Analysis of Blind Image Quality Index

Vipin Milind Kamble

Department of Electrical Engineering
National Institute of Technology, Rourkela
Rourkela-769008, Odisha, INDIA

June 2013

Analysis of Blind Image Quality Index

A thesis submitted in partial fulfillment of the
requirements for the degree of

Master of Technology

in

Electronics Systems & Communication

by

Vipin Milind Kamble
(Roll-211EE1118)

Under the Guidance of

Dr. (Prof.) Supratim Gupta

Department of Electrical Engineering
National Institute of Technology, Rourkela
Rourkela-769008, Odisha, INDIA

2011-2013

Department of Electrical Engineering
National Institute of Technology, Rourkela

C E R T I F I C A T E

This is to certify that the thesis entitled ***"Analysis of Blind Image Quality Index"*** *by* ***Mr. Vipin Milind Kamble****, submitted to the National Institute of Technology, Rourkela (Deemed University) for the award of Master of Technology in* ***Electrical Engineering*** *with the specialization of* ***"Electronic Systems and Communication"****, is a record of bonafide research work carried out by him in the* ***Department of Electrical Engineering****, under my supervision. I believe that this thesis fulfills part of the requirements for the award of degree of Master of Technology. The results embodied in the thesis have not been submitted for the award of any other degree elsewhere.*

Dr. Supratim Gupta
Dept. of Electrical Engineering
National Institute of Technology
Rourkela, Odisha, 769008
INDIA

Place: N.I.T., Rourkela
Date

Acknowledgements

First and foremost, I am truly indebted to my supervisors Professor Supratim Gupta for his excellent guidance, inspiration and showing confidence in me, without which this thesis would not be in its present form. I also thank him for his encouraging words and teaching me a way to look at things very differently.

I express my gratitude to professors of my specialization, Dr. P. K. Sahu, Dr. D. Patra, Dr. S. Das, and Mrs. K. R. Suhasini, for their advise and care. I am also very much obliged to Head of the Department of Electrical Engineering, NIT Rourkela for providing all the possible facilities towards this work. Thanks also to other faculty members in the department.

I would like to thank Sushant Sawant, Susant Panigrahi (PhD Scholar) and Ayan Santra at Embedded System and Real-Time Laboratory, NIT Rourkela, for their enjoyable and helpful company.

My wholehearted gratitude to my parents, Maya and Milind Kamble and my sister, Madhavi for their encouragement and support.

Vipin Milind Kamble

Rourkela, June 2013

Abstract

Image quality index is the measure for estimating the level of degradation present in an image. Measurement of such index is challenging in the absence of reference image. Blind image quality assessment refers to evaluating the quality of an image without the need of any reference image. The quality of an image can be considered as the contrast, sharpness, brightness and other features extracted from that particular image. Other features like Discrete Cosine Transform (DCT), Wavelet Transform and Gabor filtering can also be used to extract the quality of an image.

Different algorithms are developed by researchers to solve the quality evaluation problem. These algorithms are not tested on a common platform. The algorithms that are analyzed in this thesis are Blind Image Quality Index (BIQI), Distortion Identification-based Image Verity and INtegrity Evaluation (DIIVINE), BLind Image Integrity Notator using DCT Statistics (BLIINDS) & Visual Codebook. Laboratory for Image & Video Engineering (LIVE) database which is a standard database is used to analyze the mentioned algorithms. Spearman and Pearson correlation coefficients are used for validating the algorithms.

Recently Visual Codebook algorithm was proposed by Peng Ye and David Doermann. The existing Visual Codebook algorithm is optimized with respect to the number of clusters used in K-Means clustering part of algorithm. Effect of variation in patch size on the performance of algorithm is studied in this thesis and an optimum value of patch size is proposed.

Contents

List of Abbreviations

Abbreviation	Description
NSS	Natural Scene Statistics
DCT	Discrete Cosine Transform
SSIM	Structural Similarity Index
BIQI	Blind Image Quality Index
BLIINDS	BLind Image Integrity Notator using DCT Statistics
IQA	Image Quality Assessment
NR	No Reference
FR	Full Reference
RR	Reduced Reference
PSNR	Peak Signal to Noise Ratio
DIIVINE	Distortion Identification-based Image Verity and INtegrity Evaluation
DMOS	Differetial Mean Opinion Score
VC	Visual Codebook
SROCC	Spearman Rank Order Correlation Coefficient
LCC	Linear (Pearson's) Correlation Coefficient
LIVE	Laboratory for Image & Video Engineering

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Quality of an image refers to the amount of degradation present in an image. A high quality image is always desirable. For example, the images taken by camera can be of varying quality. There can be presence of noise or any other distortion in an image. Distortions can occur due to acquisition, compression, storing and decompression of an image, movement of camera while capturing image or addition of noise in an image. The quality of an image cannot be decided by only few parameters like brightness, contrast or sharpness. A sharp image can have salt and pepper noise present in it. There need a standardize procedure to evaluate the quality of an image regardless of the type of distortion that has affected the image.

One way to evaluate the quality of any image is to have a subjective evaluation. In subjective evaluation, the image is shown to some observers, for example lets us assume that the image is shown to 10 observers. Evaluation by only one observer cannot be perfect as the observer's eye sight might not be perfect, so to remove this discrepancy, more than one observer is used for subjective evaluation. The distance between the observer and the image, viewing angle, lighting conditions and other affecting parameters are kept similar for all observers. The observers are then asked to give a quality score to the image on certain scale say 0 to 100, where 0 represents best quality and 100 represents worst quality. Average of the quality scores given by 10 observers gives the final quality of that image.

This is a lengthy procedure to give quality score to an image. It is time consuming as human observations are involved. The quality score cannot be accurate as average of quality value of different observers is taken. So this method of quality evaluation of an image cannot be used in real time applications. An automated system is required to evaluate the quality of an image in real time. This quality assessment problem can be categorized into 3 classes namely,

1) Full reference image quality assessment (FR-IQA)
2) Reduced reference image quality assessment (RR-IQA)
3) No reference image quality assessment (NR-IQA)

Full reference image quality assessment (FR-IQA) algorithms needs a reference or undistorted image beforehand to judge a quality of distorted image. The quality score is found out by comparing the distorted image with the undistorted image. Depending on the extent of distortion present in the image, the quality score is accordingly given. Example of FR-IQA is Structural Similarity Index (SSIM) [2], Fast SSIM [3] and Peak signal to noise ration (PSNR) [4]. The PSNR of an image is not a promising factor for quality evaluation which can be seen in figure 1.1 [1], all the images have same mean square error of 144 even though quality of each image is different which can be found out by visual inspection.

The main limitation of full reference image quality assessment algorithms is that they require the original, undistorted image to evaluate the quality of distorted image. These algorithms cannot be used in applications where reference image is not available. Also the PSNR values are not consistent with the human visual system [5].

Reduced reference image quality assessment (RR-IQA) algorithms [6] use a training approach to evaluate the quality of an image. In RR-IQA, first the algorithm is trained for the change in quality score for the extent

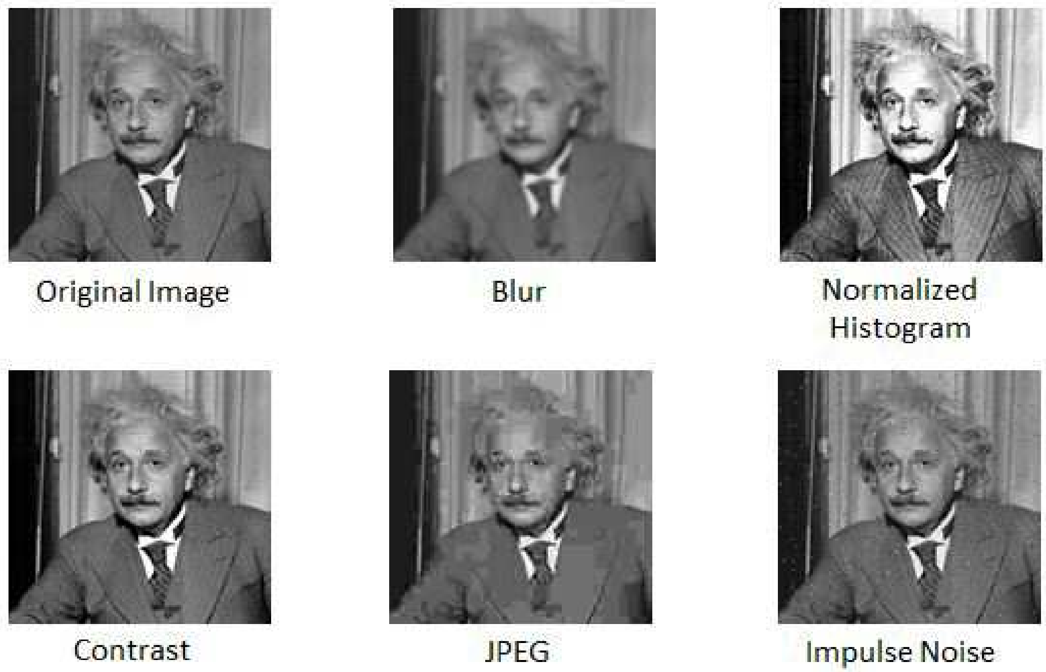


Figure 1.1: Einstein images with same MSE = 144 [1]

of distortion, a training data set is formed. When the algorithm is applied on a test image, the parameters extracted from the test image are compared with the training data set to get the quality score. The algorithm is trained for specific type of distortions, so if a test image having a distortion which is not used during training comes for quality evaluation, this algorithm fails to give correct quality score.

No reference image quality assessment (NR-IQA) algorithms give the quality score just by processing the test image. There is no need of the reference image or any training images hence it is also called Blind IQA. Different parameters can be used to evaluate the quality of an image blindly. Anisotropy [7], Discrete Cosine transform (DCT) [8, 9], wavelet [10, 11] and Gabor filtering [12] are some of the parameters used for NR-IQA.

The NR-IQA algorithms developed so far are distortion specific, i.e. the algorithm works fine only for specific type of distortions. The type of distortion present in the image should be mentioned before applying the algorithm. If the type of distortion is unknown then the algorithm wont give proper results. Generally the algorithm is designed for distortions like blur, noise, fast fading, jpeg2000 and jpeg.

1.2 Literature Review

Every individual perceives images differently. As the perception is different, the image quality viewed by individual is different from others. An algorithm or system which can evaluate the quality of an image can remove the discrepancy of difference in perception.

The full reference algorithms like PSNR [4] & SSIM [2] provides a way to estimate the quality of an image but these algorithms need an original image with which the distorted image can compared to calculate the quality of image. These algorithms give the amount by which the distorted image differs from the original image. But the need of original image limits the use of these algorithms. So research work is being carried out to develop an algorithm which can evaluate the quality of an image without the need of reference image.

Different transforms like Discrete Cosine Transform (DCT) & Wavelet Transform are used by researchers to evaluate the quality of an image. Image has the property of anisotropy i.e. its value is different in different directions. So the transforms used is applied for various orientations. Different scales of transforms are also used for quality evaluation purpose. Using these techniques, different algorithms are developed to solve the quality evaluation problem.

Table 1.1 shows the no-reference image quality assessment algorithms developed so far. The algorithms are compared using Spearman Rank Correlation Coefficient (SROCC) and Linear (Pearson's) Correlation Coefficient (LCC).

Drawbacks of algorithms mentioned in table 1.1 are

- The algorithms are not fully No-Reference, a standard data set is required to first train the algorithm for specific type of distortions.

Table 1.1: No-Reference Image Quality Assessment Algorithms

Name of Algorithm	Developer	Features Used	Correlation Parameters	Reference Indices
BIQI [10]	A.K. Moorthy A.C. Bovik	Wavelet Transform in 3 scales and 3 orientations	SROCC LCC	PSNR
DIIVINE [11]	A.K. Moorthy A.C. Bovik	Wavelet Transform to obtain sub-band coefficients for statistical features	SROCC LCC	SSIM PSNR
BLIINDS [8]	M.A Saad A.C. Bovik C. Charrier	Discrete Cosine Transform based contrast & structure features	SROCC	PSNR
BLLINDS-II [9]	M.A Saad A.C. Bovik C. Charrier	Model based Discrete Cosine Transform domain NSS features	SROCC	SSIM PSNR
Visual Codebook [12]	Peng Ye D. Doermann	Gabor Filtering in 4 orientations and 5 frequencies	SROCC LCC	SSIM PSNR

- Computational time is more so not suitable for real time applications.
- Difference in actual Differential Mean Opinion Score (DMOS) and quality score given by present algorithms.
- Present algorithms are distortion specific, i.e. the they are trained for limited type of distortions.

1.3 Motivations

Existing image quality assessment algorithms provide quality scores which are close to actual quality of that image but not the exact quality. Also the present NR-IQA are distortion specific but an algorithm should be generalized and should work for all type of distortions. A new no-reference image quality assessment algorithm need to be developed which can overcome these problem.

1.4 Applications

NR-IQA can be used in multimedia services like internet and television. Quality score of an image can be sent along with the image so loss of image quality due to channel loss or any other reason can be found out. Similarly in television, the quality score can be usefull to check the quality of tv signal.

NR-IQA can also be used in astronomical images. Consider the example images taken by of Hubble telescope, the cost (in terms of time) of images to be sent to earth station is high. So it would be useful to check the quality of image taken by telescope before transmitting the image to earth station, if quality is not ut to the mark then the same image can be recaptured without any delay.

1.5 Objectives

The objectives of the thesis are:

i. Analysis of existing NR-IQA.

ii. Implementing Visual Codebook algorithm which is one of the existing NR-IQA.

iii. Analysis of performance of algorithm by changing the patch size in Visual Codebook.

1.6 Contributions of the Thesis

The following are the salient contributions of the thesis.

- The effect of increasing degradation level on the output of NR-IQA algorithms is studied.
- Present NR-IQA algorithms are validated on a standard database i.e. Laboratory for Image & Video Engineering (LIVE) [13] using Spearman & Pearson correlation.
- An optimum patch size for Visual Codebook algorithm is proposed.

1.7 Thesis Organization

The thesis is organized as follows.

- Chapter 1 gives an introduction to image quality assessment problem.
- In Chapter 2, comparison of different NR-IQA is presented.
- Chapter 3 presents the results on LIVE database.
- In chapter 4, description of Visual Codebook algorithm is presented.
- Chapter 5 describes the effect of changing patch size in Visual Codebook algorithm.
- Chapter 6 concludes the thesis.

Chapter 2

Comparison of different NR-IQA algorithms

2.1 Need For Comparison

Different algorithms for No-Reference Image Quality Assessment are existing today. Which algorithm is better than the other is a question to be solved. An algorithm may produce accurate result by taking more computation time whereas another algorithm may give inaccurate result but with less computation time. The use of algorithm is determined by need of accuracy and execution time. The existing algorithms are not tested on a common platform. So to access the performance, the algorithms are tested on same database with a common index.

2.2 Correlation Coefficients

Performance of existing algorithms can be validated using different correlation coefficients. In practice, Spearman & Pearson correlation coefficients are more popular for performance measurement. The accuracy of an algorithm is decided by the value of correlation coefficient. The No-Reference Image Quality Assessment algorithms needs to be compared with some standard algorithm. For this purpose, Structural Similarity Index Measurement (SSIM)

[2] is used. SSIM is a Full-Reference Image Quality Assessment algorithm with a high correlation value. Both correlation coefficients are explained in the next section.

2.2.1 Pearson Correlation

Pearson correlation [14] is used to measure the dependency between two variables. It gives the correlation between two variables. Its value lies between '-1' & '+1' where value close to '+1' indicates that the two variables have positive correlation and values close to '-1' indicates that the two variables have negative correlation. Value close to zero implies that the two variable are not correlated. Pearson correlation between two variables 'X' & 'Y' is shown in equation 2.1 -

$$\rho = \frac{cov(X, Y)}{\sigma_x \cdot \sigma_y} \tag{2.1}$$

2.2.2 Spearman Correlation

Spearman correlation [15] also gives the correlation between two variables. The range of spearman correlation lies between '-1' and '+1' and values close to zero specifies less correlation between the two variables under test and values near '-1' and '+1' specifies that the two variables are highly correlated. Spearman correlation between two variables 'X' & 'Y' is shown in equation 2.2 -

$$\rho = 1 - \frac{6 \sum d^2}{n(n^2 - 1)} \tag{2.2}$$

where 'd' is the difference in ranks of the two variables and 'n' is the number of scores of variables 'X' & 'Y'.

2.3 Correlation Observations

The performance evaluation of different NR-IQA is done by following procedure. The two variable in the correlation formula are the quality scores of an image calculated by two algorithms under test. As SSIM is used as a reference algorithm to compare performance of other NR-IQA, one variable in the Pearson correlation is the SSIM values. A standard image 'barba.bmp' shown in figure 2.1 is used for quality evaluation. PSNR values are also evaluated to show the performance of one FR-IQA algorithm.

Figure 2.1: Test image : Barba

The test image is degraded by increasing the variance of noise. Gaussian noise, Speckle noise and Gaussian blur are used to degrade the test image. Quality score by each NR-IQA is obtained at each level of degradation. So for each algorithm, a vector is obtained which contains the quality score given by that algorithm for increasing values of noise variance. Correlation coefficients are calculated using these vectors. To find the correlation of each algorithm, the quality score vector obtained by SSIM and the quality score vector obtained by the algorithm are taken and their correlation is calculated.

The quality score given by SSIM lies between '0' and '1', where '1' represents best quality image and '0' represents worst quality image. For other NR-IQA algorithms, the range of quality score is from 0 to 100 where 0 represents best quality and 100 represents worst quality. As the scale for SSIM and all other NR-IQA is opposite, the correlation values is negative as shown in following tables.

The variations of quality score of different algorithms with increasing Gaussian noise variance are shown in figures 2.2, 2.3 & 2.3. The variations of quality score of different algorithms with increasing Speckle noise variance are shown in figures 2.5, 2.6 & 2.3. The variations of quality score of different algorithms with increasing Gaussian blur variance are shown in figures 2.8, 2.9 & 2.3.

Table 2.1 shows the Pearson correlation values of PSNR and NR-IQA algorithms with SSIM.

Table 2.1: Pearson Correlation Values

Algorithm	Gaussian Noise	Speckle Noise	Gaussian Blur
PSNR	0.9995	0.9935	0.9956
BIQI	-0.9984	-0.9989	-0.4527
BLIINDS-II	-0.9754	-0.9728	-0.8871
DIIVINE	-0.9963	-0.9934	-0.8953
VC	-0.9789	-0.8797	-0.5146

Table 2.2 shows the Spearman correlation values of PSNR and NR-IQA algorithms with SSIM.

Table 2.2: Spearman Correlation Values

Algorithm	Gaussian Noise	Speckle Noise	Gaussian Blur
PSNR	1	1	1
BIQI	-0.9879	-0.9972	-0.3188
BLIINDS-II	-0.9515	-0.9265	-0.8203
DIIVINE	-1	-0.9964	-0.9168
VC	-0.9879	-0.9635	-0.4958

2.4 GUI for Different Quality Assessment Algorithms

A Graphical User Interface (GUI) is created in MATLAB to show the output of different image quality assessment algorithms. The GUI provides

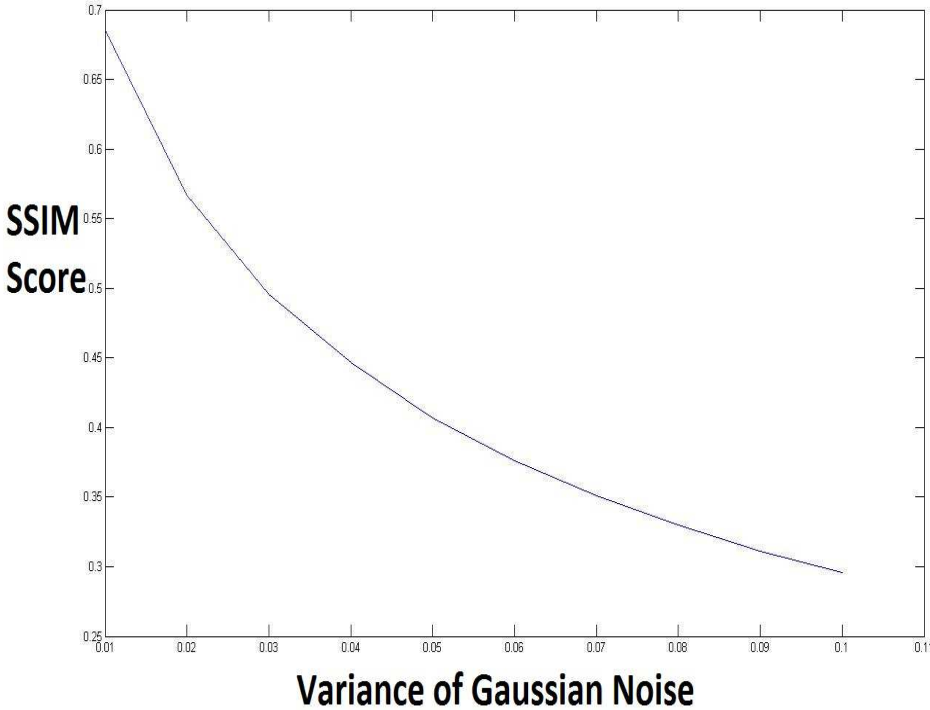


Figure 2.2: SSIM Quality score versus Variance of Gaussian Noise

quality scores of different algorithms and also the Pearson correlation values. Figure 2.11 shows the GUI. The GUI is provided with two ‘Browse’ buttons to browse for reference image and the distorted image. The ‘Quality’ button calculates the quality score given by each algorithm. SSIM & PSNR uses both, reference image and test image to calculate the quality score whereas other NR-IQA algorithms use only the test image. For the calculation of correlation values, the reference image is degraded using Gaussian noise, Speckle noise & Gaussian blur and the output of algorithms is correlated with SSIM.

2.5 Spearman versus Pearson Correlation

Spearman correlation takes into account only the ranks of two variables. Consecutive values are given consecutive ranks. So the two function one of which is linearly increasing and the other increasing with variable slope will have Spearman correlation value as ‘+1’.

Pearson correlation considers the covariance and standard deviation

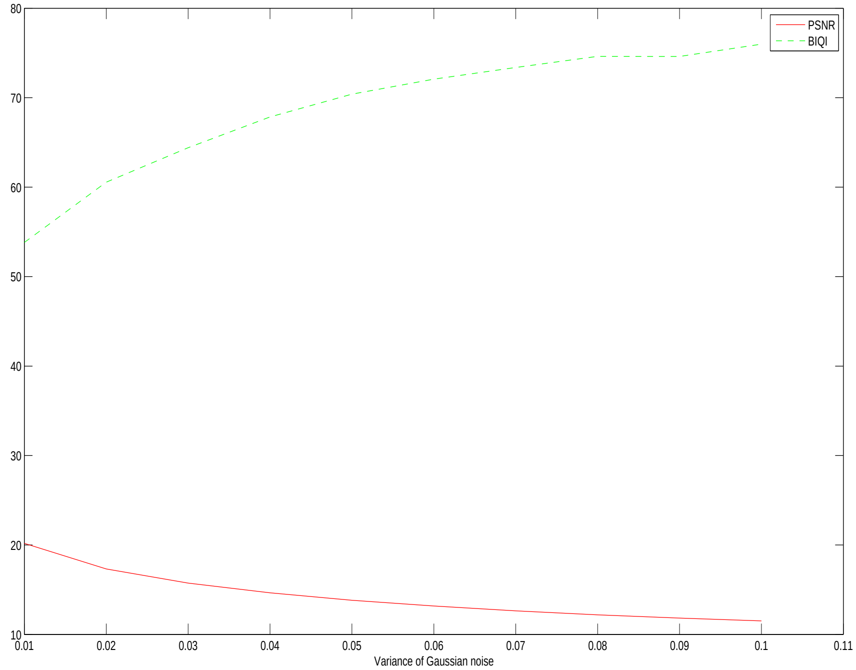


Figure 2.3: PSNR & BIQI Quality score versus Variance of Gaussian Noise

of the two variables under test. Figure 2.12 shows the comparison between Spearman and Pearson correlation, two increasing functions are used and correlation values are calculated.

From figure 2.12, it can be observed that Pearson correlation coefficient value gives a better information about correlation of two variables. Spearman correlation value gives extent of association between two ranked variables whereas Pearson correlation value indicates the measure of linearity between two variables.

2.6 Summary

The effect of increasing level of degradation of an image on the output of Image Quality Assessment algorithms is described in this chapter. Spearman and Pearson correlation coefficients are explained and are used to measure the performance of different NR-IQA algorithms. Also the comparison between Spearman and Pearson correlation coefficients is discussed. A graphical user

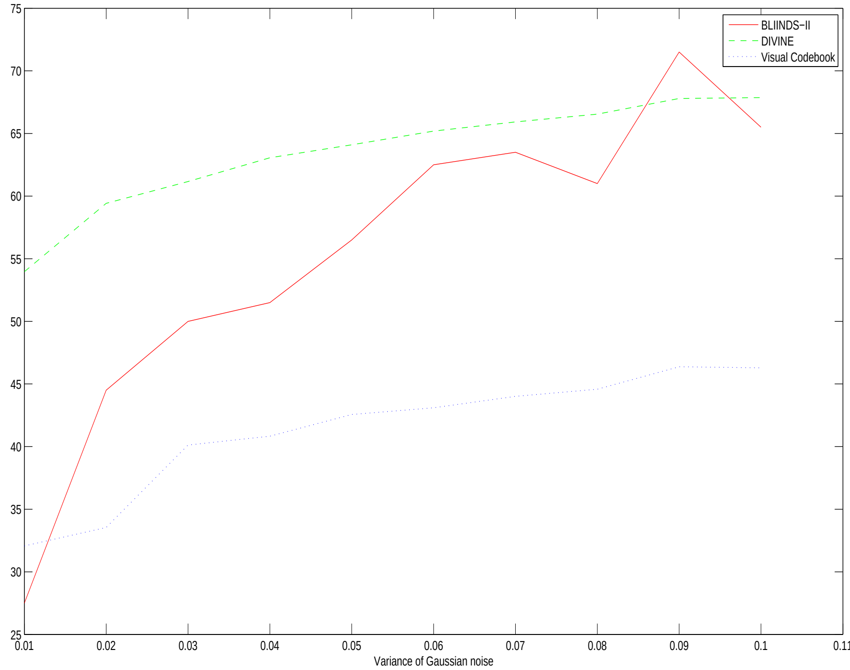


Figure 2.4: BLIINDS-II, DIIVINE & Visual Codebook Quality score versus Variance of Gaussian Noise

interface is created to analyze the existing NR-IQA algorithms.

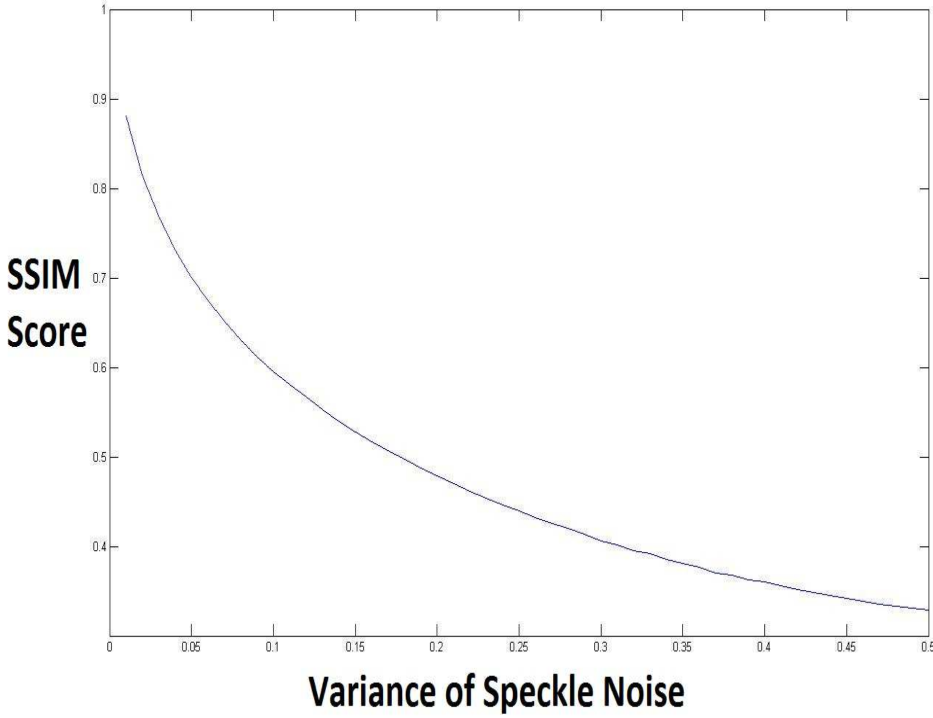


Figure 2.5: SSIM Quality score versus Variance of Speckle Noise

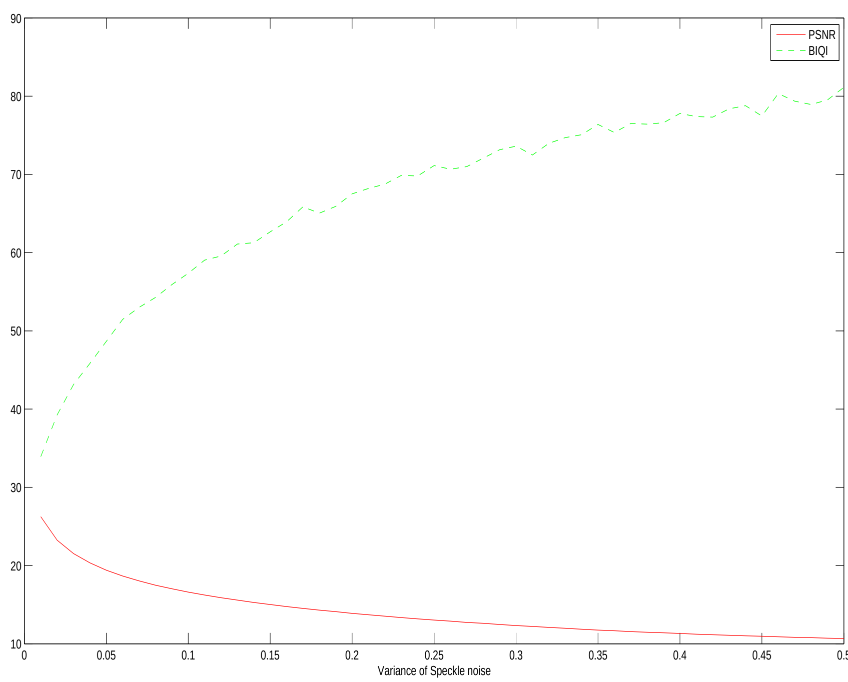


Figure 2.6: PSNR & BIQI Quality score versus Variance of Speckle Noise

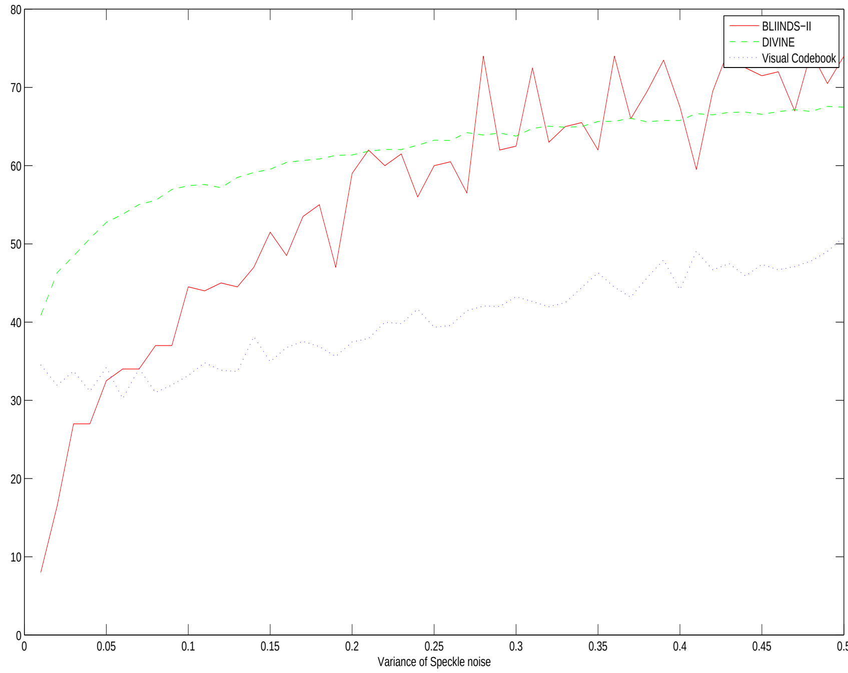


Figure 2.7: BLIINDS-II, DIIVINE & Visual Codebook Quality score versus Variance of Speckle Noise

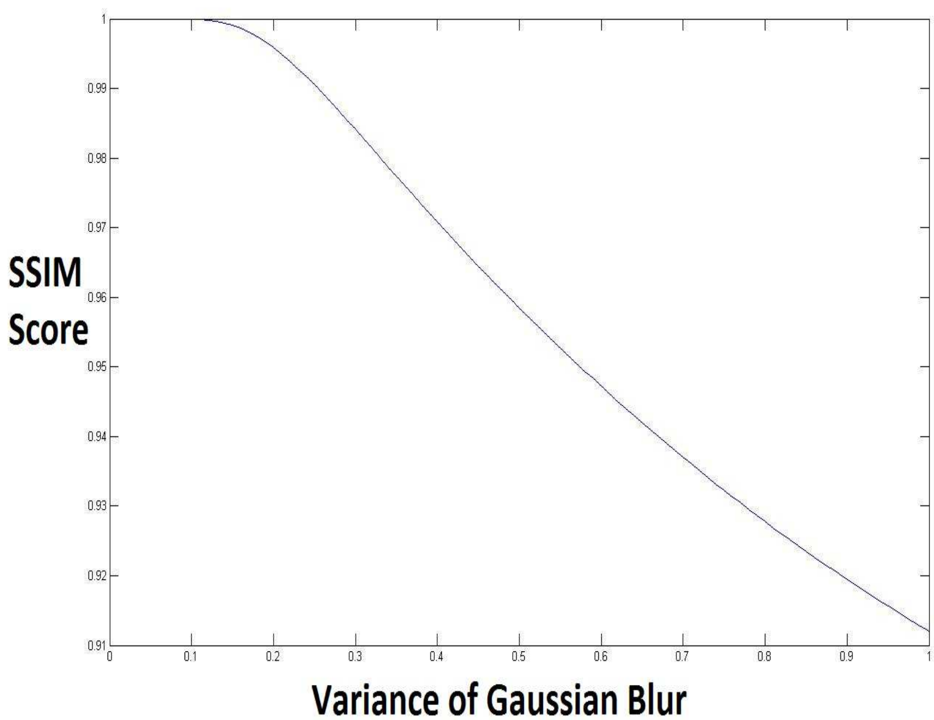


Figure 2.8: SSIM Quality score versus Variance of Gaussian blur

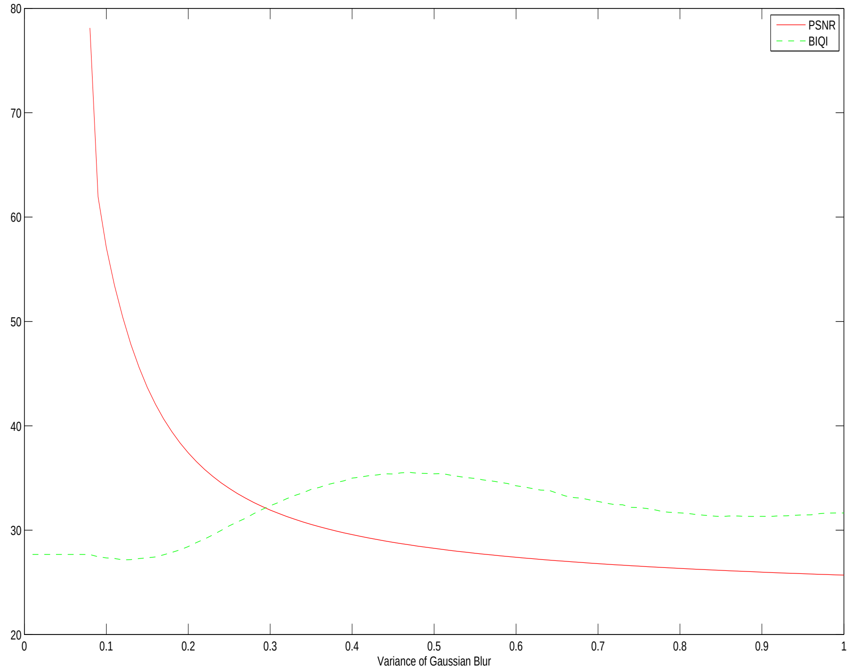


Figure 2.9: PSNR & BIQI Quality score versus Variance of Gaussian blur

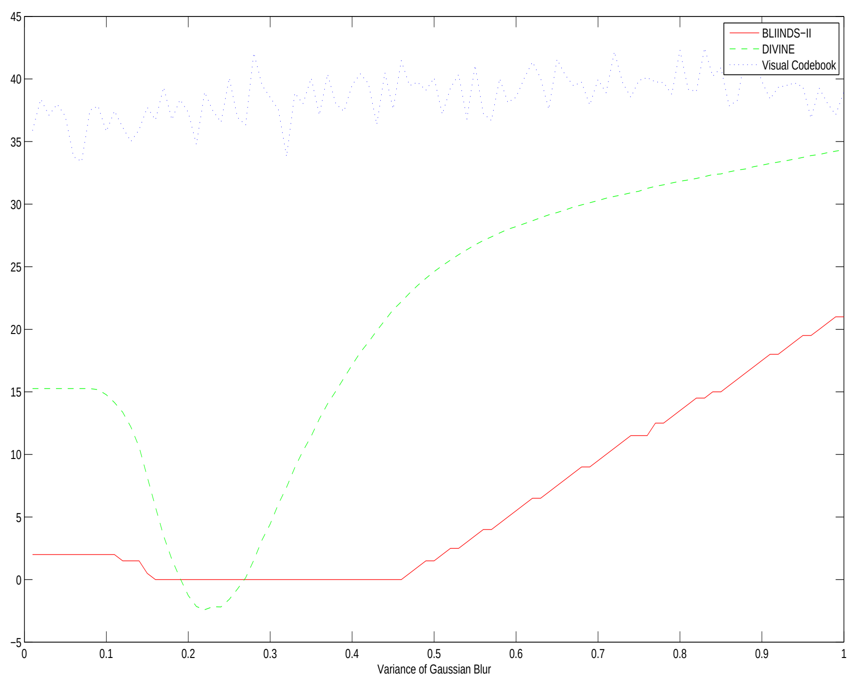


Figure 2.10: BLIINDS-II, DIIVINE & Visual Codebook Quality score versus Variance of Gaussian blur

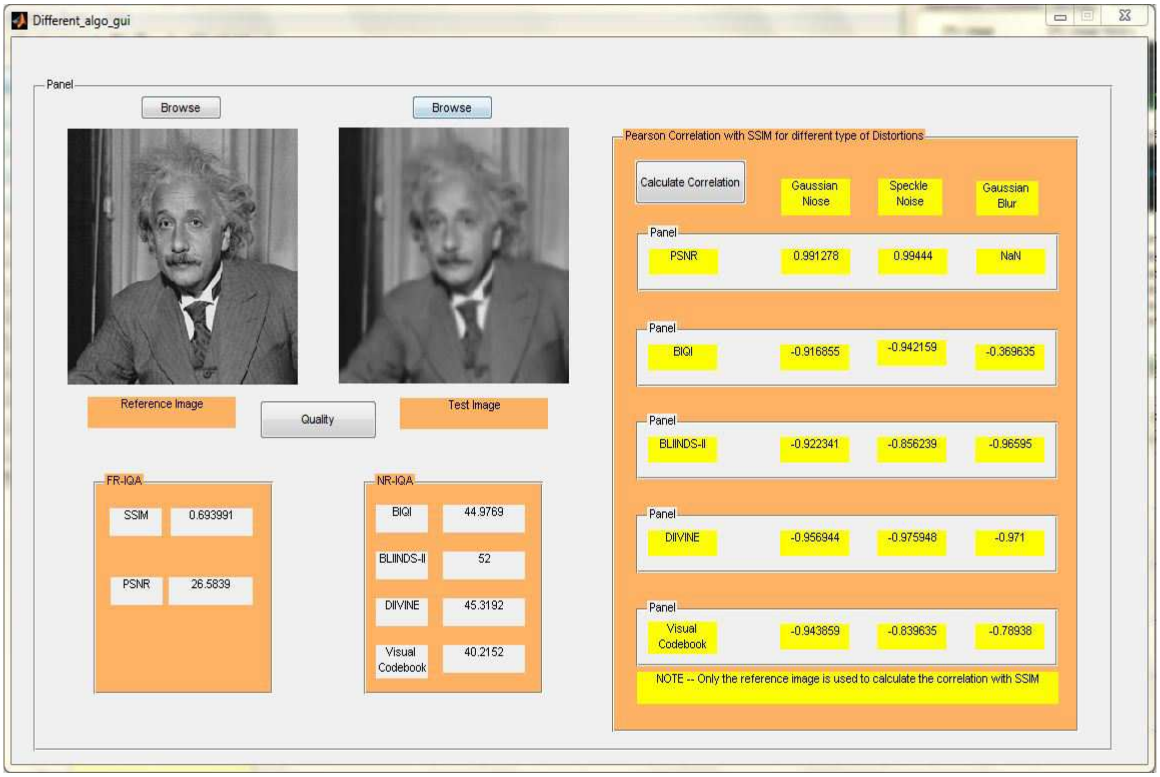


Figure 2.11: Image Quality Assessment Algorithms GUI

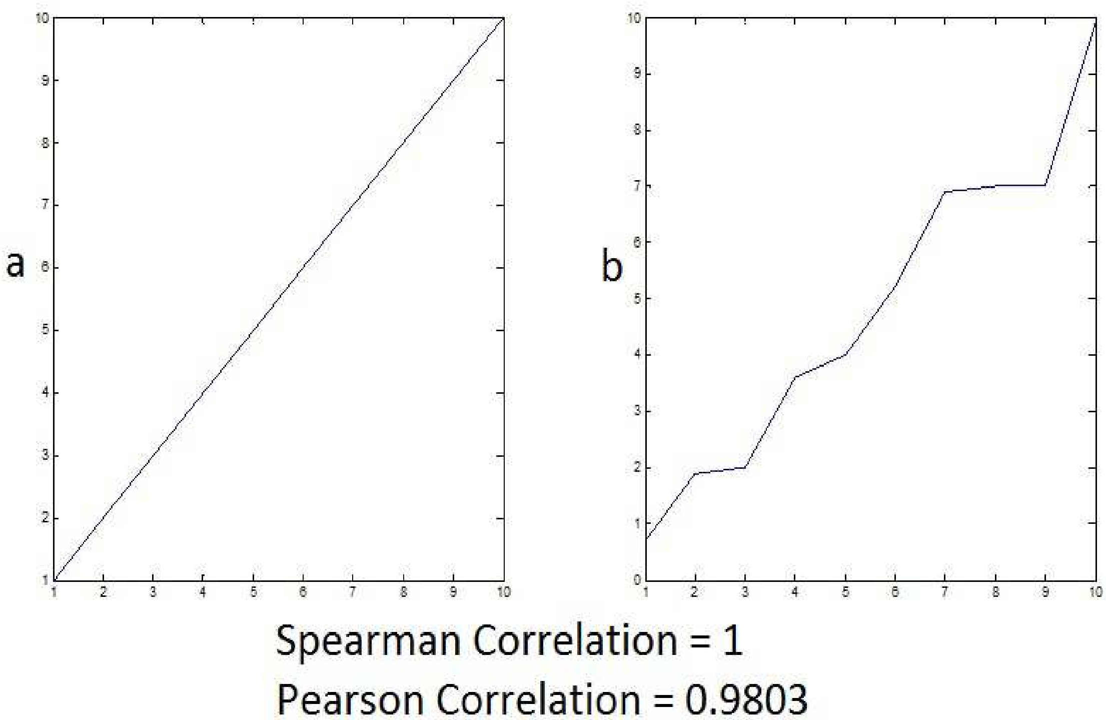


Figure 2.12: Spearman versus Pearson Correlation

Chapter 3

Analysis of Existing NR-IQA Algorithms

The existing No-Reference Image Quality Assessment algorithms needs to be validated using a common platform. A common platform helps in comparing the existing NR-IQA algorithms with respect to the accuracy of output of the algorithm. The algorithms need to be tested on a set of standard images which incorporate all major types of distortions.

3.1 LIVE Database

Laboratory for Image & Video Engineering (LIVE) [13] is a standard database which contains a set of images which can be used for validation of image quality assessment algorithms. The database contains both types of images, reference and its distorted versions. There are 29 reference images which are distorted by 5 type of distortions with different degradation levels. The types of distortions are -

- JPEG compression distortion (169 images)
- JPEG2000 compression distortion (175 images)
- Gaussian blur distortion (145 images)
- White noise distortion (145 images)
- Fast fading distortion (145 images)

The database is provided with a MATLAB file which contains the Differential Mean Opinion Score (DMOS) of each image present in the database. DMOS is the mean of quality scores given by different human observers. This score is considered as a standard quality score with which output of different NR-IQA algorithms is to be compared.

3.2 Correlation Observations

As shown in chapter 2, the performance evaluation of different NR-IQA can be done using correlation coefficients. In this chapter, LIVE database is used to get output of an algorithm and then the output is compared with DMOS. The two variable in the correlation formula are the quality scores of images in LIVE database calculated by NR-IQA algorithm under test and the DMOS. Spearman and Pearson correlation coefficients are used to compare the performance of different NR-IQA algorithms. Spearman correlation values are shown in Table 3.1 and Pearson correlation values are shown in Table 3.2.

Table 3.1: LIVE Database: Spearman Correlation

NR-IQA	JP2K	JPEG	White Noise	Gaussian Blur	Fast Fading
BIQI	0.9023	0.9121	0.9600	0.9632	0.8217
BLIINDS-II	0.9420	0.9076	0.9702	0.9345	0.8957
DIIVINE	0.8491	0.8107	0.9796	0.9697	0.8462
Visual Codebook	0.9380	0.9423	0.9401	0.9024	0.8943

Table 3.2: LIVE Database: Pearson Correlation

NR-IQA	JP2K	JPEG	White Noise	Gaussian Blur	Fast Fading
BIQI	0.8726	0.8301	0.9277	0.9296	0.7764
BLIINDS-II	0.9423	0.9004	0.9437	0.9269	0.8829
DIIVINE	0.8277	0.7384	0.9598	0.9561	0.8390
Visual Codebook	0.9335	0.9461	0.9268	0.8998	0.9018

In LIVE[13] database, each folder contains degraded images along with reference images. The quality score alloted to reference images is 0. But the NR-IQA algorithms need distorted images to evaluate their performance. So the Spearman and Pearson correlation values are again evaluated by taking only the distorted images in LIVE database. Spearman correlation values are shown in Table 3.3 and Pearson correlation values are shown in Table 3.4

Table 3.3: LIVE Database (excluding reference images): Spearman Correlation

NR-IQA	JP2K	JPEG	White Noise	Gaussian Blur	Fast Fading
BIQI	0.9187	0.8886	0.9903	0.9543	0.8205
BLIINDS-II	0.9163	0.8868	0.9596	0.9102	0.8349
DIIVINE	0.9025	0.7511	0.9878	0.9584	0.8592
Visual Codebook	0.8711	0.8829	0.9008	0.8357	0.8218

Table 3.4: LIVE Database (excluding reference images): Pearson Correlation

NR-IQA	JP2K	JPEG	White Noise	Gaussian Blur	Fast Fading
BIQI	0.9124	0.8242	0.9928	0.9598	0.8072
BLIINDS-II	0.9019	0.8977	0.9653	0.8993	0.8445
DIIVINE	0.8939	0.7020	0.9806	0.9589	0.8506
Visual Codebook	0.8630	0.8450	0.9038	0.8249	0.8181

It can be seen from tables 3.3 & 3.4 that the correlation values ranges from 0.7 to 0.99. A high correlation value is desirable. The performance of algorithms varies with the type of distortion which can be seen by difference in correlation values for different types of distortions. An ideal algorithm should perform equally for all type of distortions. The difference in correlation values for different types of distortions suggests that the algorithm is distortion specific.

3.3 Algorithm output Versus DMOS

The output of algorithm and the Differential Mean Opinion Score (DMOS) can be plotted on same graph using scatter plot. Scatter plots are obtained for quality score given by different NR-IQA and the actual quality score of an image. Figures 3.1, 3.2, 3.3, 3.4 & 3.5 shows the scatter plot for different types of distortions in LIVE database excluding reference images. X-axis is the Differential Mean Opinion Score (DMOS) and Y-axis is the quality score given by NR-IQA algorithm. A dispersed scatter plot indicates that the correlation between output of NR-IQA algorithm and the DMOS is less and vice versa. If the scatter plot forms only a line then the correlation value is one.

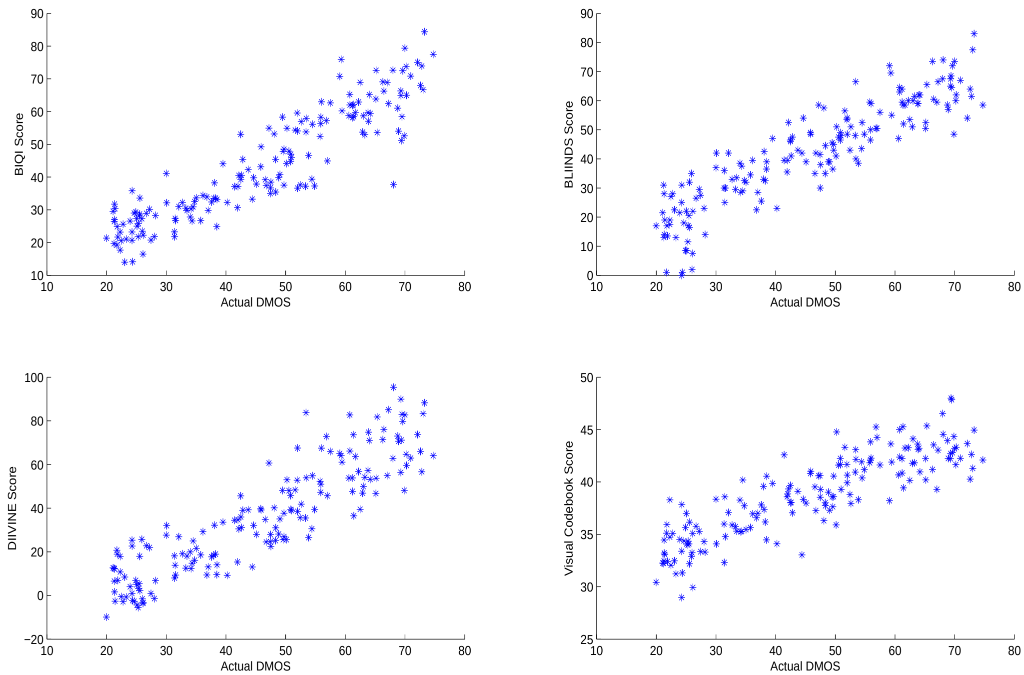


Figure 3.1: Scatter plot : JPEG2000 Distortion

3.4 NR-IQA Algorithms Performance Database

A database of output of existing NR-IQA algorithms is made which contains the output of each algorithm and the time taken by the algorithm to calculate the quality score. LIVE database images are used to make the database of algorithms' output. For each image present in the LIVE database, the image is subjected to Gaussian noise of variance values equal to 0.015 and 0.080, the noisy image thus obtained is resized to a factor of 0.75 and 0.5 and then the quality score and computation time for each NR-IQA algorithm is recorded. The observations are saved in an excel file. LIVE database contains images with 5 types of distortions, so for each type of distortion, an excel file is obtained.

The database in excel files shows the behavior of each NR-IQA algo-

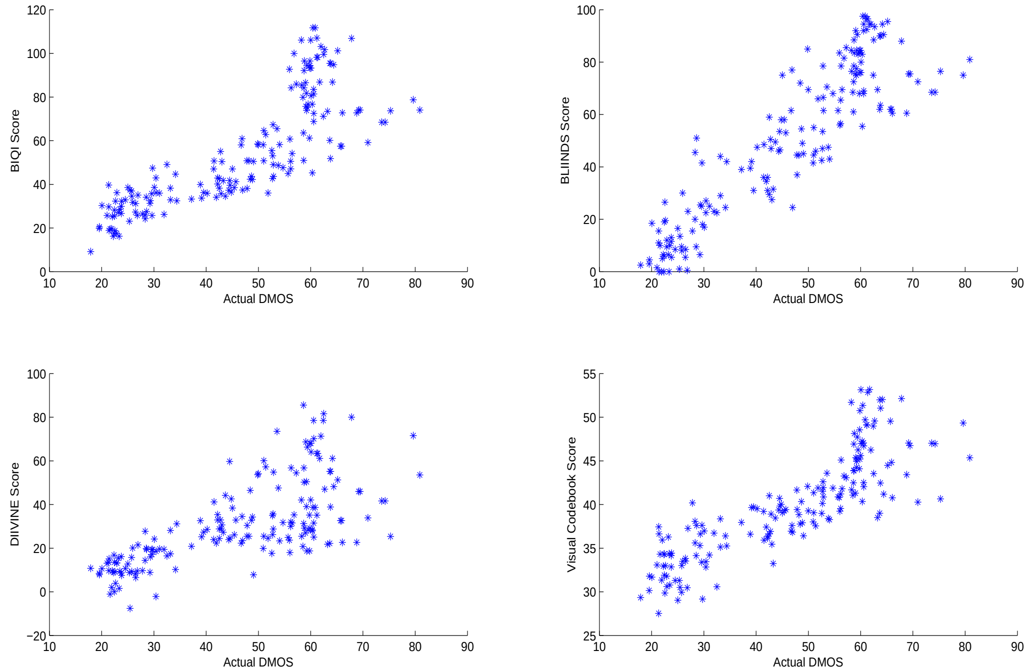


Figure 3.2: Scatter plot : JPEG Distortion

rithm for changing the size and noise strength present in an image. This database can be used for reviewing performance of each NR-IQA algorithm efficiently. The quality score and time taken by each NR-IQA algorithm are saved in the excel files so the need to execute the algorithm is eliminated. The excel file can be directly referred to get the quality score and computation time taken by all NR-IQA for LIVE database. Snapshot of one such excel file is shown in figure 3.6, it shows the data obtained from first and second image of JPEG type of distortion of LIVE database.

3.5 Summary

This chapter examines the performance of existing NR-IQA algorithms using LIVE database. Spearman and Pearson correlation values are used to validate the algorithms. The performance of algorithms is demonstrated by scatter plots for different types of distortions in images.

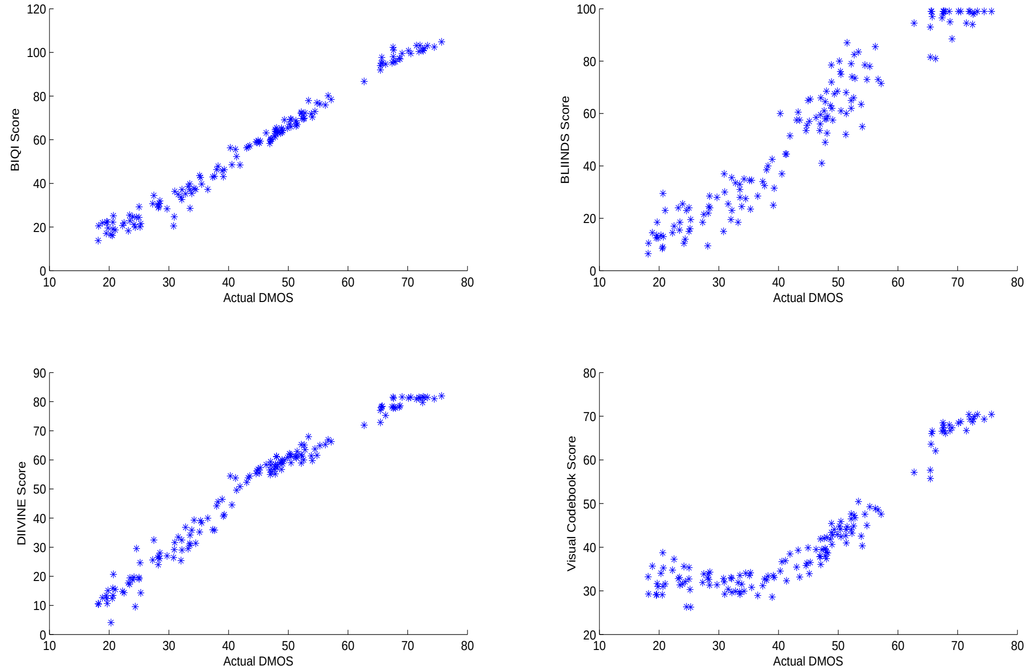


Figure 3.3: Scatter plot : White Noise Distortion

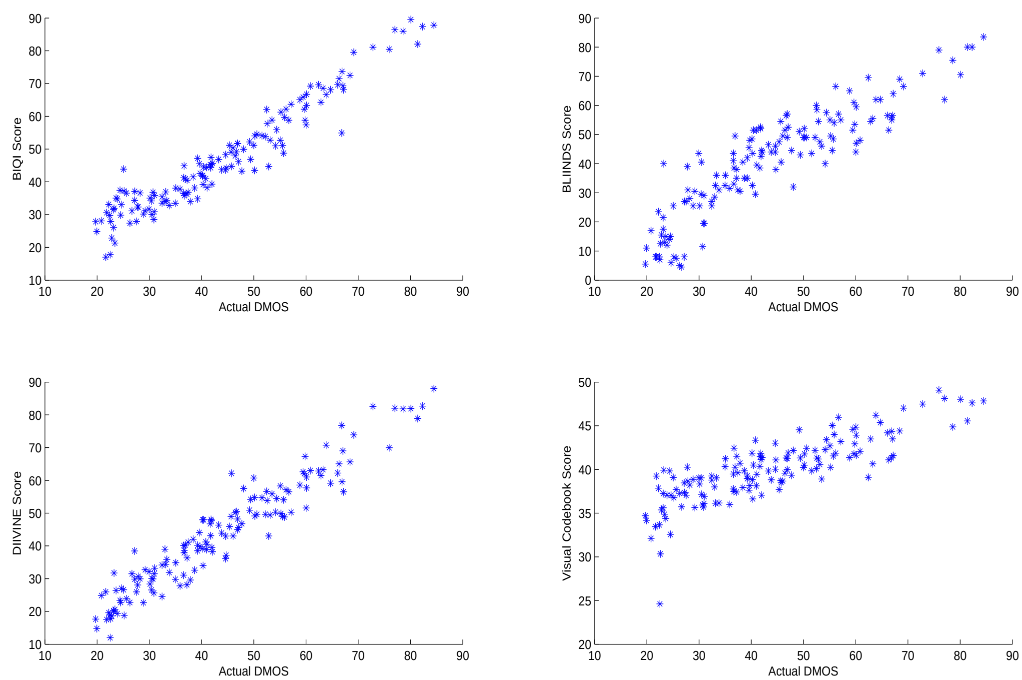


Figure 3.4: Scatter plot : Gaussian Blur Distortion

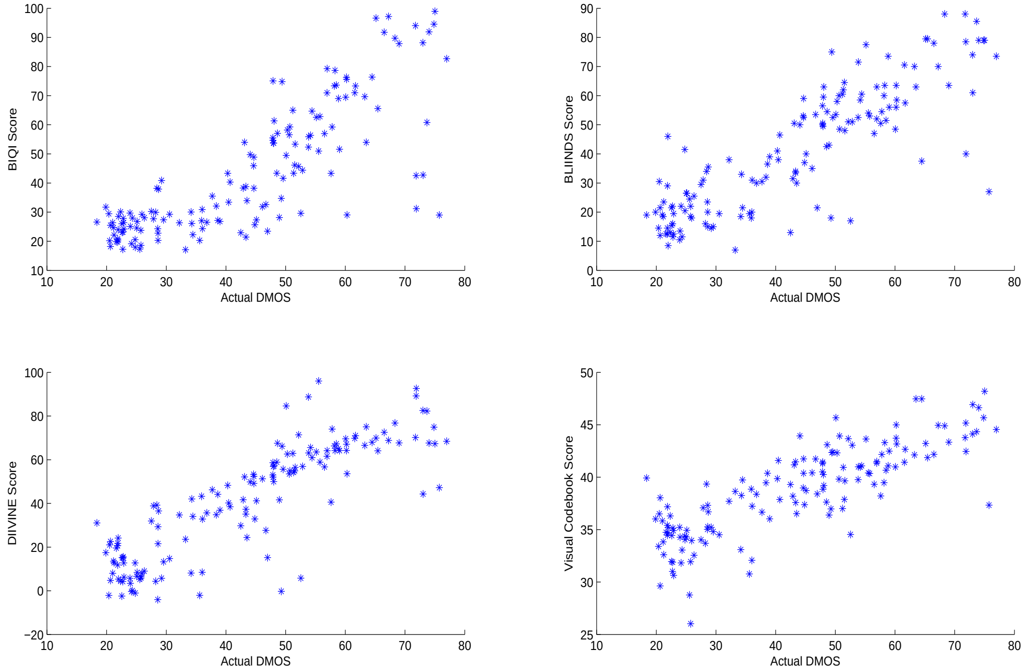
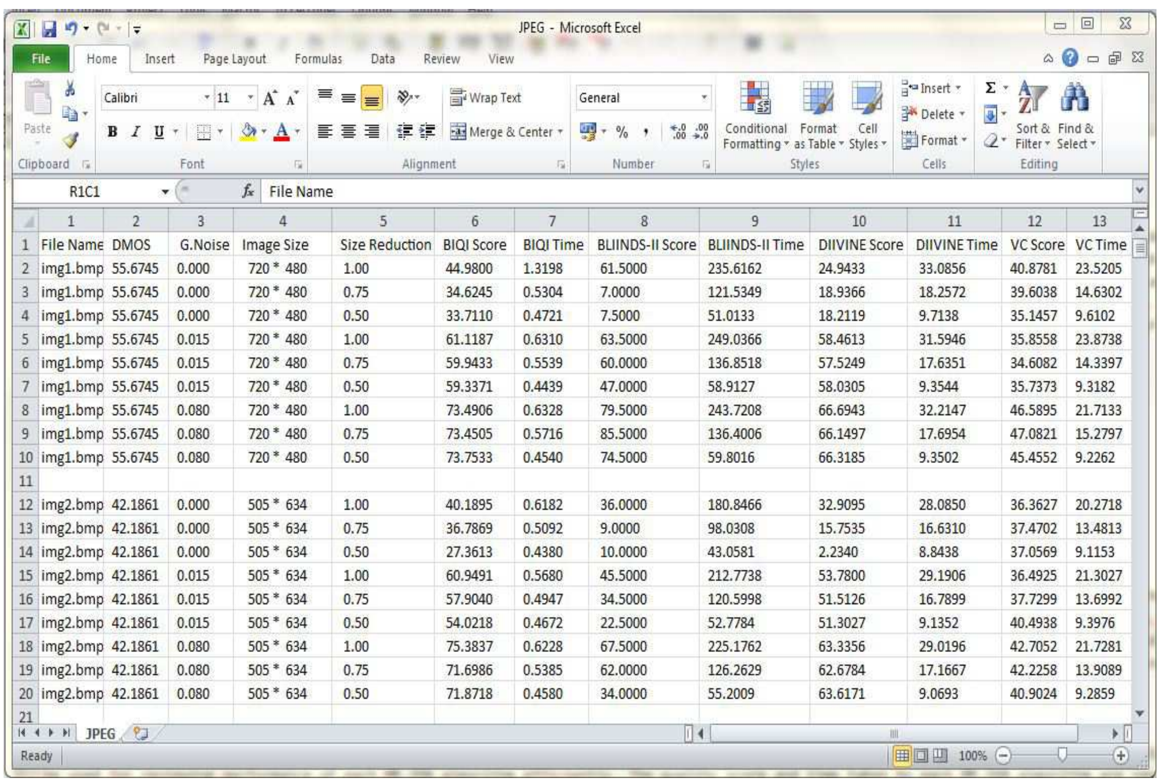


Figure 3.5: Scatter plot : Fast Fading Distortion



	1	2	3	4	5	6	7	8	9	10	11	12	13
1	File Name	DMOS	G.Noise	Image Size	Size Reduction	BIQI Score	BIQI Time	BLIINDS-II Score	BLIINDS-II Time	DIIVINE Score	DIIVINE Time	VC Score	VC Time
2	img1.bmp	55.6745	0.000	720 * 480	1.00	44.9800	1.3198	61.5000	235.6162	24.9433	33.0856	40.8781	23.5205
3	img1.bmp	55.6745	0.000	720 * 480	0.75	34.6245	0.5304	7.0000	121.5349	18.9366	18.2572	39.6038	14.6302
4	img1.bmp	55.6745	0.000	720 * 480	0.50	33.7110	0.4721	7.5000	51.0133	18.2119	9.7138	35.1457	9.6102
5	img1.bmp	55.6745	0.015	720 * 480	1.00	61.1187	0.6310	63.5000	249.0366	58.4613	31.5946	35.8558	23.8738
6	img1.bmp	55.6745	0.015	720 * 480	0.75	59.9433	0.5539	60.0000	136.8518	57.5249	17.6351	34.6082	14.3397
7	img1.bmp	55.6745	0.015	720 * 480	0.50	59.3371	0.4439	47.0000	58.9127	58.0305	9.3544	35.7373	9.3182
8	img1.bmp	55.6745	0.080	720 * 480	1.00	73.4906	0.6328	79.5000	243.7208	66.6943	32.2147	46.5895	21.7133
9	img1.bmp	55.6745	0.080	720 * 480	0.75	73.4505	0.5716	85.5000	136.4006	66.1497	17.6954	47.0821	15.2797
10	img1.bmp	55.6745	0.080	720 * 480	0.50	73.7533	0.4540	74.5000	59.8016	66.3185	9.3502	45.4552	9.2262
11													
12	img2.bmp	42.1861	0.000	505 * 634	1.00	40.1895	0.6182	36.0000	180.8466	32.9095	28.0850	36.3627	20.2718
13	img2.bmp	42.1861	0.000	505 * 634	0.75	36.7869	0.5092	9.0000	98.0308	15.7535	16.6310	37.4702	13.4813
14	img2.bmp	42.1861	0.000	505 * 634	0.50	27.3613	0.4380	10.0000	43.0581	2.2340	8.8438	37.0569	9.1153
15	img2.bmp	42.1861	0.015	505 * 634	1.00	60.9491	0.5680	45.5000	212.7738	53.7800	29.1906	36.4925	21.3027
16	img2.bmp	42.1861	0.015	505 * 634	0.75	57.9040	0.4947	34.5000	120.5998	51.5126	16.7899	37.7299	13.6992
17	img2.bmp	42.1861	0.015	505 * 634	0.50	54.0218	0.4672	22.5000	52.7784	51.3027	9.1352	40.4938	9.3976
18	img2.bmp	42.1861	0.080	505 * 634	1.00	75.3837	0.6228	67.5000	225.1762	63.3356	29.0196	42.7052	21.7281
19	img2.bmp	42.1861	0.080	505 * 634	0.75	71.6986	0.5385	62.0000	126.2629	62.6784	17.1667	42.2258	13.9089
20	img2.bmp	42.1861	0.080	505 * 634	0.50	71.8718	0.4580	34.0000	55.2009	63.6171	9.0693	40.9024	9.2859
21													

Figure 3.6: NR-IQA Algorithms Output Database

Chapter 4

Visual Codebook Algorithm for NR-IQA

4.1 Visual Codebook

Codebook is a set of codes which represents some object. In image processing, codebook can be used to represent an image by some codewords or codes. For example, an image of car can be summarized by the qualities of car like four wheels and four door. These parameters are sufficient to give the information that the image is of a car. So instead of storing the complete image, the descriptors can be stored which represent that image.

The basic block diagram of Visual Codebook algorithm is shown in figure 4.1. As shown in block diagram, the image whose quality is to be evaluated is first divided into patches, then Gabor filtering is applied to the patches to obtain a Gabor Feature Vector, K-Means clustering is done on obtained Gabor Feature Vectors to get cluster centroids which are then used to calculate the quality score of image.

4.2 Gabor Filter

Gabor filter is obtained by modulating a sine wave with a Gaussian function. The frequency and orientation of Gabor filter is dependent on properties of sine wave and Gaussian function. Since image is a two dimensional function, a 2-D Gabor filter is required to extract features from an image.

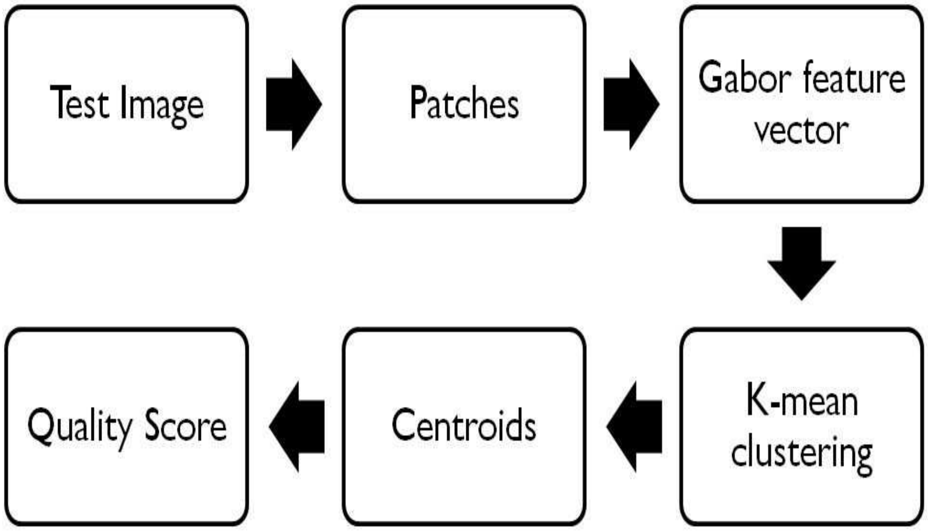


Figure 4.1: Visual Codebook Flowchart

In [12], it is shown that Gabor Filter based features are used to describe the content of an image. The image is divided into patches which are Gabor filtered in different frequencies and orientations. The equation for Gabor filter function [16] in two dimension is shown in equation 4.1 -

$$\Psi(x,y,f,\theta) = \frac{f^2}{\pi\gamma\eta} exp[-\frac{f^2}{\gamma^2}x'^2 - \frac{f^2}{\eta^2}y'^2 + j2\pi f x'] \tag{4.1}$$

where,

$x' = xcos(\theta) + ysin(\theta)$

$y' = -xsin(\theta) + ycos(\theta)$

f - frequency of sinusoidal plane wave

θ - rotation of Gaussian envelop

γ, η - spatial widths of Gabor filter along major & minor axes

Figure 4.2 shows that a 2D Gabor filter function [17] is basically a Gaussian kernel modulated by a sinusoid.

Let the image be represented by $\xi(x,y)$ and gabor filter function by $\Psi(x,y,f,\theta)$, the response of gabor filter to the image is convolution of ξ and Ψ and is given by $g(x,y,f,\theta)$.

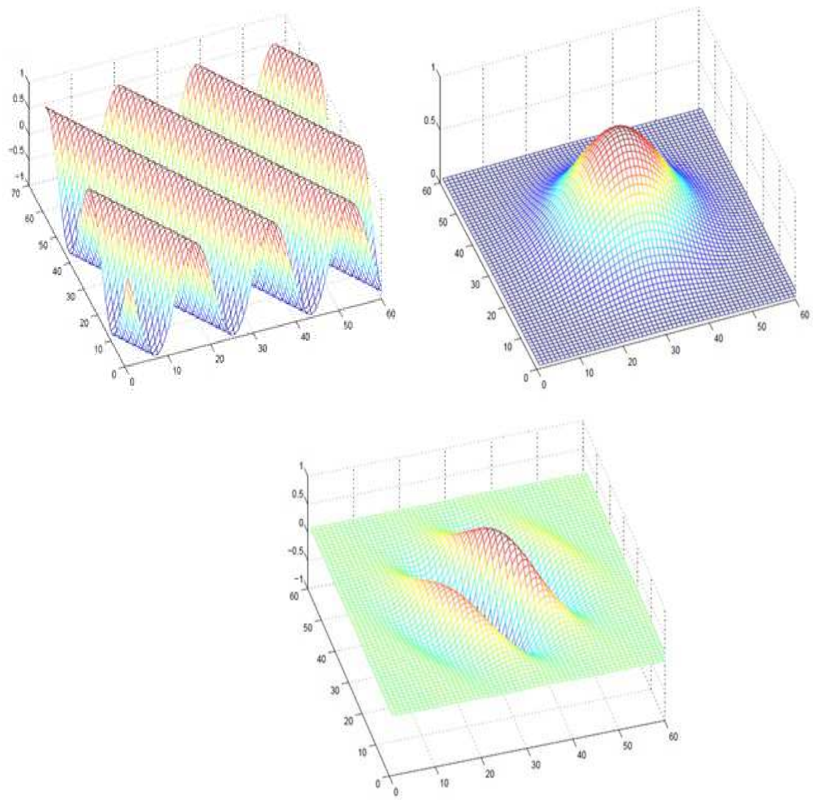


Figure 4.2: (a) 2D sinusoid oriented at 30^o with the x-axis (b) Gaussian kernel (c) Gabor filter [17]

$$g(x, y : f, \theta) = \Psi(x, y, f, \theta) * \xi(x, y) \tag{4.2}$$

The output of filtering action is of same size as that of the input image. If filtering is performed on a patch then the output is also a patch of same size as shown in figure 4.3.

Gabor filtering is performed at five frequencies (1, $1/\sqrt{2}$, $1/2$, $1/2\sqrt{2}$ & $1/4$) and four orientations ($0°$, $45°$, $90°$ & $135°$). Given an image patch, Gabor filtering is done for all possible combinations of frequency and orientation. For 5 frequencies and 4 orientations, there are total 20 combinations. So for each patch, there exist 20 outputs which are of same size as that of patch. Figure 4.4 shows the 20 outputs obtained for all combinations of frequencies & orientations. Mean of these 20 outputs is taken which gives 20 values which are arranged in a vector, similarly variance of 20 outputs is taken which again gives 20 values. Appending the 20 mean values and 20 variance

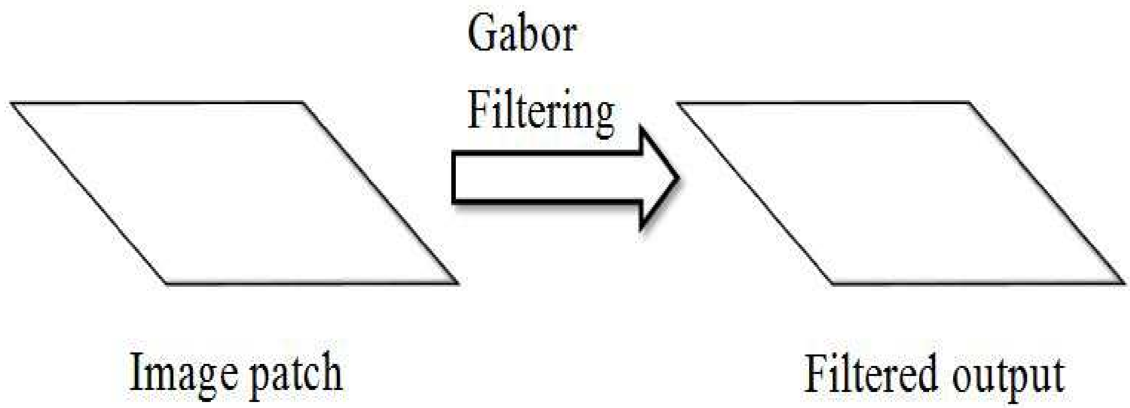


Figure 4.3: Gabor filtering for frequency 'f' & orientation 'θ'

values produces a 40*1 vector which is called Gabor feature vector. Each non-constant patch of an image is associated with a 40*1 vector.

4.3 Codebook Construction

Codebook is constructed by the Gabor feature vectors obtained by image patches. The image is divided into patches of size 11*11, constant patches are removed. Gabor feature vector of these patches is calculated. Each Gabor feature vector can be treated as a point in a 40 dimensional space. A set of Gabor feature vectors is obtained for each image and 200 clusters are formed from this set using K-means [18, 19] clustering algorithm. These 200 cluster centroids are labeled by the quality score of the training image from which the cluster centroids are obtained. Set of training images are used to construct the codebook. Laboratory for Image & Video Engineering (LIVE) [13] database containing 982 images is used to form the codebook.

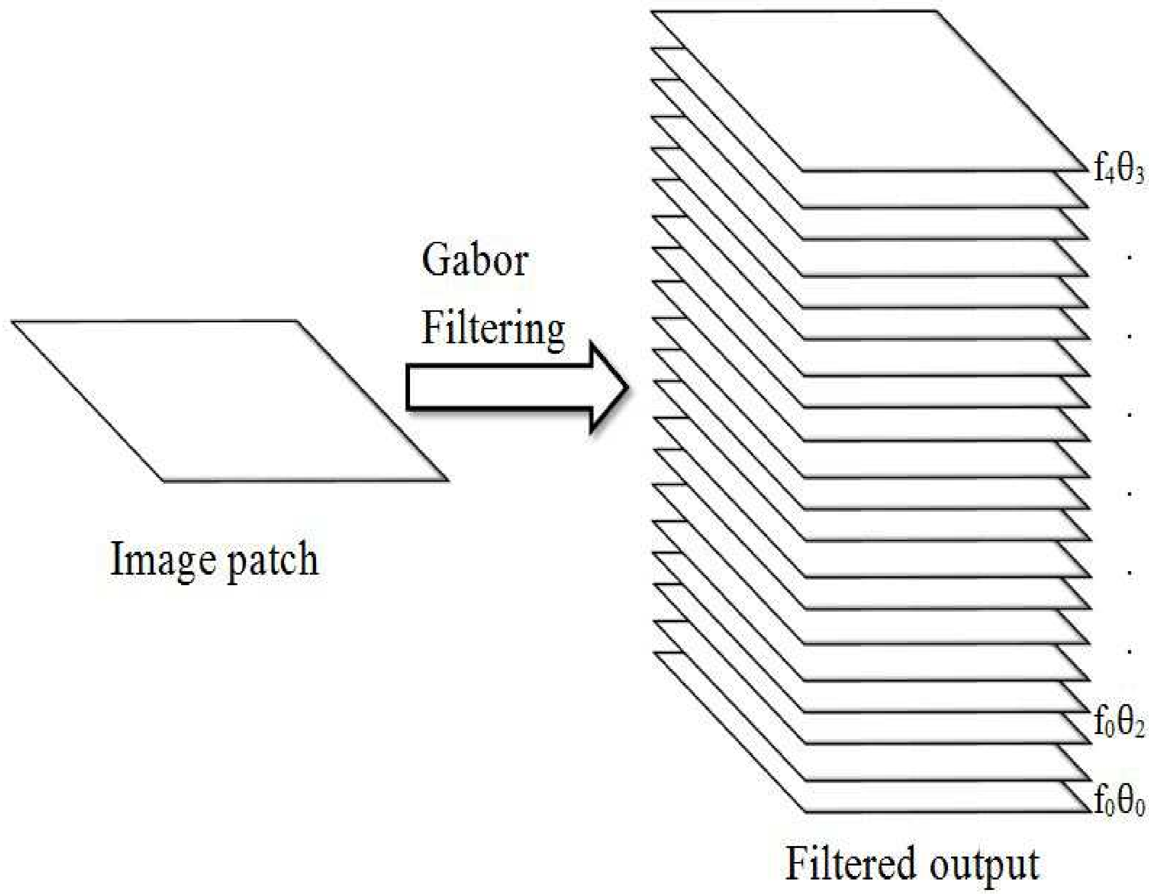


Figure 4.4: Gabor filter output for 5 frequencies & 4 orientations

4.4 Quality Score Evaluation

The test image is divided into patches of size 11*11 and Gabor filtering is performed on these patches. Gabor feature vectors are obtained from these patches. K-means clustering algorithm is used to form 200 clusters of these Gabor feature vectors. Cluster centroids are obtained from these 200 clusters. Nearest neighbour of these 200 clusters in the codebook are found and the average of quality score labels of nearest neighbors gives the quality score of test image.

4.5 Validation on LIVE Database

For validation purpose, correlation between Visual Codebook algorithm output and the DMOS of an image is calculated. LIVE database containing 982 images is used to find correlation. Spearman and Pearson correlation values are calculated to show the performance of Visual Codebook algorithm. Table 4.1 shows the observations on LIVE database excluding reference images.

Table 4.1: LIVE Database: Spearman & Pearson Correlation

Correlation	JP2K	JPEG	White Noise	Gaussian Blur	Fast Fading
Spearman	0.8711	0.8829	0.9008	0.8357	0.8218
Pearson	0.8630	0.8450	0.9038	0.8249	0.8181

4.6 Summary

This chapter explains the working of Visual Codebook algorithm. Two dimensional Gabor filter and how Gabor filtering can be used for Visual Codebook algorithm is discussed. Performance of Visual Codebook algorithm is validated using LIVE database and results are shown.

Chapter 5

Modified Visual Codebook Algorithm

5.1 Need for Modification

In Visual Codebook algorithm [12], the test image is divided into patches of size 11*11. Gabor feature vectors corresponding to each non-constant patch is obtained and the Gabor feature vectors are clustered in 200 clusters using K-Means clustering. While experimenting on Visual Codebook algorithm it is found that the algorithm fails if the test image size is not sufficient enough to produce 200 non-constant patches i.e. if the image does not have 200 non-constant patches then there wont be 200 Gabor feature vectors, which will result in error at K-Means clustering as number of points to be clustered should always be more or equal to the number of clusters in which the points needs to be divided.

This problem can be overcome by either reducing the number of clusters or by reducing the patch size. As mentioned in the reference paper [12], the optimum number of clusters used in K-Means for Visual Codebook algorithm is 200 but nothing is mentioned about the optimum value of patch size.

5.2 Effect of Varying Patch Size

To make Visual Codebook algorithm work for images which does not have 200 non-constant patches of size 11*11, a reduced value of patch size is taken and codebook is constructed using the same procedure as that of original Visual Codebook algorithm.

A reduced patch size is needed but what should be the new patch size is an important question. Also the algorithm constructed using new patch size should perform at par with the original Visual Codebook algorithm. For this purpose, performance of Visual Codebook algorithm is evaluated for patch size of 3*3, 5*5, 7*7 & 9*9. All steps in Visual Codebook algorithm are followed and separate codebook is constructed for each patch size. The modified versions of Visual Codebook algorithm are tested on LIVE database excluding reference images and the correlation values are calculated. Spearman correlation values are shown in table 5.1 & Pearson correlation values are shown in table 5.2.

Table 5.1: LIVE Database : Spearman Correlation for Visual Codebook

Patch Size	JP2K	JPEG	White Noise	Gaussian Blur	Fast Fading	Overall
3*3	0.9454	0.9027	0.9825	0.8759	0.9225	0.9258
5*5	0.9194	0.9200	0.9816	0.9331	0.9178	0.9344
7*7	0.8376	0.8736	0.9578	0.8744	0.8394	0.8766
9*9	0.8118	0.8412	0.9199	0.7362	0.7869	0.8192
11*11 (Original)	0.8711	0.8829	0.9008	0.8357	0.8218	0.8625

Table 5.2: LIVE Database : Pearson Correlation for Visual Codebook

Patch Size	JP2K	JPEG	White Noise	Gaussian Blur	Fast Fading	Overall
3*3	0.9455	0.9057	0.9784	0.8882	0.9163	0.9268
5*5	0.9152	0.9134	0.9768	0.9315	0.9186	0.9311
7*7	0.8330	0.8438	0.9682	0.8118	0.8200	0.8554
9*9	0.8027	0.8061	0.9363	0.7565	0.7806	0.8164
11*11 (Original)	0.8630	0.8450	0.9038	0.8249	0.8181	0.8510

It can be observed from table 5.1 & 5.2 that the correlation value is highest for Visual Codebook algorithm using 5*5 as patch size. Pearson correlation bar plot is shown in figure 5.1 and Spearman correlation bar plot is shown in figure 5.2.

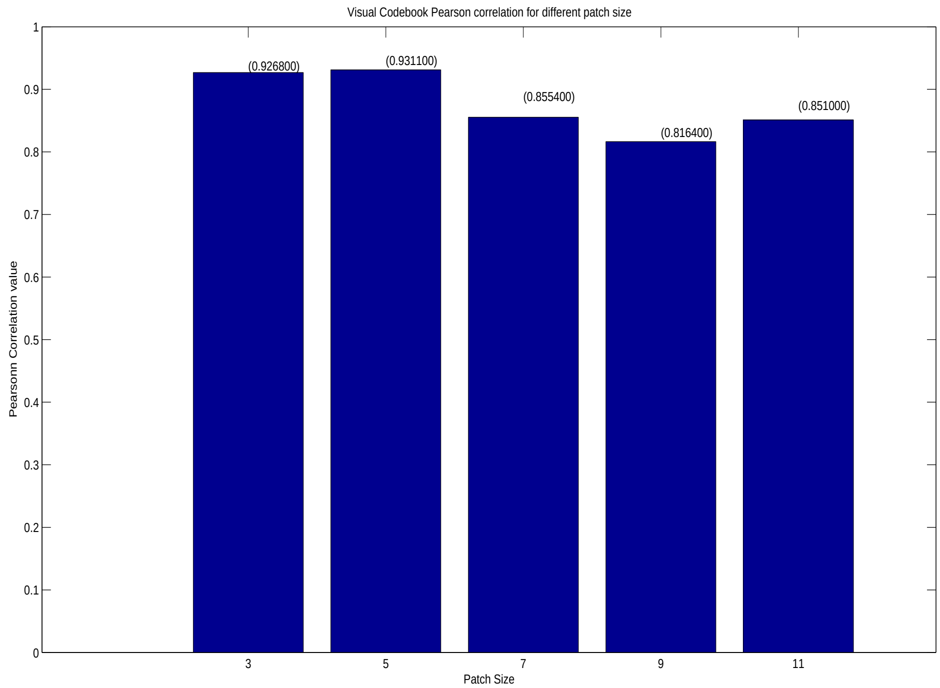


Figure 5.1: Pearson Correlation Barplot

5.3 Variations in Performance of Visual Codebook Algorithm

Errorbar is used to shows the variation of data. The center of errorbar represents the mean value and the length of errorbar represents twice of standard deviation of data. More length of errorbar means that the deviation in data is more. In case of Visual Codebook algorithm, the variation is the difference in correlation value for different type of distortions. In errorbar shown in figure 5.3 & 5.4, X-axis represents the patch size and Y-axis represents the correlation value. For each version of Visual Codebook, correlation values are calculated. The correlation values for each type of distortion are shown in table 5.1 & 5.2. An ideal algorithm should perform equally well for all types of distortions, so that the correlation value for each distortion type is same. In Visual Codebook, as the algorithm is trained only for specific types of distortions, the performance varies depending on type of distortion. More the variation in correlation values, more is the length of errorbar.

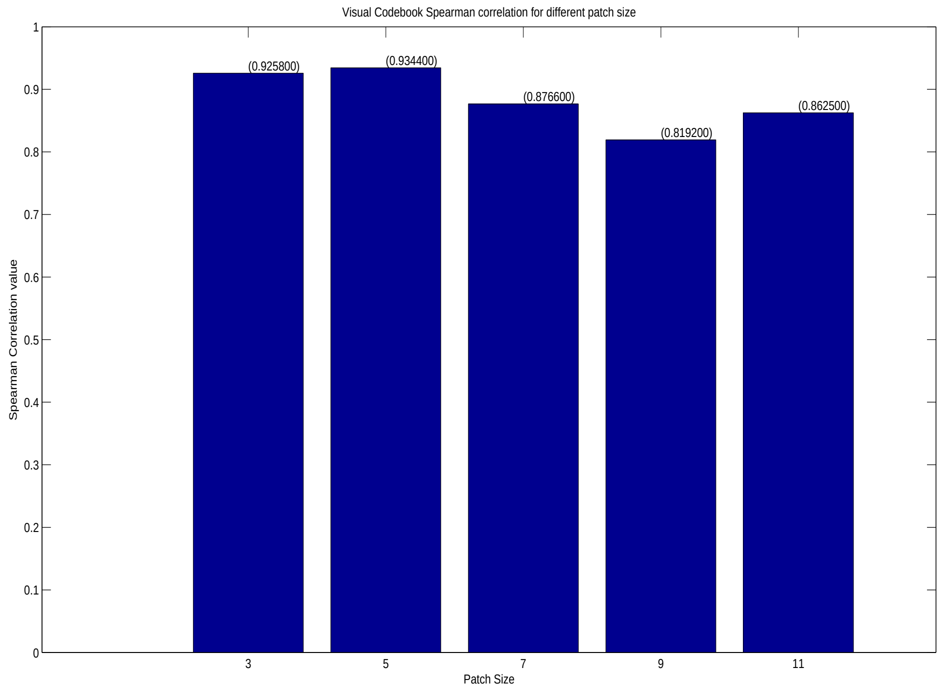


Figure 5.2: Spearman Correlation Barplot

Boxplot also shows the variation of data. The center line in box represents the mean value of data points, the edges of the box represents 25th & 75th percentiles of data points. The extended whiskers represents the extreme data points. Boxplot for Pearson correlation values is shown in figure 5.5 & boxplot for Spearman correlation values is shown in figure 5.6 in which X-axis represents patch size and Y-axis represents the correlation value. A stretched boxplot represents that the spread of data is more and vice versa.

5.4 Algorithm output Versus DMOS

The output of Visual Codebook algorithm and the Differential Mean Opinion Score (DMOS) can be plotted on same graph using scatter plot. Scatter plots are plotted for Visual Codebook algorithm constructed using different patch sizes. The compactness of scatter plot signify a high correlation between output of algorithm and the Differential Mean Opinion Score (DMOS)

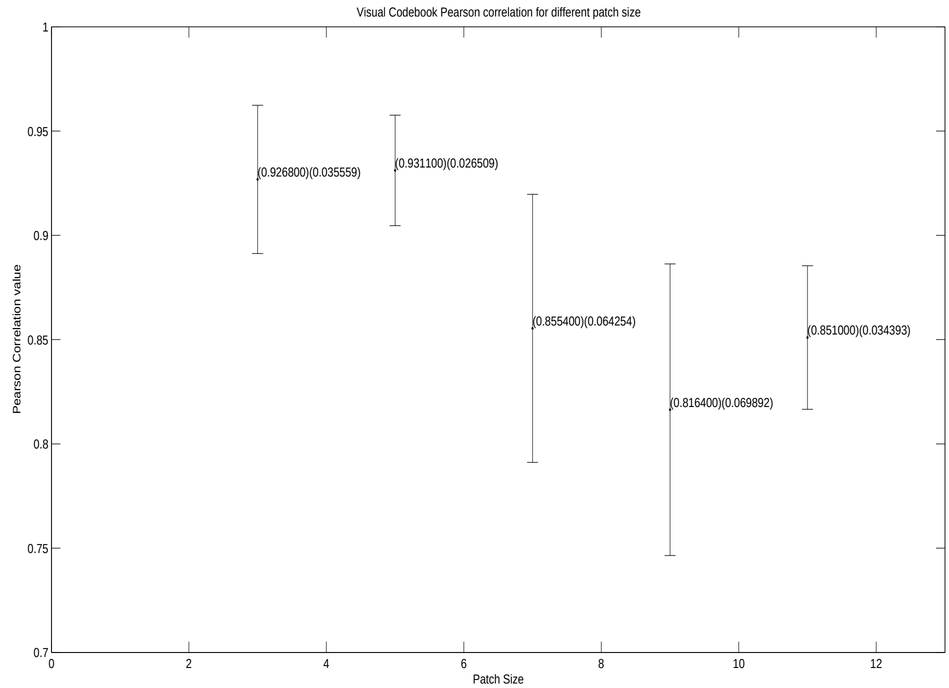


Figure 5.3: Pearson Correlation Errorbar

and vice versa. In the scatter plot, X-axis is DMOS and Y-axis is output of algorithm. Laboratory for Image & Video Engineering (LIVE) database is used to evaluate the performance of Visual Codebook algorithm constructed for different patch sizes. LIVE database contains images with five types of distortions. Scatter plots for distortions namely JPEG2000, JPEG, white noise, Gaussian blur & fast fading are shown in figures 5.7, 5.8, 5.9, 5.10 & 5.11 respectively. It can be observed that the scatter plot for Visual Codebook algorithm constructed using 5*5 as patch size gives a compact scatter plot as compared to others.

5.5 Optimum Patch Size for Visual Codebook Algorithm

It can be observed from table 5.1 & 5.2 that the correlation value is highest for Visual Codebook algorithm constructed using patch size as 5*5. Also the boxplot, errorbar & scatter plots shows that the performance of

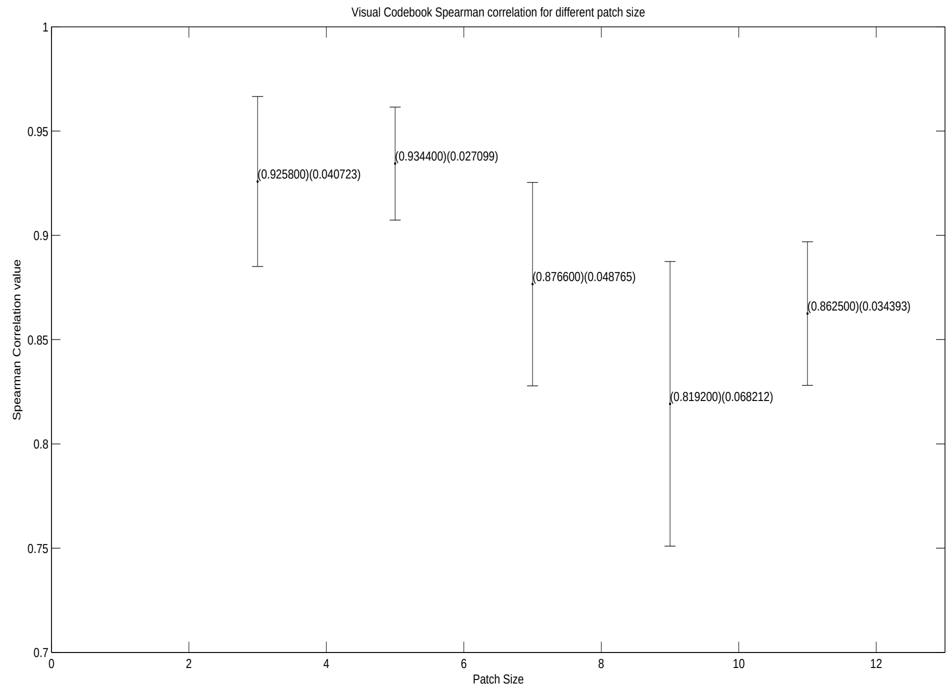


Figure 5.4: Spearman Correlation Errorbar

Visual Codebook algorithm constructed using patch size as 5*5 is consistent for different types of distortions. So it can be concluded that the optimum patch size for Visual Codebook algorithm is 5*5.

5.6 GUI for Visual Codebook

A Graphical User Interface (GUI) is created in MATLAB to show the output of Visual Codebook algorithm constructed using different patch sizes. Figure 5.12 shows the GUI, a 'Browse' button is provided to browse for the test image, 'Evaluate' button calculates the quality scores given by different versions of Visual Codebook algorithm. The test image used is from LIVE database and have a DMOS of 28.0038. The output of Visual Codebook algorithm constructed using different patch sizes is shown in figure 5.12.

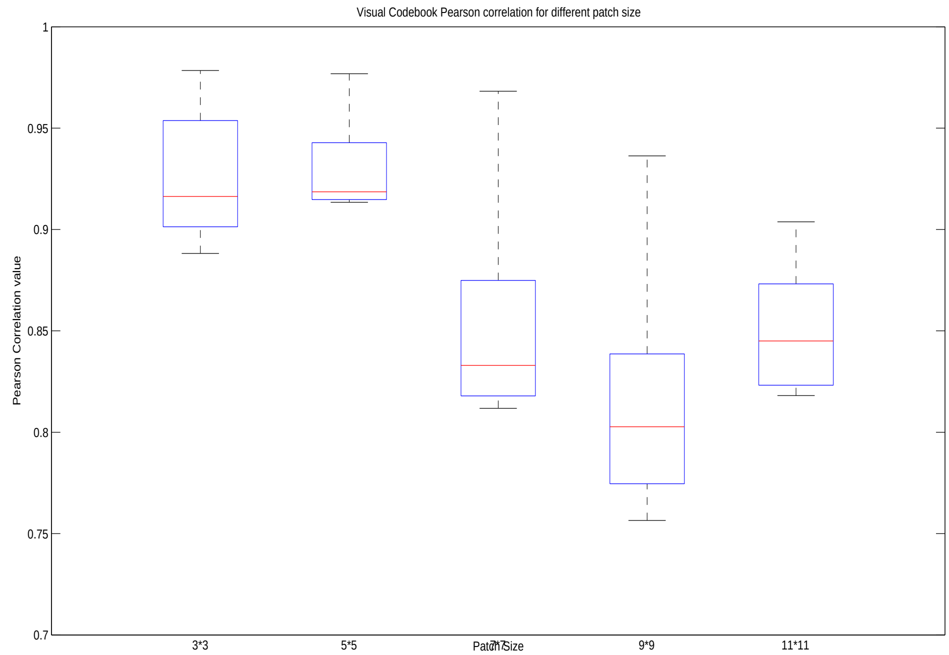


Figure 5.5: Pearson Correlation Boxplot

5.7 Summary

This chapter explains the need for modification in Visual Codebook algorithm and the effect of changing patch size in algorithm on performance of algorithm. Errorbar and boxplot are used to graphically represent the effect of varying patch size. An optimum value of patch size is found to be 5*5. A graphical user interface is created to check the output of Visual Codebook algorithm constructed using different patch sizes.

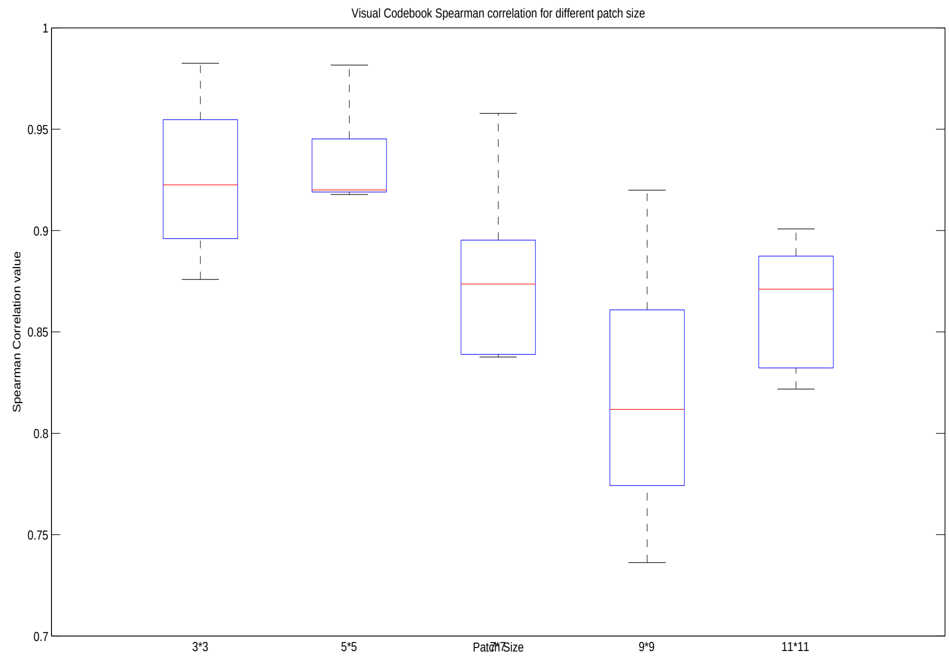


Figure 5.6: Spearman Correlation Boxplot

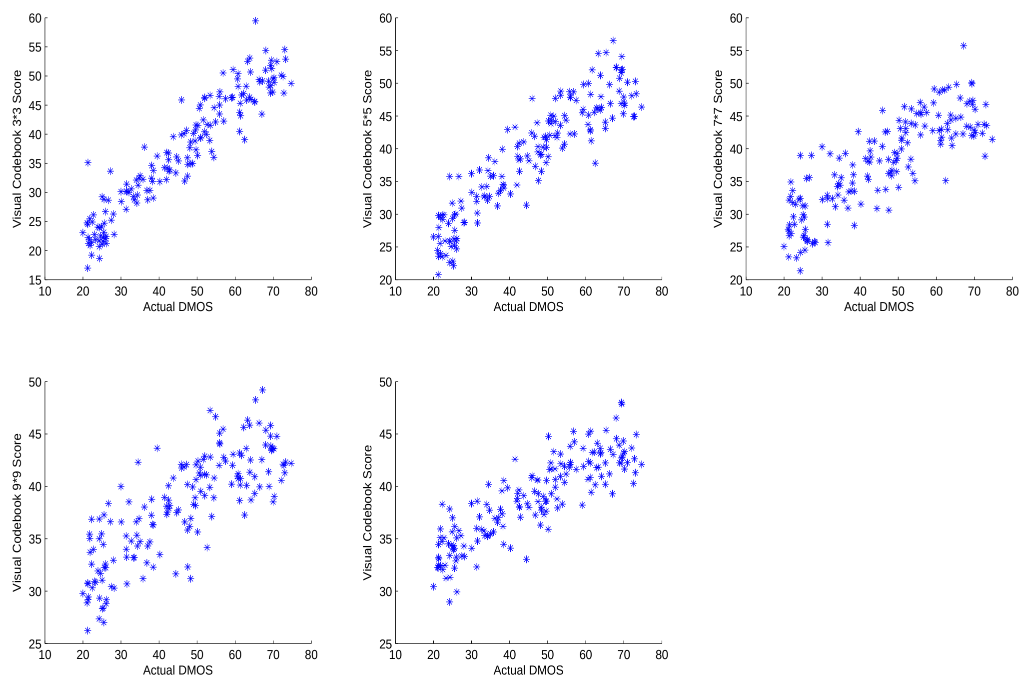


Figure 5.7: Scatter plot : JPEG2000 Distortion

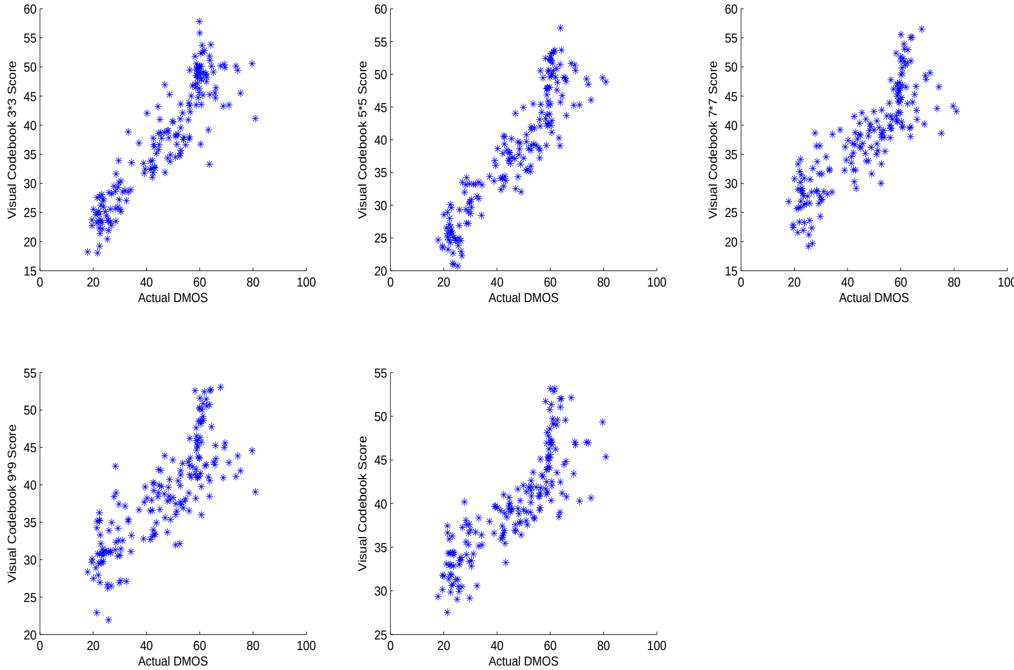


Figure 5.8: Scatter plot : JPEG Distortion

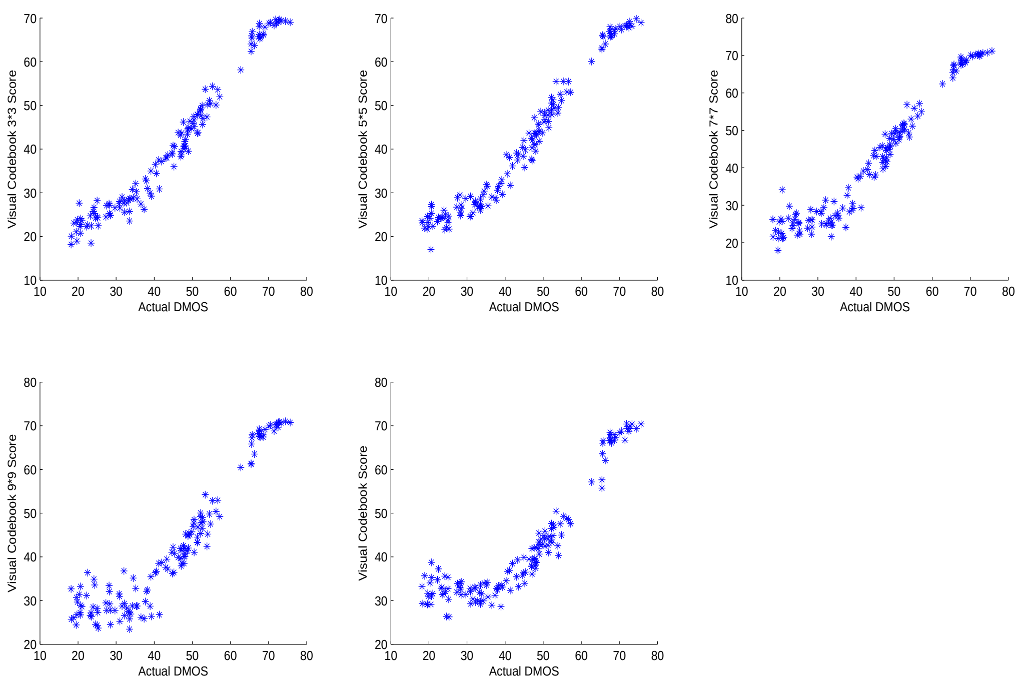


Figure 5.9: Scatter plot : White Noise Distortion

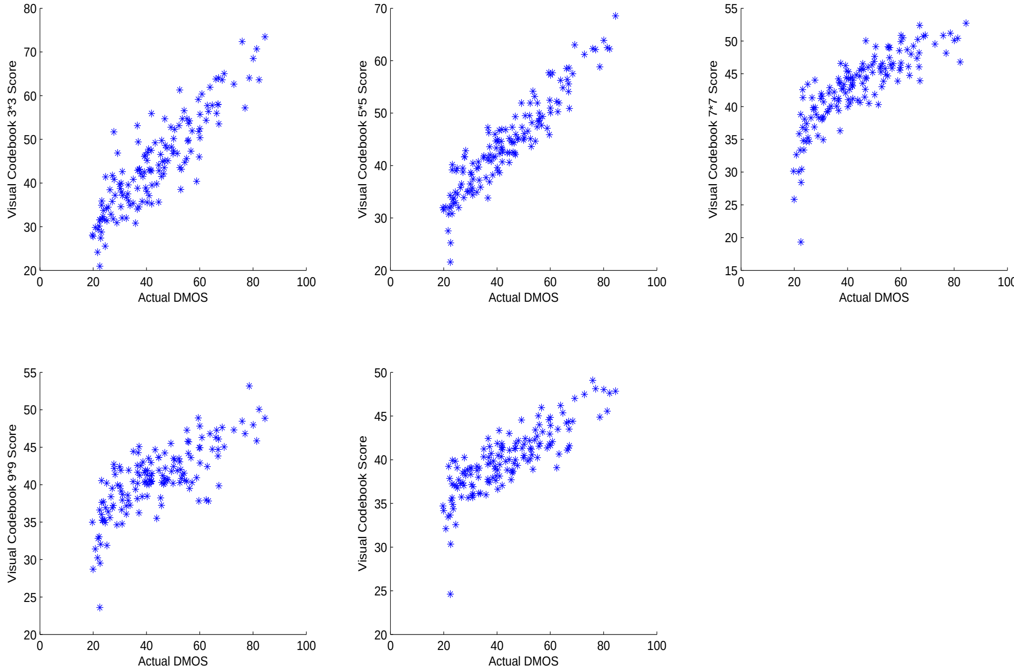


Figure 5.10: Scatter plot : Gaussian Blur Distortion

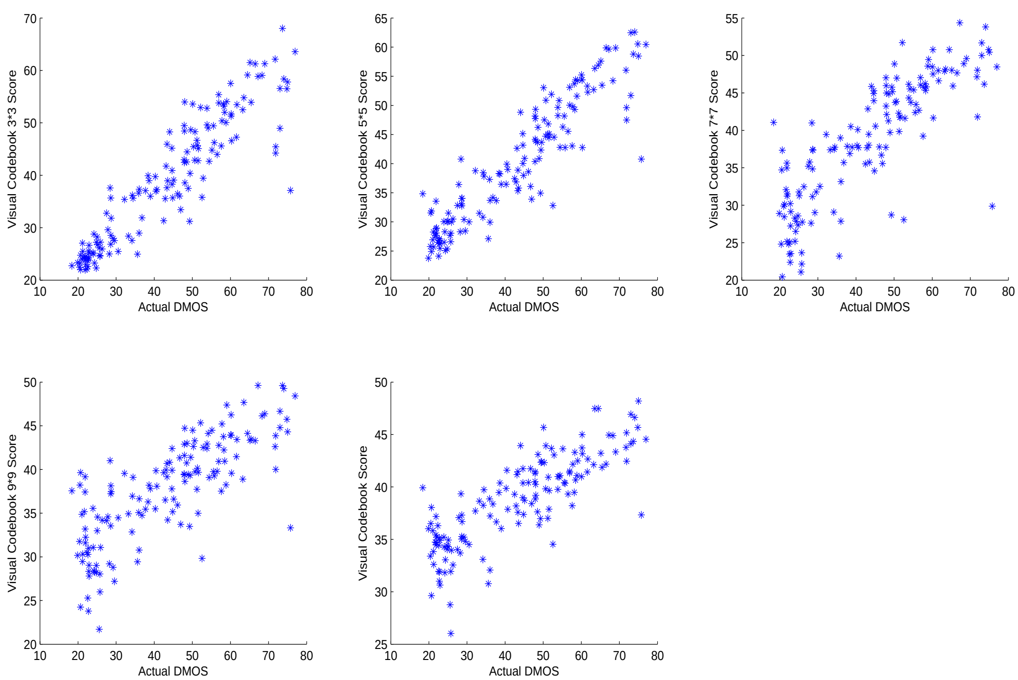


Figure 5.11: Scatter plot : Fast Fading Distortion

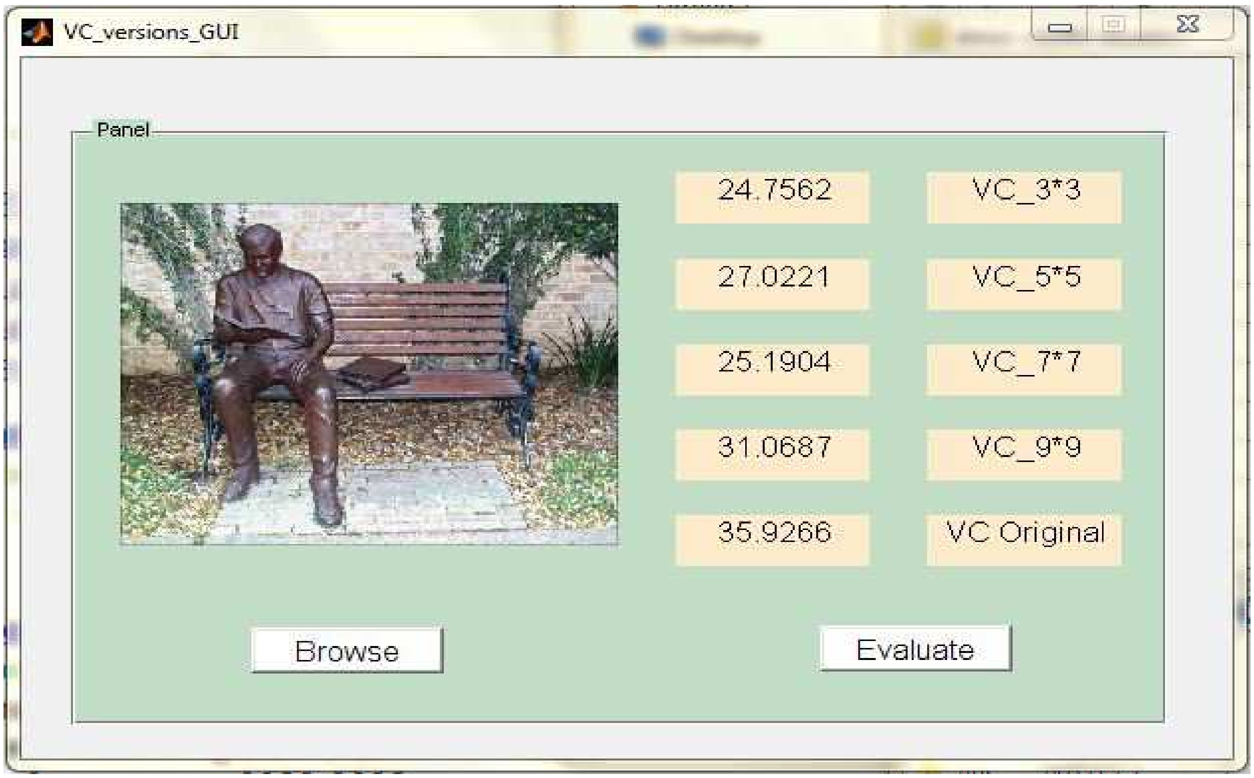


Figure 5.12: Visual Codebook: Graphical User Interface

Chapter 6

Conclusion

Existing No-Reference Image Quality Assessment (NR-IQA) algorithms do not provide exact quality score of an image as no NR-IQA algorithm have correlation value equal to 1 and also the computation time is more which limits its application in real time systems. Another limitation of present NR-IQA is that the algorithms are distortion specific i.e. the algorithm give proper results only if the algorithm is trained for the distortion present in the test image. This can be seen from difference in the correlation values for different types of distortions. A new NR-IQA needs to be developed which can overcome these drawbacks.

The performance of Visual Codebook algorithm can be improved by changing the patch size. A higher correlation value is obtained when the patch size is changed from original 11*11 to 5*5 which also makes the algorithm work on small size images.

6.1 Future Scope of Work

Only accuracy of output of algorithms is discussed here, time complexity calculation and study of theoretical and practical value of time complexity is the future scope of work.

Bibliography

[1] H. R. S. Zhou Wang, Alan C. Bovik and E. P. Simoncelli, "Einstein images with same MSE." `https://ece.uwaterloo.ca/~z70wang/research/ssim/`.

[2] W. Zhou, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[3] M.-J. Chen and A. C. Bovik, "Fast structural similarity index algorithm," *Journal of Real-Time Image Processing*, vol. 6, no. 4, pp. 281–287, 2011.

[4] Q. Huynh-Thu and M. Ghanbari, "Scope of validity of psnr in image/video quality assessment," *Electronics letters*, vol. 44, no. 13, pp. 800–801, 2008.

[5] Z. Wang, A. C. Bovik, and L. Lu, "Why is image quality assessment so difficult?," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, vol. 4, pp. IV–3313, IEEE, 2002.

[6] Z. Wang and A. C. Bovik, "Modern image quality assessment," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 2, no. 1, pp. 103–117, 2006.

[7] S. Gabarda and G. Cristóbal, "Blind image quality assessment through anisotropy," *JOSA A*, vol. 24, no. 12, pp. B42–B51, 2007.

[8] M. Saad, A. Bovik, and C. Charrier, "A dct statistics-based blind image quality index," *Signal Processing Letters, IEEE*, vol. 17, no. 6, pp. 583–586, 2010.

[9] M. Saad, A. Bovik, and C. Charrier, "Dct statistics model-based blind image quality assessment," in *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pp. 3093–3096, IEEE, 2011.

[10] A. Moorthy and A. Bovik, "A two-step framework for constructing blind image quality indices," *Signal Processing Letters, IEEE*, vol. 17, no. 5, pp. 513–516, 2010.

[11] A. Moorthy and A. Bovik, "Blind image quality assessment: From natural scene statistics to perceptual quality," *Image Processing, IEEE Transactions on*, vol. 20, no. 12, pp. 3350–3364, 2011.

[12] P. Ye and D. Doermann, "No-reference image quality assessment using visual codebooks," *Image Processing, IEEE Transactions on*, vol. 21, no. 7, pp. 3129–3138, 2012.

[13] H. Sheikh, Z. Wang, L. Cormack, and A. Bovik, "Live image quality assessment database release 2," 2005.

[14] J. Lee Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The American Statistician*, vol. 42, no. 1, pp. 59–66, 1988.

[15] S. L. Weinberg and S. K. Abramowitz, *Data analysis for the behavioral sciences using SPSS*. Cambridge University Press, 2002.

[16] V. Kyrki, J.-K. Kamarainen, and H. Kälviäinen, "Simple gabor feature space for invariant object recognition," *Pattern Recognition Letters*, vol. 25, no. 3, pp. 311–318, 2004.

[17] V. Prasad and J. Domke, "Gabor filter visualization," tech. rep., Technical Report, University of Maryland, 2005.

[18] J. M. Pena, J. A. Lozano, and P. Larranaga, "An empirical comparison of four initialization methods for the k-means algorithm," *Pattern recognition letters*, vol. 20, no. 10, pp. 1027–1040, 1999.

[19] S. J. Redmond and C. Heneghan, "A method for initialising the k-means clustering algorithm using kd-trees," *Pattern recognition letters*, vol. 28, no. 8, pp. 965–973, 2007.