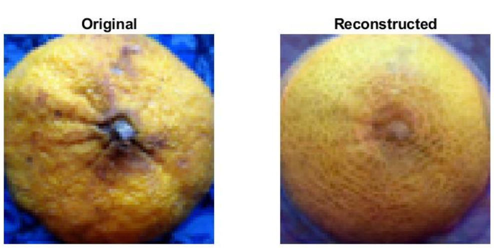


Image: 10

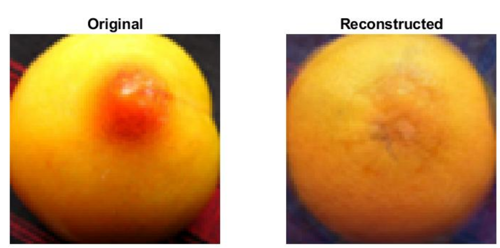


Image: 11

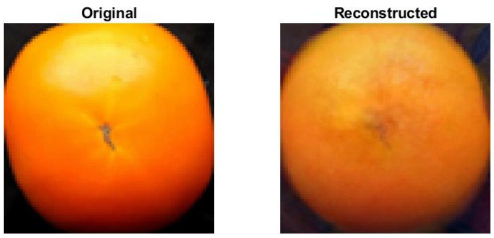


Image: 12

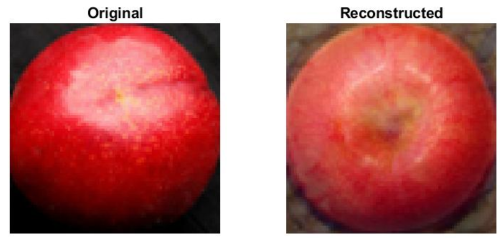


Image: 13

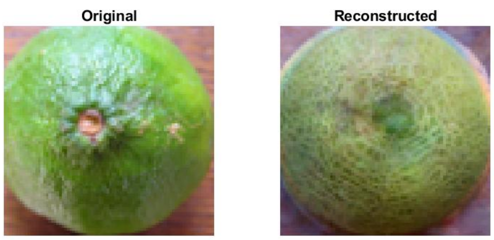


Image: 14

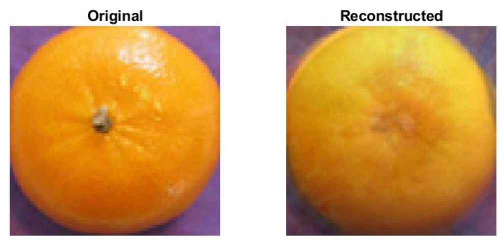


Image: 15

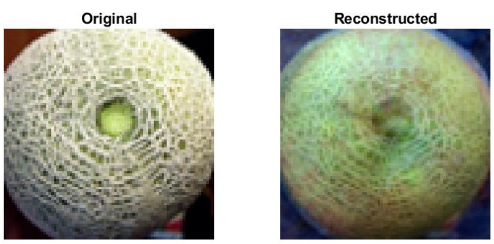
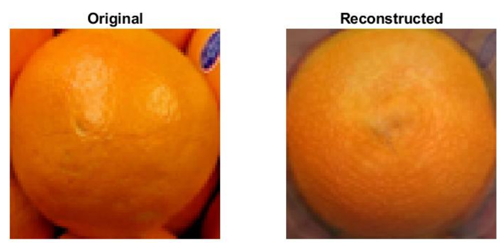


Image: 16



Generating new images:

For generating new images, we need to sample data in the reduced domain and then project it back into the original domain.
In the projected domain, each point looks like $\mathbf{c} = [c_1\ c_2\ c_3\ c_4]^T \in \mathbb{R}^4$, to project this back into the original domain ($\mathbb{R}^d$):

$$x_{gen} = \mu + \sum_{j=1}^{4} c_j \mathbf{e_j}$$

Thus, we need to sample $\mathbf{c}$.
We assume that $c \sim \mathcal{N}(\mathbf{0}, C)$ (A 0-mean multivariate gaussian with covariance matrix $C \in \mathbb{R}^{4\times 4}$)
We take $C = diag([\lambda_1, \lambda_2, \lambda_3, \lambda_4])$, since the variance of the original data along the chosen eigenvectors is given by the corresponding eigenvalues.
Now we sample random vectors $\mathbf{c}$ from this multivariate gaussian, project back to the original space and plot the images.
We achieve the following results which are different, but representative of the dataset:

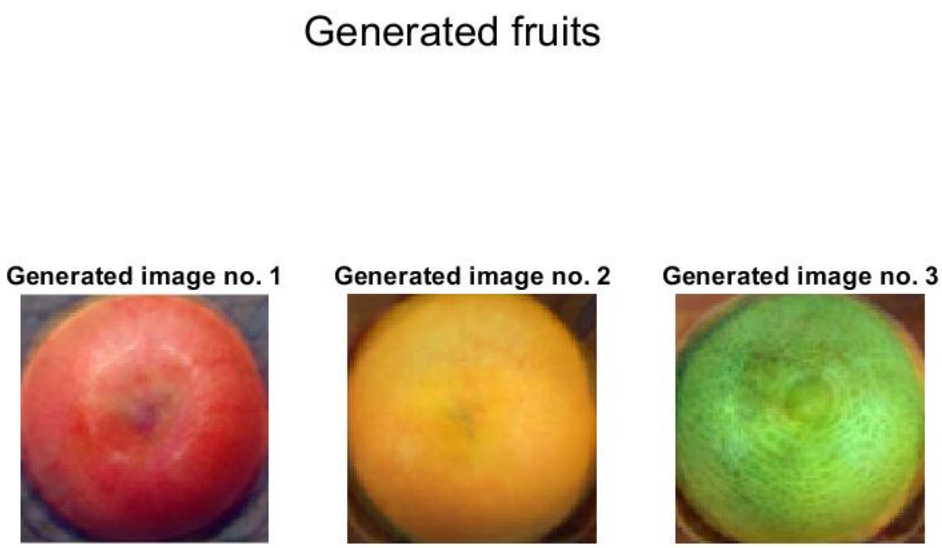


Figure 1: Generated images