Pattern Recognition

CS6690

IIT Madras

Assignment 1

By:
Group 25:
Arun Baby (CS15S016)
Vishal Subbiah (MM12B035)

Problem 1

Image 1

Figure 1: Original Image

Singular value decomposition:

In these six experiments we performed SVD on the RGB channels individually as well as combining them to form a 24-bit image. We then reconstructed the RGB using N singular values. We then reconstructed the individual channels in to a single image and checked for the error as seen in the graphs.

$$[U, S, V] = svd(I) \tag{1}$$

$$I_{new} = U * S * V' \tag{2}$$

Experiment 1 In experiment 1 we took the top N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

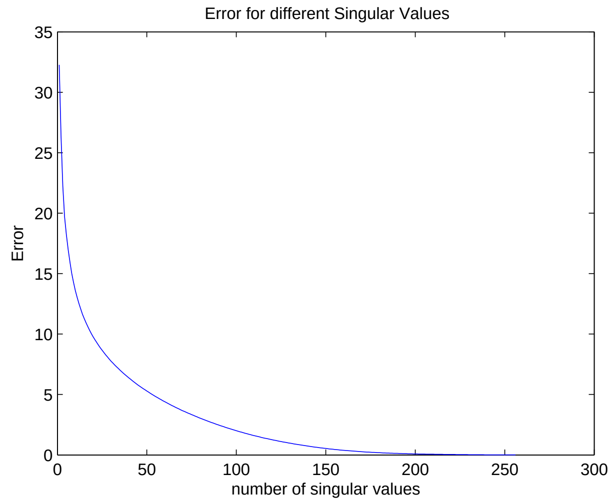

Figure 2: Error plot for experiment 1

As we can see, the error reduces significantly as we increase the number of singular values. The graph shows the corresponding errors from 1 to 256 singular values.

As seen we do not need all the singular values to obtain an image with less error. With around 25 singular values we are able to obtain a very low error and hence the image is compressed as we no longer need 256 singular values for the image. In SVD the top singular values affect the image the most. Their value decreases as we progress along the diagonal matrix. We used root mean square to find the error.

Experiment 2 In experiment 2 we took the bottom N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

As we increase the number of singular values, the increase in quality is initially negligible as the lower values are least significant.It reaches a critical point where the value of the singular values is noticeable to the image and hence we see sharp drop in the error.

Experiment 3 In experiment 3 we took the random N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

In this experiment we can see the error and quality of the image fluctuating as the singular values are selected in random. When the top singular values are selected, the error is lower while when the bottom values are selected, the error is very high.

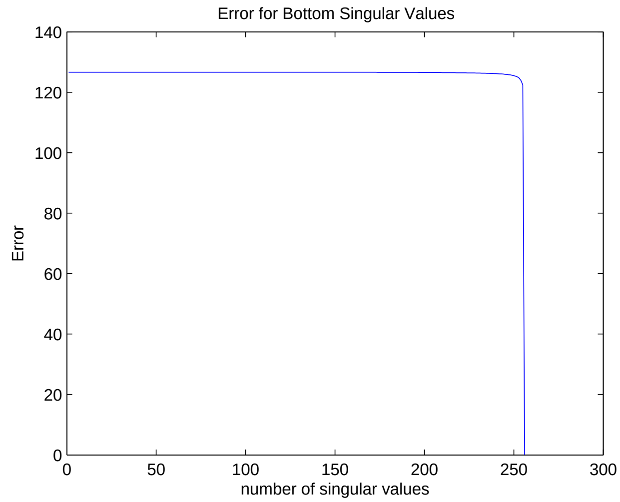

Figure 3: Error Plot for experiment 2

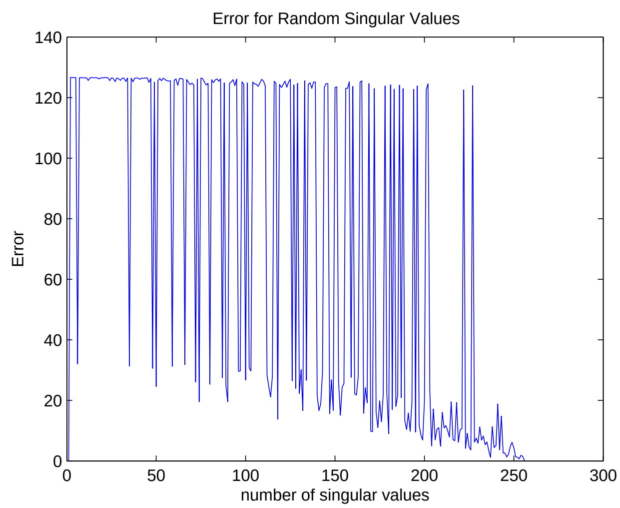

Figure 4: Error Plot for experiment 3

Experiment 4 In experiment 4 we converted to 24 bit and took the top N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

In this experiment, we see similar results as seen in experiment 1. The errors are a little higher but the advantage is that we only do SVD on the image once as we have converted it to a 24-bit image instead of individually for the red, green and blue channels.

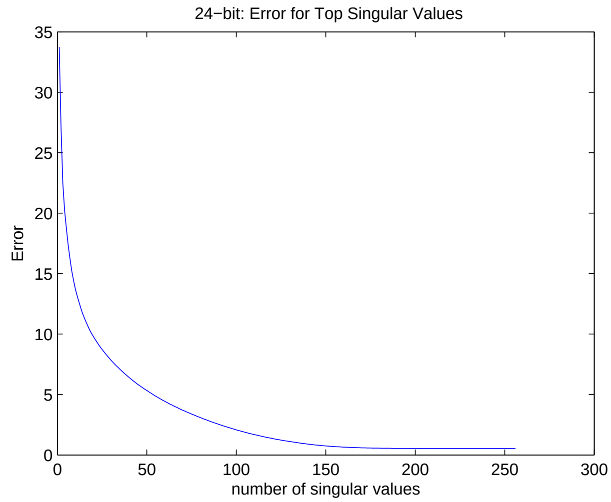

Figure 5: Error plot for experiment 4

Experiment 5 In experiment 5 we converted to 24 bit and took the bottom N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

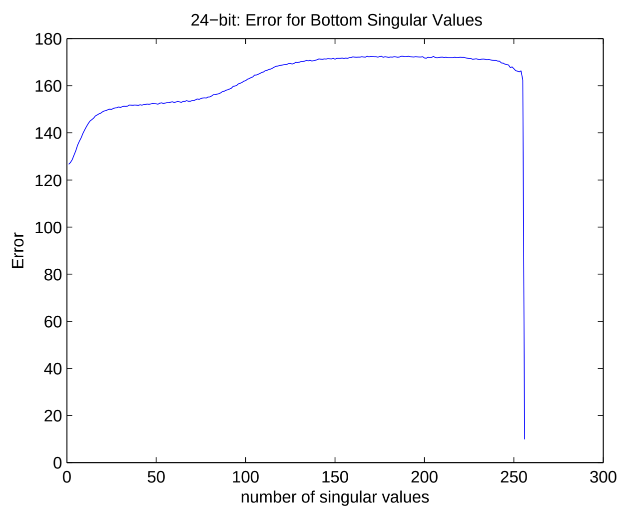

Figure 6: Error plot for experiment 5

In this experiment we see the results to be similar to experiment 2 but with the advantages and disadvantages of experiment 4.

Experiment 6 In experiment 6 we converted to 24 bit and took random N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

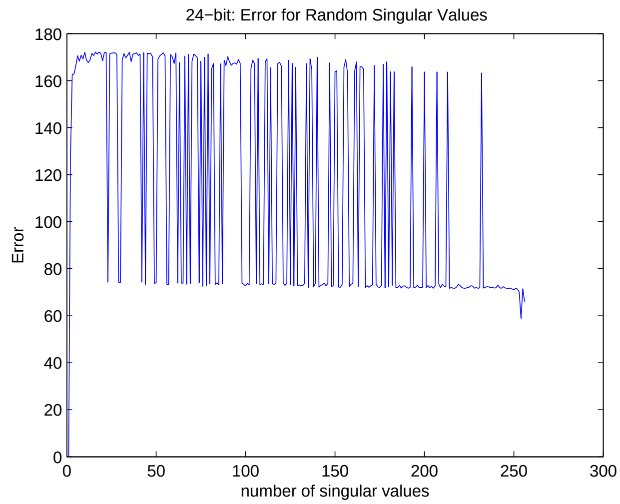

Figure 7: Error plot for experiment 6

Though the results are similar to experiment 3, we see the error is much higher than in experiment 3.

Also, in experiments 4,5 and 6 we see a dominant colour (red or blue or green) during the initial images while the another dominant colour in the error image(red or blue or green). We do not observe this in experiments 1,2 and 3.

Eigenvalue decomposition:

In these six experiments we performed eigenvalue decomposition on the RGB channels individually as well as combining them to form a 24-bit image. We then reconstructed the RGB using N eigenvalues. We then reconstructed the individual channels in to a single image and checked for the error as seen in the graphs.

$$[V, D] = eig(I) \tag{3}$$

$$I_{new} = V * D * V^{-1} \tag{4}$$

Experiment 1 In experiment 1 we took the top N eigen vectors and generated the image, the error image and calculated the error between the generated and original image.

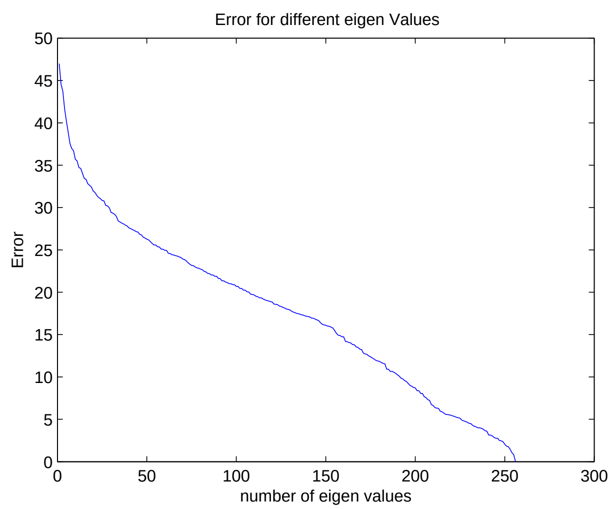

Figure 8: Error plot for experiment 1

As we can see, the error reduces as we increase the number of singular values but not as significantly as singular values. The graph shows the corresponding errors from 1 to 256 eigenvalues.

As seen we do not need all the eigenvalues to obtain an image with less error. With around 25 eigenvalues we are able to recognize the image and hence the image is compressed as we no longer need 256 eigenvalues for the image. In eigenvalues are not sorted in the descending order as in SVD, hence we sorted the eigenvalues and their corresponding vectors to obtain better results. We used root mean square to find the error.

Experiment 2 In experiment 2 we took the bottom N eigen vectors and generated the image, the error image and calculated the error between the generated and original image.

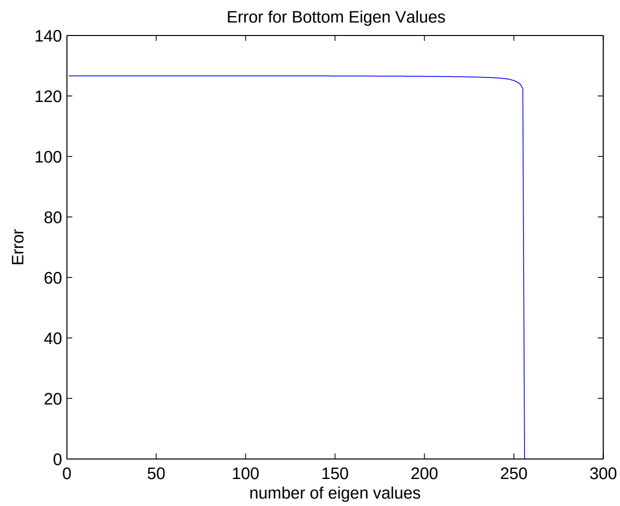

Figure 9: Error plot for experiment 2

As we increase the number of eigenvalues, the increase in quality is initially negligible as the lower values are least significant. It reaches a critical point where the value of the eigenvalues is noticeable to the image and hence we see sharp drop in the error.

Experiment 3 In experiment 3 we took the random N eigen vectors and generated the image, the error image and calculated the error between the generated and original image.

In this experiment we can see the error and quality of the image fluctuating as the eigenvalues are selected in random. When the top eigenvalues are selected, the error is lower while when the bottom values are selected, the error is very high.

Experiment 4 In experiment 4 we converted to 24 bit and took the top N eigen vectors and generated the image, the error image and calculated the error between the generated and original image.

In this experiment, we see similar results as seen in experiment 1. The errors are a little higher but the advantage is that we only do eigenvalue decomposition on the image once as we have converted it to a 24-bit image instead of individually for the red, green and blue channels.

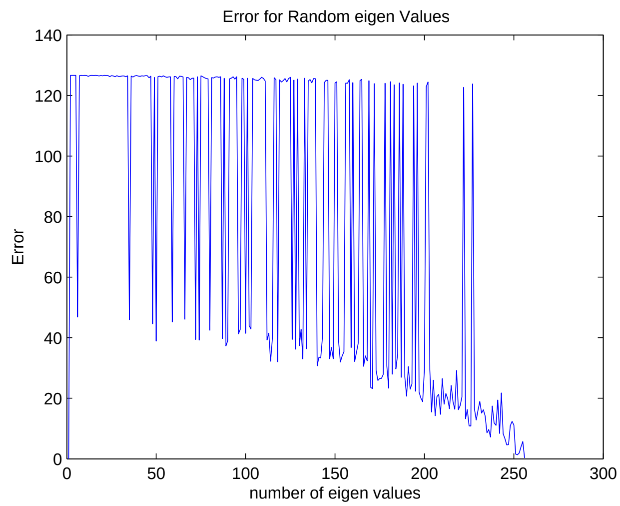

Figure 10: Error plot for experiment 3

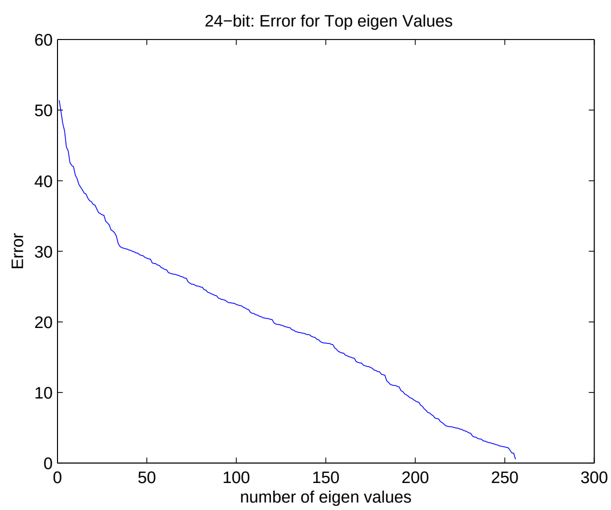

Figure 11: Error plot for experiment 4

Experiment 5 In experiment 5 we converted to 24 bit and took the bottom N eigen vectors and generated the image, the error image and calculated the error between the generated and original image.

In this experiment we see the results to be similar to experiment 2 but with the advantages and disadvantages of experiment 4.

Experiment 6 In experiment 6 we converted to 24 bit and took random N eigen vectors and generated the image, the error image and calculated the error between

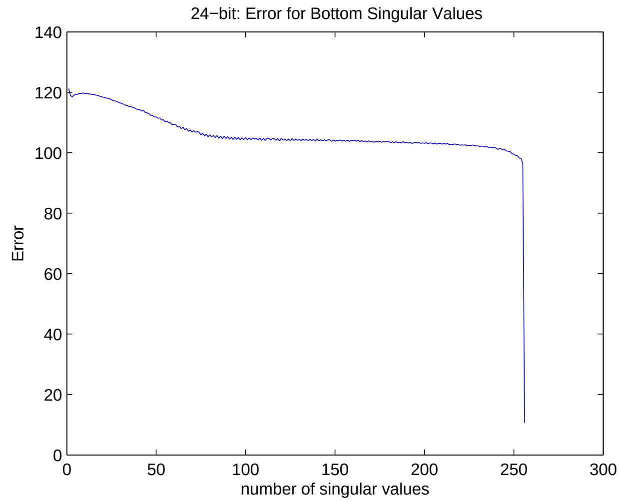

Figure 12: Error plot for experiment 5

the generated and original image.

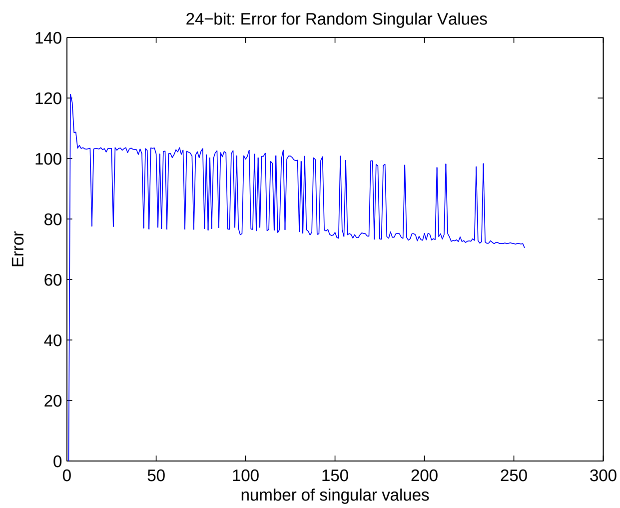

Figure 13: Error plot for experiment 6

Though the results are similar to experiment 3, we see the error is much higher than in experiment 3.

Also, in experiments 4,5 and 6 we see a dominant colour (red or blue or green) during the initial images while the another dominant colour in the error image(blue or green or red). We do not observe this in experiments 1,2 and 3.

Image 2

As seen image 2 is rectangular rather than square.

Figure 14: Original Image

Singular value decomposition:

For singular value decomposition, the procedure is the same as the one followed in image 1. Unlike image 1, the size of image is significantly larger and hence we could not take all possible N (min(row,column)). Instead we took upto 25 singular values.

Experiment 1 In experiment 1 we took the top N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

As we can see, the error reduces significantly as we increase the number of singular values. The graph shows the corresponding errors from 1 to 256 singular values.

As seen we do not need all the singular values to obtain an image with less error. With less than 40 singular values we are able to obtain a very low error and hence the image is compressed as we no longer need all the singular values for the image. In SVD the top singular values affect the image the most. Their value decreases as we progress along the diagonal matrix. We used root mean square to find the error.

Experiment 2 In experiment 2 we took the bottom N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

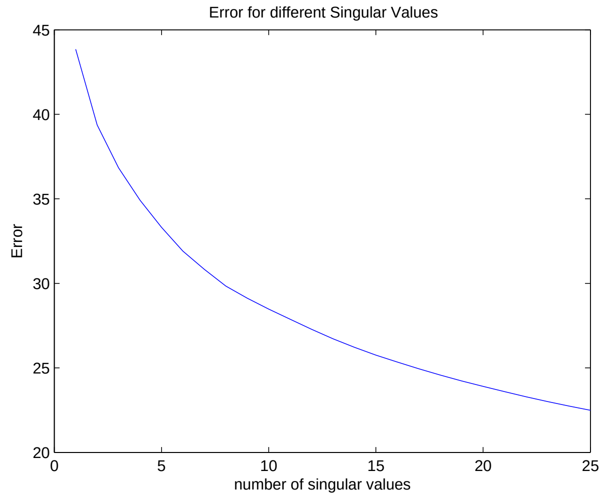

Figure 15: Error plot for experiment 1

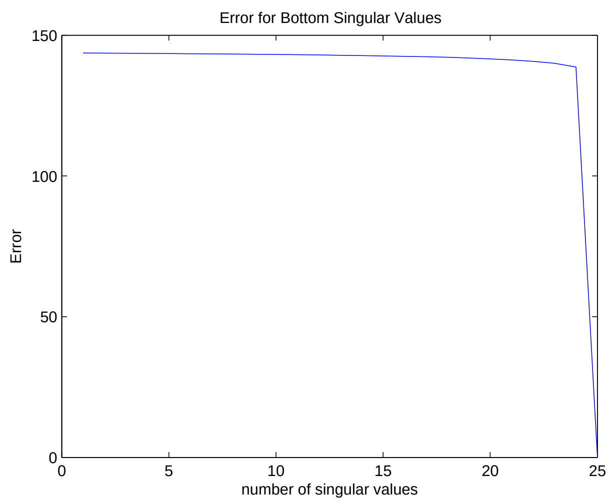

Figure 16: Error plot for experiment 2

As we increase the number of singular values, the increase in quality is initially negligible as the lower values are least significant.It reaches a critical point where the value of the singular values is noticeable to the image and hence we see sharp drop in the error.

Experiment 3 In experiment 3 we took the random N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

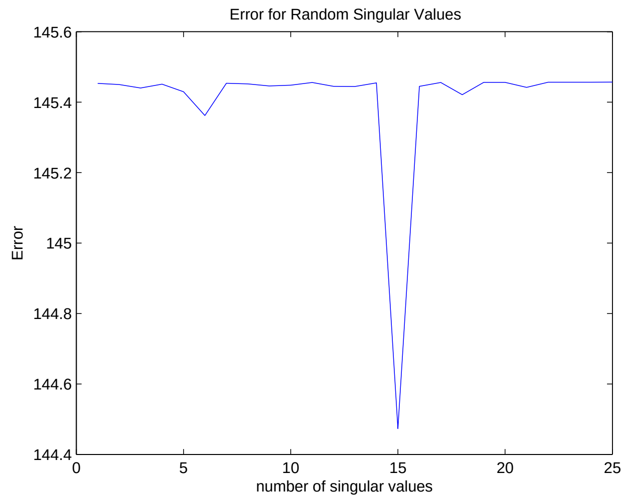

Figure 17: Error plot for experiment 3

In this experiment we can see the error and quality of the image fluctuating as the singular values are selected in random. When the top singular values are selected, the error is lower while when the bottom values are selected, the error is very high.

Experiment 4 In experiment 4 we converted to 24 bit and took the top N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

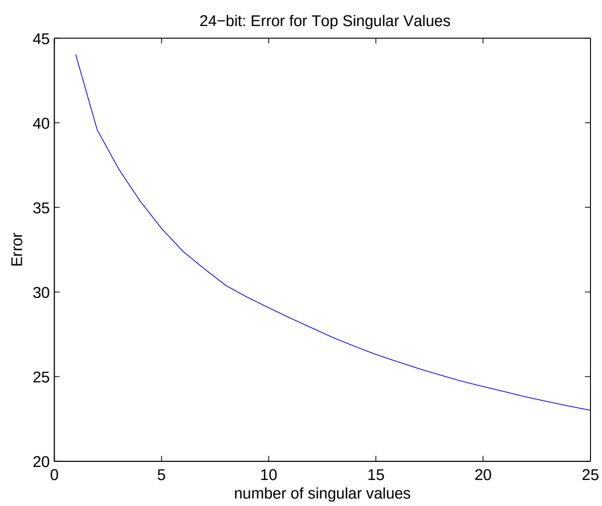

Figure 18: Error plot for experiment 4

In this experiment, we see similar results as seen in experiment 1. The errors are a little higher but the advantage is that we only do SVD on the image once as we have converted it to a 24-bit image instead of individually for the red, green and blue channels.

Experiment 5 In experiment 5 we converted to 24 bit and took the bottom N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

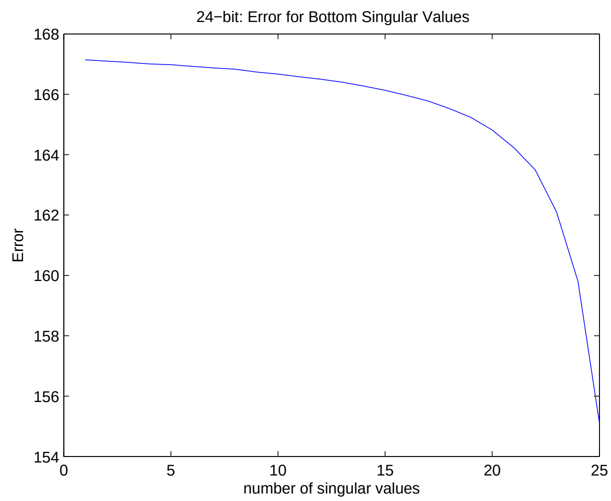

Figure 19: Error plot for experiment 5

In this experiment we see the results to be similar to experiment 2 but with the advantages and disadvantages of experiment 4.

Experiment 6 In experiment 6 we converted to 24 bit and took random N singular vectors and generated the image, the error image and calculated the error between the generated and original image.

Though the results are similar to experiment 3, we see the error is much higher than in experiment 3.

Also, in experiments 4,5 and 6 we see a dominant colour (red or blue or green) during the initial images while the another dominant colour in the error image(red or blue or green). We do not observe this in experiments 1,2 and 3.

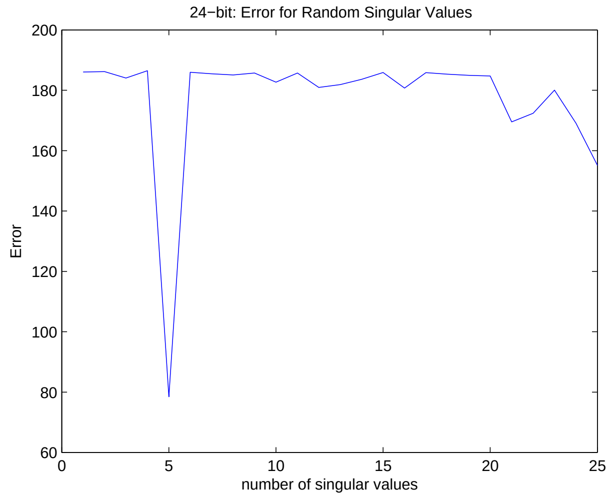

Figure 20: Error plot for experiment 6

Eigenvalue decomposition:

Since the image is rectangular and eigenvalue decomposition is possible only for square matrices, we have two ways of approaching this problem. We can follow the suggestion given and transform the matrix.

$$I_1 = A'_{mxn} * A_{mxn} \tag{5}$$

$$I_1 = \sqrt{I_1} \tag{6}$$

$$[V_1, D_1] = eig(I_1) \tag{7}$$

$$I_2 = A_{mxn} * A'_{mxn} \tag{8}$$

$$I_2 = \sqrt{I_2} \tag{9}$$

$$[V_2, D_2] = eig(I_2) \tag{10}$$

Let us assume n is greater than m. We take D as a matrix of mxn either with D_1 or D_2.

$$A_{new} = V_2 * D * V'_1 \tag{11}$$

The second possibility is to divide the image into square sections. In the given image the ratio of length : breadth is 4 : 3. We can then follow the procedure as given for image 1 and combine the images to get back the original image.

Upon searching for other potential solutions, we came across that our first method will not return the image. In fact, it is recommended to use only SVD for image compression specifically rectangular images.

Experiment 1 In experiment 1 we took the top N eigen vectors and generated the image, the error image and calculated the error between the generated and original image.

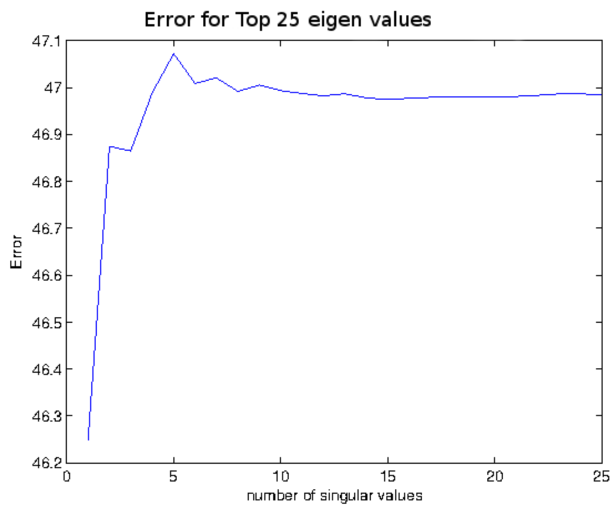

Figure 21: Error plot for experiment 1

As seen from the graph, the results are very different with very high error from the results we found for image 1. This supports our hypothesis that we cannot accurately return the image.

Experiment 2 In experiment 2 we took the bottom N eigen vectors and generated the image, the error image and calculated the error between the generated and original image.

As seen from the graph, the results are very different with very high error from the results we found for image 1. This supports our hypothesis that we cannot accurately return the image.

Experiment 3 In experiment 3 we took the random N eigen vectors and generated the image, the error image and calculated the error between the generated and original image. As seen from the graph, the results are very different with very high error from the results we found for image 1. This supports our hypothesis that we cannot accurately return the image.

Figure 22: Error plot for experiment 2

Figure 23: Error plot for experiment 3

Experiment 4 In experiment 4 we converted to 24 bit and took the top N eigen vectors and generated the image, the error image and calculated the error between the generated and original image.

As seen from the graph, the results are very different with very high error from the results we found for image 1. This supports our hypothesis that we cannot accurately return the image.

Experiment 5 In experiment 5 we converted to 24 bit and took the bottom N eigen vectors and generated the image, the error image and calculated the error between

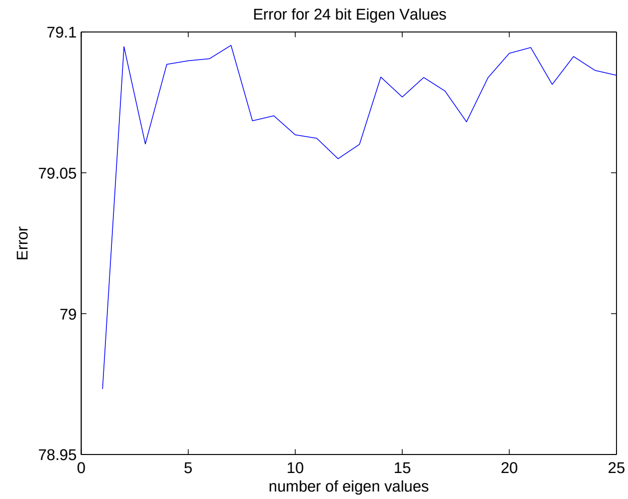

Figure 24: Error plot for experiment 4

the generated and original image.

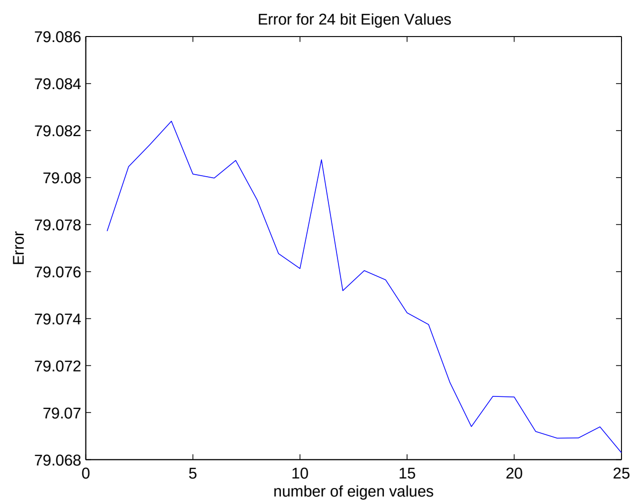

Figure 25: Error plot for experiment 5

As seen from the graph, the results are very different with very high error from the results we found for image 1. This supports our hypothesis that we cannot accurately return the image.

Given the found results, we decided not to do the sixth experiment as it seemed redundant though we are submitting the code for the same.

Inference

- Power is maximum on top singular values.
- After about top 25 singular values, we cant find significant improvement in the image quality.
- For random N the quality depend mainly on the values that is randomly selected.
- Same procedure for SVD for any image while eigenvalue decomposition can be applied only to square matrices.
- Converting to 24 bit increases the error on average for a given N values.

Problem 2

For this problem we used normal equation method to do linear regression for the three cases.

$$\theta = (X^T X)^{-1} X^T y \tag{12}$$

where θ - coefficients to be found for the fitting equation
X - Given independent variables
y - Given dependent variable

We used 70% of the data for training. We used 20% of the data for validation and the last 10% for testing.

One Variable

As seen from the graph, the coefficients give us a line. The fourth image shows the graph between the actual and predicted values. As we can see it forms an almost 45^o angle to the axis and behaves almost like $y = x$, which confirms the given estimation is accurate.

Two Variable

As seen from the graph, the coefficients give us a plane.The fourth image shows the graph between the actual and predicted values. As we can see it forms an almost 45^o angle to the axis and behaves almost like $y = x$, which confirms the given estimation is accurate.

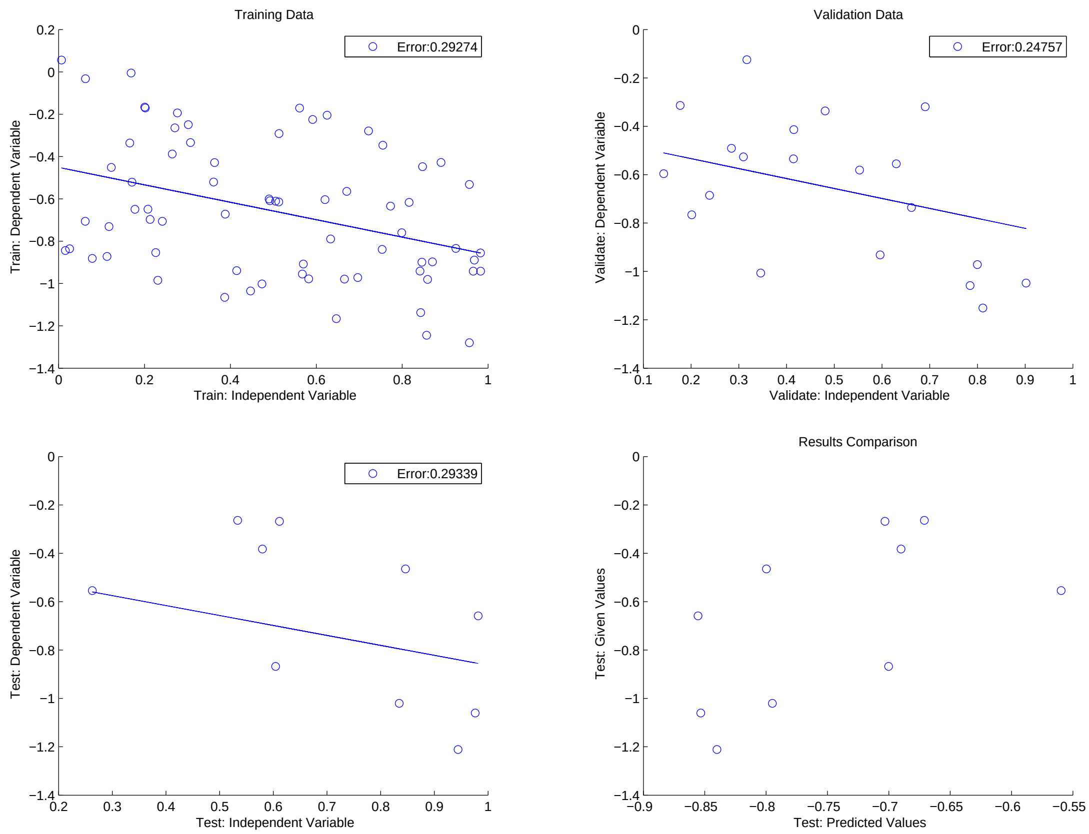

Figure 26: 1 independent variable

Hundred Variable

As seen from the graph, the coefficients give us a hyperplane. The image shows the graph between the actual and predicted values. As we can see it forms an almost 45^o angle to the axis and behaves almost like $y = x$, which confirms the given estimation is accurate.

Inference

- One can find the direction of the predominant vectors in the training data using coefficients.
- This method gives us an accurate value of the coefficients if we have enough data to train the model.
- Since we cannot visualize for more than 2 independent variables to see if our linear regression is correct, we can plot the given results with the predicted results and it should be close to the equation $y = x$.

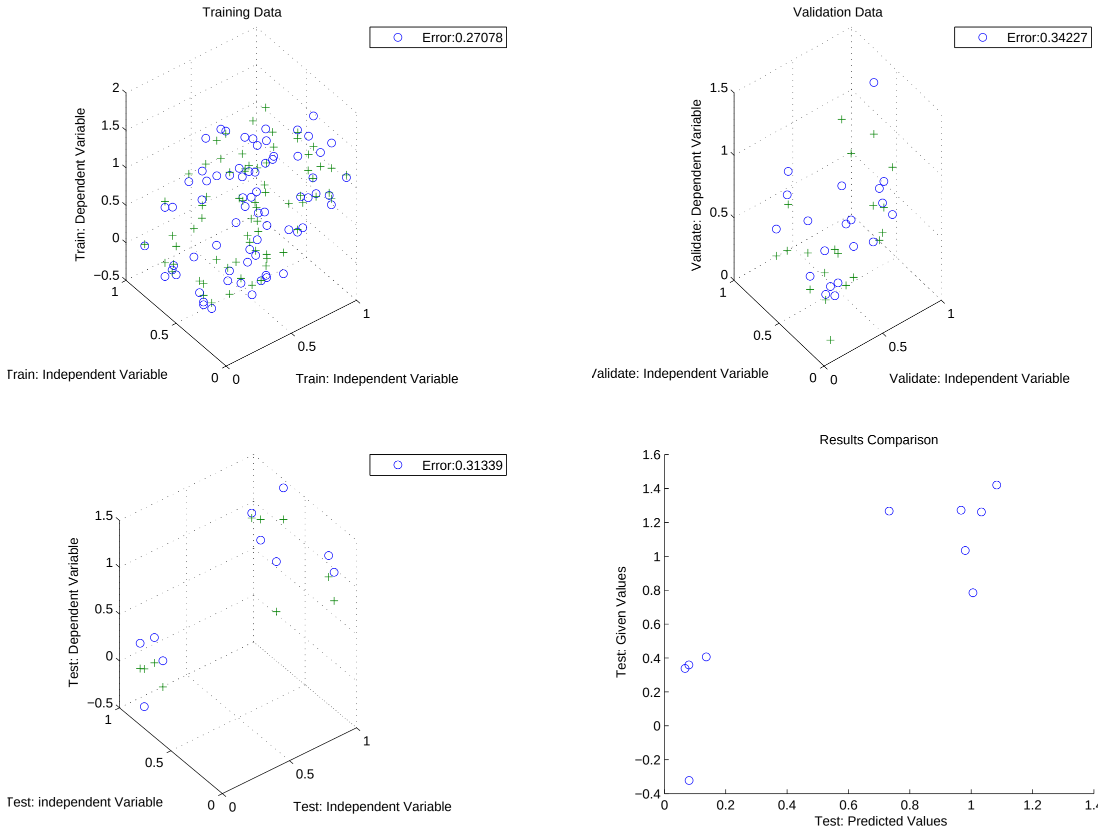

Figure 27: 2 independent variables

We have not added any of the images due to the file size constraint. We will show the images obtained during the viva-voice.

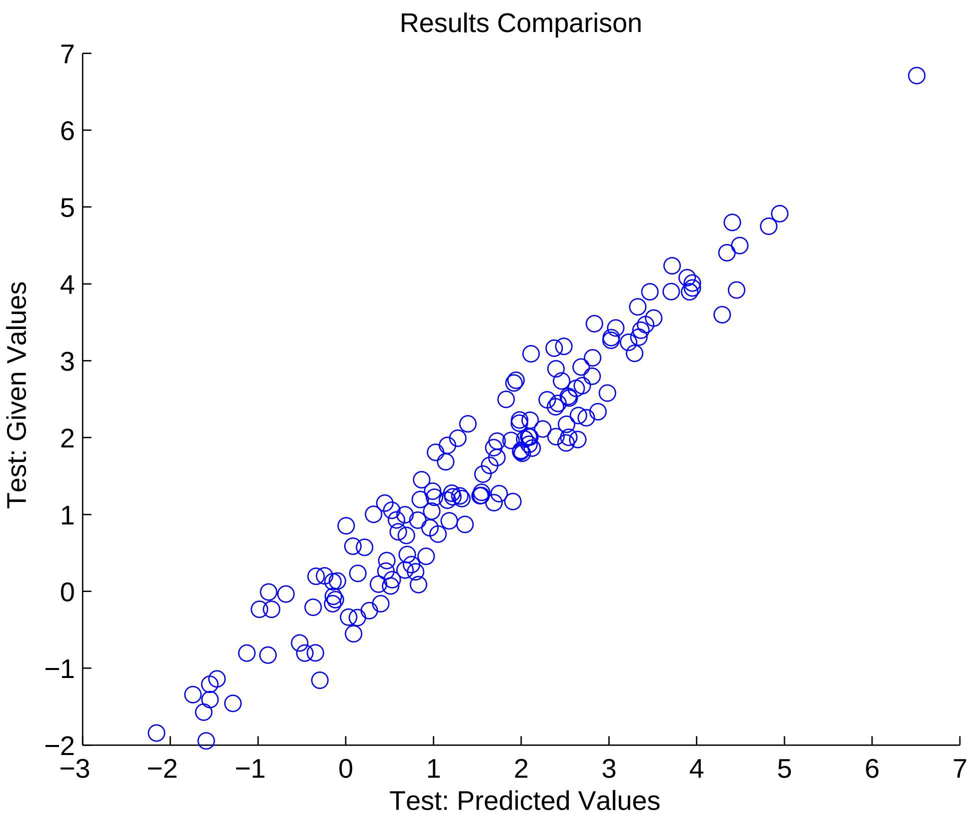

Figure 28: 100 independent variables