8-2016

Image Quality Estimation: Soft-ware for Objective Evaluation

He Liu
Purdue University

Graduate School Form 30
(Revised 08/14)

PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance

This is to certify that the thesis/dissertation prepared

By He Liu

Entitled: Image Quality Estimation: Software for Objective Evaluation

For the degree of Master of Science in Electrical and Computer Engineering

Is approved by the final examining committee:

AMY R. REIBMAN

JIANGHAI HU

MICHAEL D. ZOLTOWSKI

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification/Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy on Integrity in Research" and the use of copyrighted material.

AMY R. REIBMAN

Approved by Major Professor(s): _____

Approved by: V. Balakrishnan 07/21/2016

Head of the Department Graduate Program Date

IMAGE QUALITY ESTIMATION:

SOFTWARE FOR OBJECTIVE EVALUATION

A Thesis

Submitted to the Faculty

of

Purdue University

by

He Liu

In Partial Fulfillment of the

Requirements for the Degree

of

Master of Science in Electrical and Computer Engineering

August 2016

Purdue University

West Lafayette, Indiana

ACKNOWLEDGMENTS

First and foremost I would like to express my sincere gratitude to my advisor Professor Amy R. Reibman for her support of my study and research, for her patience, motivation and knowledge. Her guidance helped me finish my research work and thesis writing.

I would like to say thank you to the rest of my committee, Professor Jianghai Hu and Professor Michael D. Zoltowski, for their insightful comments and suggestions.

I am also thankful to my fellow colleges Biao Ma, Chen Bai and Chengzhang Zhong for their help of my work and memorable time we spent in the lab.

Finally I would like to thank my family for their financial and spiritual support, especially for the years that I went aboard.

TABLE OF CONTENTS

Page

Page

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

QE	Quality Estimators
STIQE	Stress Testing Image Quality Estimator (software name)
MLE	Maximum Likelihood Estimation
IQA	Image Quality Assessment
FR	Full Reference
NR	None Reference
RR	Reduced Reference
MSE	Mean Square Error
NSS	Natural Scene Statistic
HSV	Human Visual System
CDF	Cumulative Distribution Function
AWGN	Additive White Gaussian Noise
JPEG	JPEG image compression method
JP2K	JPEG 2000 image compression method
MOS	Mean Opinion Scores
AMT	Amazon Mechanical Turk (online platform)
HIT	Human Intelligence Test

ABSTRACT

Liu, He M.S.E.C.E, Purdue University, August 2016. Image Quality Estimation: Soft-ware for Objective Evaluation . Major Professor: Amy R. Reibman.

Digital images are widely used in our daily lives and the quality of images is important to the viewing experience. Low quality images may be blurry or contain noise or compression artifacts. Humans can easily estimate image quality, but it is not practical to use human subjects to measure image quality in real applications. Image Quality Estimators (QE) are algorithms that evaluate image qualities automatically. These QEs compute scores of any input images to represent their qualities. This thesis mainly focuses on evaluating the performance of QEs. Two approaches used in this work are objective software analysis and the subjective database design.

For the first, we create a software consisting of functional modules to test QE performances. These modules can load images from subjective databases or generate distortion images from any input images. Their QE scores are computed and analyzed by the statistical method module so that they can be easily interpreted and reported. Some modules in this software are combined and formed into a published software package: Stress Testing Image Quality Estimators (STIQE).

In addition to the QE analysis software, a new subjective database is designed and implemented using both online and in-lab subjective tests. The database is designed using the pairwise comparison method and the subjective quality scores are computed using the Bradley-Terry model and Maximum Likelihood Estimation (MLE). While four testing phases are designed for this databases, only phase 1 is reported in this work.

1. INTRODUCTION

1.1 Background

Image Quality Assessment (IQA) has become an important concept in image processing area for decades. In image processing field, IQA are used to detect the artifacts of the processing chain, such as image and video acquisition and display, encoding and decoding and re-purposing and enhancement [1]. This assessment can also be applied in control area such as quality control systems, optimizing the parameters in image related embedded systems [2] etc. Image quality can be easily determined by human subjects but the challenge is that we cannot ask human subjects to examine the quality for every image in every process of imaging systems. For years, researchers have developed algorithms to solve this problem.

Image Quality Estimators (QEs) are algorithms used in IQA which are designed to simulate human beings' judgments on estimating the quality of images. The ideal QE should have the same response as a human when evaluating the same image. These algorithms receive images as input and convert them into numbers that represents image quality. Every QE will compute objective scores based on the provided information of images themselves without human viewing. Based on these algorithms, it is possible to evaluate qualities of large number of images automatically and quickly.

1.2 Three Types Quality Estimators

Based on the information available to QEs, image quality estimators can be divided into three types: Full Reference (FR), Reduced Reference (RR) and No Reference (NR). The difference between these three types of QEs is whether the algorithm has a reference available. FR QEs estimate a distorted image using another unim-

paired image as a reference, RR QEs need some other supporting information instead of a whole unimpaired image, and NR QEs just evaluate the distorted image without extra information.

1.2.1 Full Reference (FR)

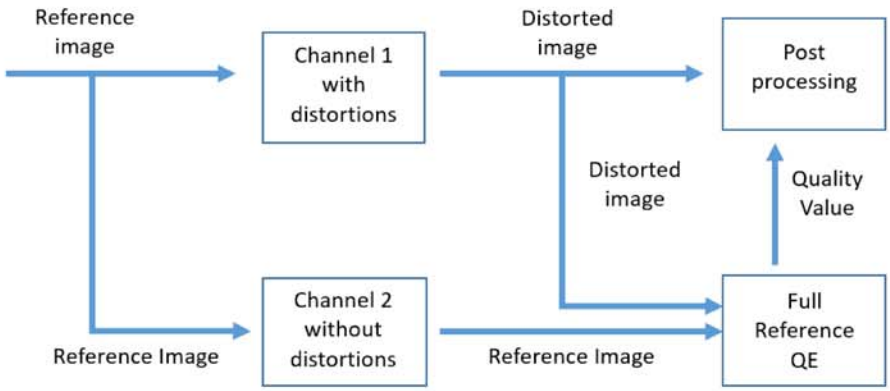


Figure 1.1. FR QE flow chart

Full Reference (FR) QEs need both the distortion image and the reference image as input, and the reference image is normally the original unimpaired image. It can be observed from Figure 1.1 that the reference image will be available without distortion. These QE algorithms make comparisons between two images and most of QEs use the score to show the degree to how the distorted image is similar to the original image. This similarity not only covers the visual perception of images (image fidelity), but the difference between every pixel of two images. If the distorted image is more similar to the original one, it can be considered that this image has a better quality. After the quality value is computed, the value may be used by the system to do some post-process on the distorted image such as image correction. Currently, although there are still some limitations and problems, these FR QEs are most studied and developed.

The Mean Square Error (MSE) is the most widely used mathematical tool in applying FR algorithm without considering the nature of Human Visual System (HVS). By computing the value differences between every corresponding pixel from both images, the MSE can clearly show the pixel differences mathematically. Peak Signal to Noise Ratio (PSNR) is derived from MSE value and PSNR is also a commonly used FR QE. However, MSE does not accurately predict perceived image quality [3]. During the last 15 years, many new FR QEs have appeared, such as Structure SIMilarity index (SSIM) [4], Visual Information Fidelity (VIF) [5], Visual Signal-Noise-Ratio (VSNR) [6], Multi-scale Image Quality Estimation (MIQE) [7], Feature SIMilarity index (FSIM) [8] and so on. These algorithms use different theories: SSIM considers the luminance, contrast and structure feature of nature images [4]. VIF applies information theory and compares the visual information between of two images [5]. VSNR uses wavelet decomposition on images and compare the similarity of different channels [6]. MIQE takes the viewing distance into account for quality evaluating [7]. FSIM [8] focused on comparing the similarity of extracted features of images.

Considering limitations, most of the QEs are designed for gray scale images; it is necessary to transform color images into gray images before computing quality. For this problem, researchers extended the QE working environment to color images. Gupta [9] designed QE working on color images applying HVS characteristics. Zianou and Fella [10] applied the color distortion and gradient similarity as an IQA scheme. Multi-Scale SSIM (MSSSIM) [11] FR QE is improved to work on color images by comparing the Color Just Noticeable Difference (CJND) in CIELAB color space [12]. FSIM algorithm also includes a FSIM Color (FSIMC) index for color images [8].

1.2.2 No Reference (NR)

However, it is not always possible to have both the original and distorted images available. This makes FR QE not suitable for many practical applications. For these kinds of applications, No Reference (NR) images QEs are designed. This type of

algorithms will only evaluate the distorted images themselves and result in quality values. From Figure 1.2, it can be seen that only the distorted images will be available and NR QE will only evaluate the image quality based on the distortion images themselves.

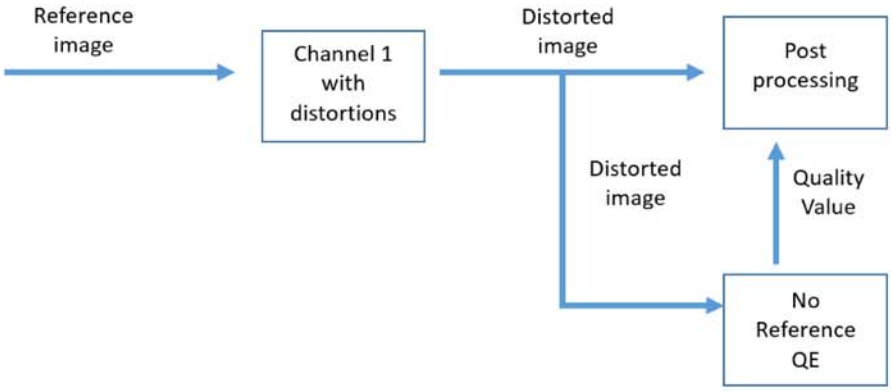


Figure 1.2. NR QE flow chart

Even though human can judge the quality of image just by the image itself, NR QE algorithms are normally difficult to be designed [3]. Comparing to FR QEs, NR algorithms do not have any reference, which means each algorithm needs to compute the score based on the information of the images have. These algorithms are normally designed with analyzing image features and local structures. Based on these features, NR QEs compare these values to a 'general standard', which is designed based on general statistical analysis of natural images from subjective databases. The images in the subjective database have the opinion scores from humans, which indicates which kinds of images have better quality. However, there are a few subjective image databases currently available and the databases themselves have their own drawbacks, such as the limited number of images, limited image resolutions and systematic flaws of subjective test. These limitations of subjective data give a challenge in NR QE developments.

However, researches have made lot of progress on this type of QE. Hemami and Reibman [1] stated a three stage framework for these QEs incorporated with human

visual system. Many algorithms were generated by applying statistic methods, such as BRISQUE [13], which is based on Natural Scene Statistic (NSS) and NIQE [14], which is designed with statistical regularities observed on natural images. Ye et al [15] proposed an algorithm which extracts features from images and combines these filter learning processes based on back-projection. Gabor filters are applied to develop NR QEs to select local features and visual codebook is used to encode the NSS [16]. By applying a log-derivative statistics of natural scenes, Zhang and Chandler proposed DESIQUE algorithm [17].

1.2.3 Reduced Reference (RR)

Considering the features and problems of NR and FR QEs, Reduced Reference (RR) QEs are designed as a combination of NR and FR. Comparing to FR QEs, these algorithms do not need the full unimpaired reference image available at the time of quality estimation. But unlike NR QEs, some reference information is made available to provide a ground truth which assists the algorithm to predicting the quality of the image.

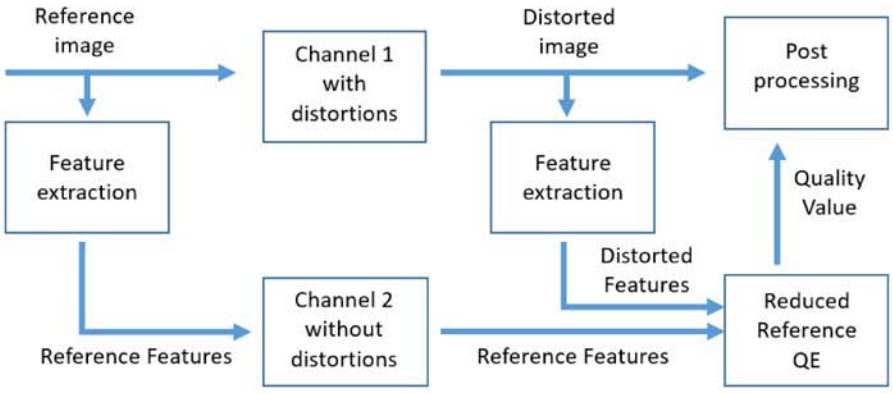


Figure 1.3. RR QE flow chart

These algorithms require inputs for the distorted image as well as some information or features from the reference image. As Figure 1.3 shows, these algorithms will

extract the corresponding information from the distorted images before the images are sent [18]. When images are received, they are compared with available reference information which is transmitted undistorted. These supporting information or features will normally be transmitted in a separate undistorted ancillary channel [19]. Sometimes the features can be transmitted together with distortion images with some error detection procedures, which saves the cost of one distortion-free channel.

The amount of reference information transmitted, or data rate, influences the accuracy of quality prediction and there is a trade off between them [3]. Comparing to FR QEs, RR algorithms do not require the full original image and save the transmission costs. Many researches explored in these algorithms. Wang et al [19] proposed a wavelet domain nature static image model for RR QE. Abdelouahad et al [20] applied Bessel K Forms (BFK) Model for Tetrolet Coefficients in this area. Rehman [21] extended the philosophy of SSIM to RR QE area.

1.3 QE Behaviors and Scores

Different QEs have different working principles and theoretical underpinnings. Some of the QEs make larger values to represent better quality images, but some of them do the opposite way. The range of different QE values are also different; some ranges from 0 to 1 but some are wider. In addition, the distributions of QE values in their ranges are also different, and most of them are not uniformly distributed. Based on these situations, it is a challenge to compare different QE algorithms' behaviors under different cases. To deal with this problem, one method is mapping different QE scores into the same domain by using mathematics methods, such as non-linear fitting [22]. For fitting methods, instead of traditional logistic regression and polynomial regression, Han et al [23] proposed monotonic regression which has a better performance. For the case of evaluating more than one QE, a strategy was proposed to jointly test objective scores of different QEs, in order to objectively identify which kinds of images show different quality scores with others [24].

When judging the behavior of a QE, several aspects need to be considered, such as evaluating stability, time consumption and most importantly accuracy. When using a QE algorithm code, it is important to guarantee the code can run successfully in different environments, and with different kinds of inputs, for example when feeding larger sized image on Mac system, or computing two images that are exactly the same. Algorithms may slow down or fail in some cases. Time consumption is another key issue because some widely used QEs are quite time consuming especially for high resolution images. Some may take more than 10 seconds for one image. This will be a big problem when processing a large number of images in huge image systems.

The accuracy is normally judged by comparing results to subjective databases. Referring to subjective databases, it can be determined whether the QE make the same decision as human assessments. The correct and false decisions are defined in [25], such as false ranking, false tie, false differentiation and correct ranking and correct tie. However, because the number of subjective databases are limited and these databases have different drawbacks, researchers try to make judgments purely based on objective QE data. Instead of applying subjective scores, Xue et al [26] applied unsupervised learning method called Quality-Aware Clustering (QAC), on Blind Image Quality Assessment (BIQA) without using human scored images. Ciaramello and Reibman [27] also proposed a systematic stress test method by setting proxy QE to replace subjective values.

1.4 Subjective Databases

The subjective databases are designed for image QE assessments. Most databases are generated based on reference (high quality) images or source images. Most of these images are natural photos from daily lives and some databases also include artificial images. The reference images are purposely impaired by different distortions, such as blur, noise and so on. For every distortion type, the reference image is impaired into several distortion levels, from lightly distorted to heavily distorted. These

images will be viewed and ranked by human subjects based on their quality. After processing the participants' opinions, each image will receive a number which is used to represent its quality ranking, or subjective ranking. When using the databases, other researchers can apply different QE algorithms on the distorted images in the database and compare the QE scores to the subjective ranking for the purpose of evaluating the QE performance.

There are several key issues about subjective tests: the format of the subjective test, the quality judging interface and the scoring system. The format of subjective test governs how the test is processed. It is possible to invite human subjects to come into the lab (In-lab test) and evaluate the quality of images with supervision. This allows researchers to guide naive participants to follow the test procedures. But this format is limited by human factors like time issues, which limits the scale of the test (small number of subjective viewers involved). Compared to this, crowd-based quality evaluation [28] is not limited by these factors. Through the Internet, the number of participants can be large and the speed of the test is much faster. But researchers cannot control their testing devices and cannot check the test requirements like viewing distance and test duration. Despite the fact that participants are not supervised, it has been proved that crowd sourcing subjective can deliver accurate and repeatable results [29].

About the quality judging interface, when human subjects are viewing images, some databases show the distorted image only, but some tests present both the distorted image together with its reference image for participants to compare. This will lead to different results, because participants will feel more confident to compare two images than to judge one image only. About the scoring system of the test, most of the subjective tests apply the absolute scoring scheme. The participants are provided with some choices, from 'good' to 'bad' or from '5' to '1', and they are required to pick one choice for each image. This method may cause problems for example different participants have different judging methods [30]. In addition a low confidence in scores can cause troubles of the test [31]. Paired comparison is another method

which only requires the participants to choose a better quality image from a pair of distortion images. These choices can be converted to scores by mathematics models such as Bradley-Terry and Thurstone-Mosteller [32].

Currently, some widely used databases are LIVE [33], CSIQ [34], TID [35] etc. These databases have some limitations, for example the image resolutions are limited to 512*512 and the distortion levels are also limited by only 5 to 7 levels. Apart from these problems, almost all the databases only present the final Mean Opinion Score (MOS) to every image, but do not report the 'raw data', such as every participants' opinions of every image or the preferences of every pair of images. Different statistical models are applied, which makes different raw test data generate different MOS. To guarantee the diversity of user's assessment, the Standard deviation Opinion Score (SOS) and was proposed [36]. If the databases publish the raw data, researchers who uses the databases can generate their MOS themselves by different methods or models.

1.5 Image Distortions

Nowadays, when we use images in our daily lives, the images are almost always impaired. The 'channel' from the natural image signal to Human Visual System (HVS) includes front-end digital processing, communication channel, back-end digital processing and display [37]. During each process of the channel, impairments occurs, such as the color loss in taking a picture, bit loss during the transmission, compression during storage etc. Before studying these distortions, researchers use models and mathematical tools to analysis natural images [38] and HVS. Some HVS models are Spatio-Temporal model [39] and 'Standard model' proposed in [40]. Based on the study of image quality and HVS, it is learned that human eyes have different tolerances for different types of distortions. So it is important to generate different types of distortions models and impair images purposely to create impaired images as needed.

Some of the most commonly known and relatively severe distortions are blur, noise, JPEG compression. Blur may result from the loss focus or unstable cameras when taking photos. In image quality area, this distortion can be modeled as Gaussian Blur. Noise often happens in dark region of images or night photos. This distortion can be modeled by Additive Gaussian White Noise (AWGN). JPEG is the most commonly used image compression format but sometimes to get a higher compression ratio, the image will be heavily compressed causing severe heavy distortions.

1.6 Thesis Summary

In this thesis, we present software to evaluate image QEs, and we also create ground truth subjective scores through subjective tests.

For the purpose of analyzing QE algorithms, some QE evaluating software were developed, such as IVQUEST [41]. This IVQUEST software is written in MATLAB and designed to evaluate the performance of not only image QE but also video QEs. However, IVQUEST is limited by the number of source images and further analysis of scores. In this work, chapter 2 discusses about a new image QE evaluation software framework based on the idea of [27]. Our new software is able to load in unimpaired images and generate distortions purposely on this images. It also supports most current QE algorithms and any new QE metrics can be added into the system. With QE scores, the software also provides basic statistical analysis functions.

The most important function of QE algorithms is to make accurate prediction of image qualities. The ground truth of these algorithms are from subjective data. Currently, only a few databases are widely used, such LIVE, CSIQ and TID. But most of them only have small sized images and the number of reference images are quite limited. In addition, most of the databases do not provide the raw subjective data but only publish the analyzed data.

Because of the limited subjective data, the objective QE test becomes more important for QE performance evaluation and developing new QE metrics. In chapter

3, we present a published software which is designed to purely evaluate the QE performances objectively. Although no subjective data is involved in this software, the software is still able to report the strength and weakness of QEs in several aspects, such as separability, invariance and monotonicity.

Although generating objective analysis software is one good method of QE evaluation, it still cannot solve the limited subjective data problem essentially. To directly solve this, new subjective data are necessary to be built. In chapter 4, a novel subjective image databases is designed and generated. This new databases include 60 images with resolution of 1024*1024. Four main distortion types are applied and each distortion type is divided into 50 distortion levels. The subjective data is gathered from an online survey and an in-lab test. Both tests apply paired comparison method and the ranking results are computed using Bradley Terry model with MLE method.

2. QE ANALYSIS SOFTWARE

This chapter is focusing on the developed QE analysis software. This software is designed based on some basic functional software modules. In section 2.1, the five basic modules are explained and discussed in detail. After that, some testing results of the software functions are presented in section 2.2.

2.1 Software Basic Modules Description

The software is implemented based on different functional modules. Every module has its own focus and each of them can be used individually. There are five modules included in this software: image impairment, QE calculation, subjective database access, objective QE scores maps and statistical analysis. These five functional modules are explicitly explained in section 2.1.1, 2.1.2, 2.1.3, 2.1.4 and 2.1.5 respectively. Every module is saved in a different script file and each script file includes all the detailed functions used in the module.

Most of the functions in this software are implemented in Python environment. In the QE computation module, the software uses QE metrics which are developed by other researchers. Most of published QE metrics are implemented in MATLAB codes. In order to compute this QE scores, a MATLAB software is necessary, but the Python functions in QE computation section is able to automatically call and run MATLAB codes in background. The supported MATLAB QE metric codes are saved in the sub folder of the software. Also users is also able to add new QE metrics into this software.

This software is written on Windows platform, with the coding environment of Python 2.7. The Python and MATLAB connection module is for Windows only so that this software now only works on Windows platform. Several widely applied

image processing modules such as PIL [42] and openCV [43] are used in this software. Apart from Python modules, a third party software called 'Kakadu' [44] is called for JPEG2000 compression.

2.1.1 Image Impair

The goal of the image impairment section is to purposely generate images distortions with a certain distortion type and to a desired distortion level. In this software, four main supported distortion types are Additive Gaussian White Noise (AGWN), Gaussian Blur, JPEG compression and JPEG2000 compression. These four distortion are chosen because they are mostly happened in daily lives. Blur and noise distortion can be easily generated from camera taking process [45], while JPEG and JPEG 2000 are the most widely applied image compression method currently. Also, most QEs are only capable for one of these four distortion types [46]. Every distortion has a certain parameter, in order to control the different distortion levels. The parameter, or knob values, can be generated purposely with special purposes, such as exponential distributed knob values may generate equal interval of distortions for certain cases.

AGWN is generated by merging the reference image with a random white noise image, which is generated by a 2D Gaussian distribution. The mean of the distribution is set to be zero and its variance is used as the knob value. A larger variance value provides greater increases of each noise pixel value, and results in a noisier image. The random seed for each random distribution is recorded for the purpose of reproducibility.

Gaussian blur is generated by convolving the reference image with Gaussian kernel. This process is implemented by an openCV function. The size of kernel is set to be 83*83 by default which is designed to suit large sized images. The distortion knob for blur is determined by the variances of the Gaussian kernel. Different from noise distortion, the blur kernel is designed based on the size of input images. Comparing to smaller sized images, large sized images need a relatively larger variance value, in

order to make same leveled images have same visual quality. We add a ratio of the knob value in order to control the image size. This ratio is the sum of height and width of the input image divided by 1024. The 1024 is the sum of 512 by 512 sized images, which means 512*512 sized images are used as reference to benchmark the knob value of different sized images.

JPEG compression is implemented by using the saving function from PIL module. In that function, there is a quality choice before saving JPEG file. The quality value 100 represents the best quality (not compressed) and 1 shows the lowest quality (heavily compressed).

Kakadu software is used for JPEG2000 compression. This third party software has a quality choice of bit rate of the output image before saving. Smaller bit rate provides more compression. However the same bit rate compression causes different visual experience on different sized images. Larger sized images need smaller bit rate to be compressed comparing to small resolution images.

Apart from these four distortion types, the impair image function is also able to compute convolutions of images. The convolution kernels can be defined by the user. The software also can generate a constant image for specific uses. Besides, some other helper functions are designed. They are used for repeatedly calling the core impairment function, for the purpose of automatically impairing large number of images. These helping functions are useful for generating large scaled image databases.

2.1.2 Subjective Database Access

The functions in this module are used to access or grab information or image from subjective databases. Four database structure are supported in this software, LIVE [33], CSIQ [47], TID2013 [48] and a self-generated database structure. This function loads in the directories of databases and outputs a data structure. This structure is designed as a large matrix. Each row represents one distortion image and each column represents one attribute of that image. These attributes are designed

to include all the information provided in the database, such as distortion type, distortion level, image location, MOS score and so on. Some attributes are specially added for specific database, such as 'org' attribute for LIVE, which represents whether the image is original and therefore is not distorted. For other databases, these special attributes are set as 'N/A'. The image structure generated in this module is the standard input for further analysis functions in this software.

When doing the statistical data analysis, images from more than one database is used for one test. For these cases, there are some supporting functions to modify this data structure, such as adding new images, deleting existing images or merging two structures. Through these functions, the user may combine all the distortion images from different databases and generate a testing image pool for further analyzing. More functions are designed for more advanced selecting data, for example selecting all the images with blur distortion or with same reference image from an image pool.

2.1.3 QE Calculation

The functions in this module are used to compute different QE values based on their codes. The software allows users to add their own QE metric codes and the newly added codes can be either MATLAB codes or Python codes. All these codes need to be included in a sub-folder inside the software and registered in one 'py' file called 'load_QE_info'. There are 23 QEs currently supported in this software: ADM [49], BIQI [50], BRISQUE [13], CORNIA [51], DIVINE [52], FSIM [8], GSM [53], IFC [5], IL-NIQE [54], IWSSIM [55], MAD [47], MIQE [56], MS-SSIM [11], NIQE [14], PSNR, PSNR-HVS-M [57], RFSIM [58], SRSIM [59], SSIM [4], UQI [60], VIF [61], VSNR [6], VSI [62].

The main QE computation function in this module reads in lists of directories of distorted images (and their corresponding reference images) in order. The input is the structure that is generated in database access module. When the data are loaded successfully, the $load\_QE\_info$ function is called to give instructions of how the QE is

computed. The $load\_QE\_info$ function stores the detail information about every QE, such as the directory of algorithm code and the QE's maximum or minimum values. For example, some QEs only accept gray images so a RGB- to-gray converting process is needed before QE computing. A MATLAB engine is called by this function and the image information is transferred from Python into MATLAB functions which return the result back to Python. The calculated QE scores are saved in a TXT file in the same order as the input images list. Some QE algorithms are quite time-consuming especially when the number of testing images is large or the image resolution is large. For some QEs, this process sometimes may take several days to finish.

2.1.4 Objective QE Scores Mapping

This goal of this module is to compute the best fitting function based on the scatter points of QE scores and MOS values. The process helps to map all different QE values into the same domain for further analysis. In every database, each image has one MOS score but more than one QE objective scores. The mapping process assigns different objective values to the same subjective score domain based on the best fitting function. In addition, the mapping process is able to merge data from different databases by projecting subjective scores from different databases to the same QE domain. There are four standard logistic mapping functions included in this software $logistic\_5$ (2.1) from Sheikh et al [63], $logistic\_4$ 1 (2.2) and $logistic\_4$ 2 (2.3) from J.149 [25] and $logistic\_4$ 3 (2.4) from [47].

$$y = t_0 \left(\frac{1}{2} - \frac{1}{1 + exp\left(t_1 \left(x - t_2\right)\right)} \right) + t_3 x + t_4 \tag{2.1}$$

$$y = t_0 + \frac{t_1}{1 + t_2 * exp\left(x + t_3\right)} \tag{2.2}$$

$$y = t_0 + \frac{t_1 - t_0}{1 + t_2 * exp\left(x + t_3\right)} \tag{2.3}$$

$$y = t_0 + \frac{t_1 - t_0}{1 + exp\left(-t_2\left(x - t_3\right)\right)} \tag{2.4}$$

Among the four equations above, y represents the value in subjective score domain, and x represents the objective QE values. The parameter vector t represents the target values to be trained for each QE algorithm. It can be observed that most of these equations have 4 parameters involved, except the equation (2.1) which has 5 parameters and it contains one more linear term.

In this module, two kind of functions are included; the training functions and mapping function. The training functions apply optimization methods such as $curve\_fit$ and $fmin$ from SCIPY module [64]. Every QE value and subjective score pair is fitted using one of the four equations and result in the parameter vector. After the training process is finished, the parameters are saved into a TXT file on the local disk. The mapping function collects the parameters of the specific QE with one fitting function and scales the QE values to subjective data domain. The scaled QE is also saved into TXT files.

2.1.5 Statistical Analysis

This part of the software aims to analyze subjective data and objective QE scores, in order to test the QE behaviors. There are two main analysis tests included, deciding the agreements between QE scores and subjective values (misclassification analysis) and resolving power analysis.

Before the analysis program runs, the software needs to find all the difference values between every possible pair of images. Both subjective data and objective QE scores are converted into difference vectors for further analysis. The detail steps for this process are shown in algorithm (4).

1. Suppose there are n images Img_1 to Img_N and each image has a MOS score mos_i and its QE value qe_i

2. Take one value mos_i and subtract all other values in vector mos, and add these difference values into a new mos_diff vector.

3. Take other values in mos and repeat step 2. Store all the difference values in mos_diff vector.

4. Repeat step 2 and 3 for qe vector and update the qe_diff vector.

The outputs of this pre-procedure are two difference vectors. Notice the length of qe_diff and mos_diff are both $\frac{n*(n-1)}{2}$. The misclassification analyzes the agreement of the QE and MOS. An ideal QE should have a high agreement rate. The test compares both the subjective scores and objective QE values for a pair of images and decide whether they both represent the same quality level. Both subjective scores and QE values have their own threshold values which are decided by users. These threshold value represents the range that the program believes two values are in the same quality level. For example if the QE value threshold is 0.2, then any images have their QE score difference less than 0.2, are considered to have same quality level. The program allows more than one QE threshold values but there is only one subjective score threshold. Based on the comparison between difference vectors and thresholds, the program does computations of both MOS values and QE scores and lead to the result of disagreement percentages, or misclassification. Detailed steps are shown below.

1. Suppose there are vectors qe_diff and mos_diff of length $\frac{n*(n-1)}{2}$. MOS threshold is a constant MOS_T and qe threshold is a vector qe_T with length l.

2. Compare every value mos_diff_i with MOS_T and assign 0 if MOS_T is larger. If MOS_T is smaller, record the sign (1 or -1) of the difference. Save 0, -1 or 1 in a vector called MOS_sign.

3. Generate a vector D of length 6, six positions represent Correct Decisions (CD), Correct Rank (CR), Correct Tie (CT), False Rank (FR), False Differentiation (FD) and False Tie (FT) respectively.

$$D = [CD, CR, CT, FR, FD, FT]$$

4. Take one value qe_T_i from qe_T vector and compare the difference with vector qe_diff like step 2. This result in a new vector is called qe_sign.

5. Take a pair of values qe_sign_i and MOS_sign_i, decide the following situations. Add 1 to the corresponding position in vector D.

   FT: MOS_sign_i equals 0 and qe_sign_i is not 0.

   FD: MOS_sign_i is not 0 and qe_sign_i equals 0.

   FR: The produce of MOS_sign_i and qe_sign_i is -1.

   CT: both MOS_sign_i and qe_sign_i are 0.

   CR: the signs of two values are same but not 0.

   CD: CT+CR.

6. Normalize 6 values by dividing them by $\frac{n*(n-1)}{2}$, and $n*(n-1)$ for CD.

7. Repeat step 3 to 6 for every value in qe_T vector and combine all the D vectors into a l by 6 matrix.

The result matrix presents the percentage of misclassification that this QE generated with specific threshold values. A better QE should have relatively higher CT, CR and CD values. This function can also focus on images with same distortion type or same reference image, by selecting all input images within that category. For example the software can only do this test on images with JPEG distortion or images generated from the same reference image.

Another statistical test is resolving power of objective QE scores. Resolving power is defined as the minimum change of QE values to make a significant change in subjective data and the computation algorithm is published on [65]. This function

divides the QE difference values into 19 groups or bins, and computes the average probability based on corresponding subjective scores. By finding the 95% and 90% probability confidence level, the corresponding resolving power is calculated.

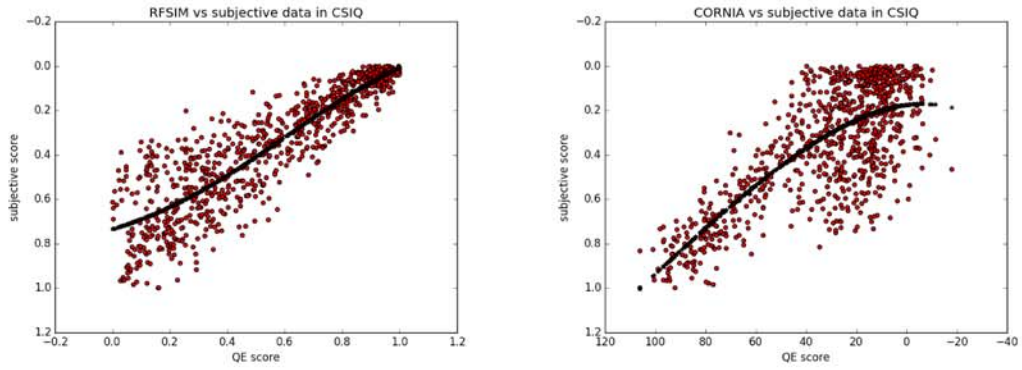
2.2 Testing results

After the software functions are explained, in this section, three main testing results of the software are presented: the QE versus MOS mapping plots in section 2.2.1, misclassification test in section 2.2.2 and resolving power test 2.2.3.

2.2.1 Subjective Data & QE Score Mapping

The result presented in this paragraph is generated based on module 'Objective QE scores mapping'. Two of the mapping plots are shown in Figure 2.1 as examples. The subjective data in CSIQ database is applied to test RFSIM and CORNIA (all other plots are shown in appendix). These plots are generated with the help from subjective database loading module, QE computing module and mapping module. By loading the subjective database, the distorted images are paired with their reference image and form an image pair list. Also the MOS scores are loaded with the same order as the image pair list. Then the QE computing module loads in the list and outputs in a list of QE values. The red scattered plot is based on the MOS score list vs the QE value list. Then mapping module applies mapping functions (equation (2.1)) to generate a best fitting function, which are shown in black dots.

It can be observed from these two plots that the mapping function is almost around the center of scatter points vertically. In CSIQ database, smaller MOS values (y axis) represents better quality, and the direction of QE values (x axis) are inverted so that the image quality increases. As a result, the scatter plot with a proportional trend shows a better performance of the QE metric. It can be observed that RFSIM shows a relatively proportional plots and its mapping function is basically able to show their relations because the variances are limited. However, by observing CORNIA,



(a) RFSIM (b) CORNIA

Figure 2.1. Two mapping plots, RFSIM and CORNIA in CSIQ database

the scattered plots are proportional when image qualities are low, but get messy when image quality is better. Its mapping function is still a good fit based on the points, but the fitted line cannot provide an accurate representation of original data, especially for high quality leveled images. This error shows that non-linear mapping may hide information and this problem needs to be solve in further research.

2.2.2 Misclassification Analysis

The result in this part is generated based on the misclassification program in module 'statistical analysis'. The following Table 2.1 shows the misclassification analysis of 'RFSIM' and 'CORNIA' with CSIQ database, which are consistent with Figure 2.1. There are six distortion types in this database and the misclassification fractions are computed with every distortion type. Instead of the four distortions commonly used, CSIQ also has two extra distortion types: fnoise (or Additive Gaussian Pink Noise) and contrast. In the table, RFSIM has correct decisions (CD) rate of more than 85% except noise distortion. Basically it is possible to say RFSIM is a reasonable QE metric because it agrees with most of the subjective data in CSIQ. While CORNIA has good CD rates for blur, JPEG and JPEG 2000 distortions, but lower

rates for other distortions. This means CORNIA is probably not suitable for analyzing images with contrast and noise distortions. It is still possible to compare these two QEs with different databases, like LIVE and TID, because there are probably testing errors inside the CSIQ subjective databases, which influence the accuracy of subjective scores.

Table 2.1.
Misclassification analysis for RFSIM and CORNIA with CSIQ

QE	Distortion	CD	CR	CT	FR	FD	FT
RFSIM	noise	0.723	0.723	0.0	0.034	0.243	0.0
RFSIM	blur	0.910	0.910	0.0	0.087	0.001	0.002
RFSIM	fnoise	0.885	0.885	0.0	0.11	0.0	0.0
RFSIM	JPEG	0.896	0.896	0.0	0.099	0.003	0.002
RFSIM	JPEG2000	0.907	0.907	0.0	0.082	0.010	0.0
RFSIM	contrast	0.906	0.906	0.0	0.092	0.002	0.0
CORNIA	noise	0.503	0.475	0.028	0.206	0.216	0.076
CORNIA	blur	0.882	0.882	0.0	0.117	0.001	0.0
CORNIA	fnoise	0.596	0.596	0.0	0.318	0.001	0.086
CORNIA	JPEG	0.842	0.842	0.0	0.105	0.003	0.051
CORNIA	JPEG2000	0.817	0.815	0.002	0.098	0.009	0.077
CORNIA	contrast	0.602	0.602	0.0	0.396	0.002	0.0

2.2.3 Resolving Power

The resolving power is a metric to determine the accuracy of a QE algorithm. This value is defined by the difference of QE values which shows to what degree two images have a statistically difference between each other, normally at 0.95 significance level [25]. The resolving power function in the software is implemented based on the method mentioned in [25]. There is an example result in Figure 2.2 showing this metric, with VIF comparing to JPEG compressed images in CSIQ database. In the plot, each red point in the figure represents a pair of images compared to other images. The X axis shows the absolute value of the QE difference of that pair and Y axes represents the significance level that one image in the pair is better than another. Two green lines are the separation for 95% of points over the right or lower side, and the blue line represents the average mean significance level for each region of delta

QE values. An ideal QE should have a relatively small QE difference to show the difference between two images, which do not have the same level of quality. In this plot, VIF has a 0.38 QE difference to show two images that have 95% probability that are not on the same quality level. This means VIF need 38% of its QE value range to proof two images are clearly to be a different quality level. /this means the accuracy is not high enough.

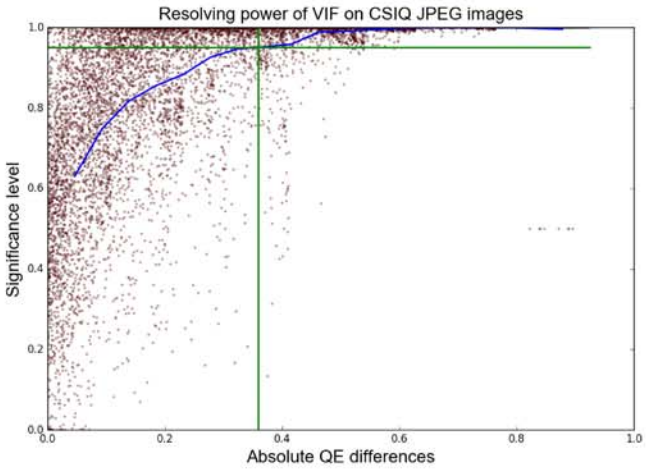


Figure 2.2. Resolving power plot of VIF with JPEG images in CSIQ database

3. SOFTWARE OF STRESS TESTING IMAGE QUALITY ESTIMATORS (STIQE)

This chapter talks about a published QE analysis software STIQE [66] based on the functional modules from chapter 2. First, we will discuss the software structure of STIQE and its function modules in section 3.1. Then we will focus on an experiment of testing QEs with STIQE. The design and generation of testing images are discussed in section 3.2. Finally the testing experiment procedures and results are presented in section 3.3.

3.1 Software Design

In this software introduction section, we will first provide its general information in section 3.1.1. Then two main functional modules are explained in detail: QE computation module in section 3.1.2 and objective analysis module in section 3.1.3.

3.1.1 General Description

Based on these five modules of programs mentioned in section 2.1, a high level software is created to focus on specific QE behavior analysis. The following paragraphs discuss one published software package: Stress Testing Image Quality Estimator (STIQE) which is written based on the modules introduced in section 3.1.1.

This developed software is designed to purely measure the behaviors of image QE algorithms objectively, which is not based on any subjective databases. This software focuses on evaluating the QE behaviors on three aspects: separation between undistorted and badly distorted images, invariance of pixel shifted images and monotonicity of images with gradually increased distortion levels. The software mainly

consists of two major sections: the QE computing section and analysis section. For QE computing section, each QE is analyzed by four test methods and all the computed QE values are saved into a P file on local disk. After the P files is generated, the software generates a brief report on how these QEs work for every test aspect in an Excel table. Then the generated P file are used by the analysis section, in order to create a thorough and detailed analysis of the behaviors of target QEs in each test.

3.1.2 QE Compute Module

There are two main functions in computing QE section, QE computation function and statistical report function. The first function loads in a folder of high quality (reference) images and a list of QEs. Then these images are impaired with different analysis requirements and the corresponding QE values are computed. The generated distortion images are deleted after the QE values are computed by default and the calculated QE values are stored to local disk in P file. The second function gives a brief report of the results. It loads in the QE values and performs a percentage analysis. The report of the analysis is saved in an excel file. The general structure of this section is shown Figure 3.1.

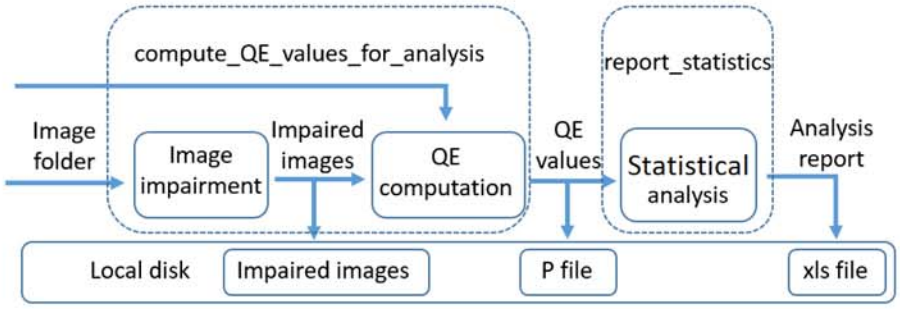


Figure 3.1. The general structure of external analysis software

The software uses four test methods: testing undistorted images, testing badly distorted images, the invariance test and monotonicity test. Each testing method

generates its own typed distortion images and do the QE computations separately. All testing methods are described in detail below.

1. Undistorted image test will not use any distortion images. The input images are high quality images, it is possible to assume they are undistorted images. Then the reference images will be regarded as distortion images and are computed for QE values. Then it compares the QE values to the upper bound of the QEs, if the actually QE values shows the images have a relative high quality, then this QE have a good behavior for undistorted images. This test is most used for NR QEs.

2. Badly distorted images test will impair each reference image to the most distorted level for every distortion type. Then the QE values of these badly distorted images will be computed. If the QE values represents a relative low quality, it is possible to believe the QE has a good behavior of badly distorted images.

3. For QE invariance test, each reference image is impaired by JPEG and JPEG 2000 with a level of 30. Then both reference image and impaired image are cropped and forms 9 pairs, with one pixel shift each pair. All 9 pairs are little bit smaller than original pair and each pair has little difference with others. But generally they are visually the same. Then these 9 pairs are computed by QEs and each pair results in one QE value. After that, the software compares the maximum difference value over 9 QE values of cropped pairs to the value of original pair. If the difference is almost 0, it means the QE believes the images are almost the same like humans, and the QE has a good behavior of invariance test.

4. In QE monotonic test, each reference image is impaired to 50 distortion levels for one distortion type, with level 50 represents the heaviest. All distortion images will be computed for their QE values. Among 50 QE values from one reference image, the software computes the maximum non-monotonic QE difference and maximum non-monotonic distortion level difference. All the QE difference values and distortion level difference values are collected. Then their percentile values are computed. Lower QE difference and lower distortion level difference value shows a better behavior of monotonicity.

All the data from four different test methods are saved in one P file. The P files is a Python dictionary structure with several layers, as shown in Figure 3.2.

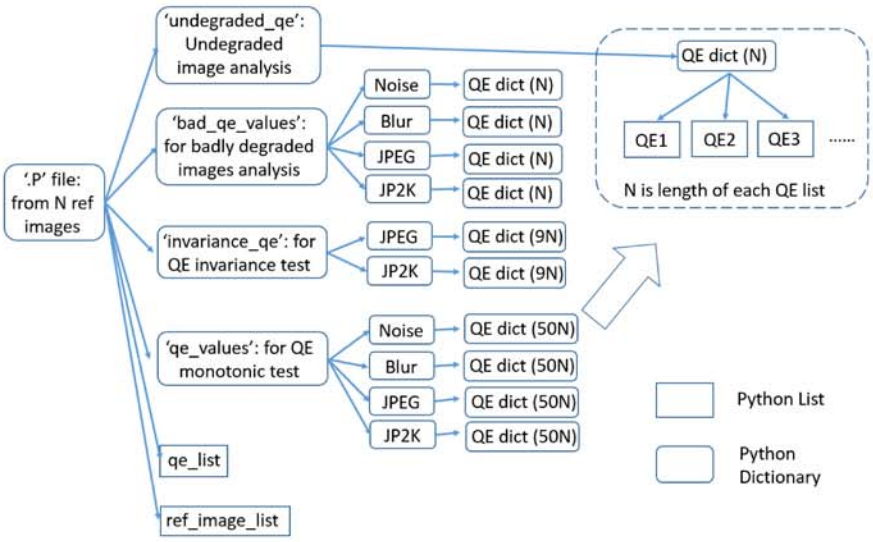


Figure 3.2. The structure of P file

The first layer contains a sub-directory for each of the four types of analysis. This layer also includes the basic information such as the names of the QEs and directories of the reference images to be tested. The second layer contains information about distortion types for each analysis. The 'undistorted image analysis' has no distorted images, and the 'QE invariance test' only has JPEG and JPEG-2000 distortions. In the third layer, all tree nodes have the same structured dictionary, which contains the QE scores for each QE type. The p-file can grow with the number of QEs, and more QEs can be added to an existing p-file.

3.1.3 Objective QE Analysis Module

The analysis section is based on P files. There are two kinds of analysis functions, one QE analysis function and pairwise QE comparison analysis function.

For one QE analysis, the QE values for undistorted images and badly distorted images are compared to determine whether the QE algorithm can make a good sepa-

ration of high and low quality images. The Cumulative Distribution Function (CDF) of high quality and low quality images are plotted on the same plot. Kolmogorov-Smirnov (KS) [67] statistical test is also applied, in order to judge whether the distributions of the QE values are overlapped with each other. For invariance test, the maximum difference of the QE values of 9 pairs are computed, which shows the QE's minimum resolution to distinguish the quality of two images. This means two images which have the QE value difference smaller than the resolution, are considered to have the same quality level by this QE. Monotonicity test determines whether the QE has a monotonic behavior with increasing distortion levels. If not monotonic, a pair of maximum QE difference and maximum difference level is computed for each reference image per distortion type.

QE pairwise comparison analysis compares one QE behavior with all other QEs that are saved in the same P file. This comparison does not provide a ground truth for judging the behaviors for the target QE because there are no subjective data involved. However, if one QE disagrees with most of others, it is possible to believe that there may be some errors for the target QE. The comparison procedure includes 3 steps, pooling, pairing and comparing. Detailed procedure is shown below.

1. Image vector I are loaded into the software for analysis.

2. Based on the pooling choice, I are separated into sub image vectors $Isub1$, $Isub2$, and so on.

3. In side each sub vector $Isub\_i$ with length n, the function compares the QE value of each image_$Isub_i$ a with all other images in $Isub\_i$ to form pairs $Psub\_i$, such as $Psub\_i(a) = (Isub\_i(a), Isub\_i(b))$, where $0 < b < n$ and $a \neq b$. The length of the pair should be $n(n-1)/2$.

4. Repeat step 3 for every sub image vector $Isub\_i$ and get pairs $Psub\_i$. (merge $Psub\_i(a)$, where $0 < a < n$.)

5. Merge all pairs $Psub\_i$ and forms one long pair list $P(i) = (A(i), B(i))$, where A and B are lists of images in the pairs.

6. For each $QE(k)$ in the QE list, find which image has a better quality based on their scores. The result is recorded in a vector R, where $R(k\_i)$ is either 1 if $A(i)$ has better quality than $B(i)$, or 0 otherwise.

7. Repeat step 6 for all QEs.

8. Select one target $QE(k)$ and compare $R(k)$ with all other R and compute the sum of all disagreed pairs with each comparison $C(k)$.

In step 3, the software does the pooling procedure based on user inputs. There are four choices with whether same or different distortion type, same or different reference image. For simplification, the software use '0' to represent 'different' and '1' for 'same'. For example flag '01' is used to represent the pooling case with different reference image and same distortion. So images with same distortion type are gathered to form pools, such as 'noise' pool and 'JPEG compression' pool. Finally the vectors $C(k)$ shows disagreement percentages for $QE(k)$, and the smaller values in $C(k)$, the less disagreement that QE has with others.

3.2 Image Preparation

This section mainly focuses on the designing and generating the test images for the use of testing QEs. First we will discuss the source of images in section 3.2.1. Then the distortion levels of reference image are explicitly explained in section 3.2.2. After that we will also cover the method of processing different sized images in section 3.2.3.

3.2.1 Raw Reference Images

The original images are selected from researchers' photo collections and they are taking by normally cameras in daily lives. All the images have a resolution higher than 2048 pixels both vertically and horizontally. Before the images become reference images in the database, they are cropped and down-sampled. Based on the content of the images, different areas (square) are selected and taken out of the image. These cropped image squares all have the resolution with 2048*2048. Then by down-sampling by 2 and 4, these images become 1024*1024 and 512*512 sized. These two groups of images form the reference images of the database and all these images are stored in the lab server, which are publicly assessable. All reference images are shown in Figure A.1 in appendix.

3.2.2 Image Distortion Levels

For better analysis of the QE behaviors, it is important to generate distortion images with different distortion levels. As designed in this image data set, each distortion type has 50 different distortion levels. The 512 sized image data set is also designed that all level 50 (worst quality) images from distortion types have the same quality level. This makes human viewers do not have a clear preference that one distortion typed images are much better than another, for example participants will not say level 50 JPEG image is better than level 50 blur image. To ensure this, VIF [61] QE is applied to calculate the quality of the level 50 distorted images between four distortion types. By changing the distortion knob values (parameters), the VIF quality scores of four level 50 images are determined to be the same, around 0.1 to 0.2. Then some distortion knob functions are modified from linear equations to exponential equations, in order to make the quality of four distortion typed image as linearly distributed as possible.

Figure 3.3 shows the averaged VIF scores of all distorted images in the data set. It can be observed that four distortion types almost drops to the same VIF value at

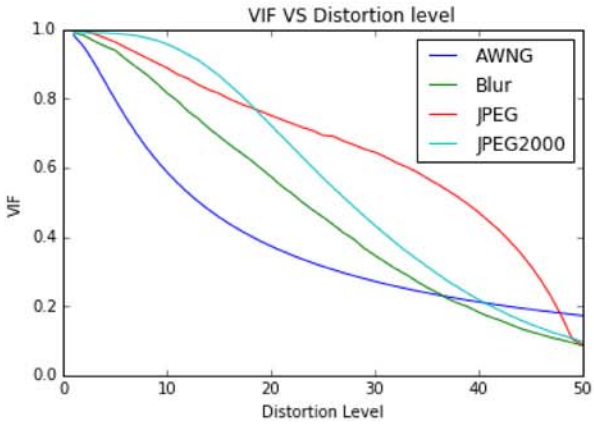


Figure 3.3. Averaged VIF scores vs distortion levels with 512 sized images

distortion level of 50. The plot is adjusted so that the lines of four distortion are decreasing in equal interval. Considering the QE used as a reference, VIF has its own flaws and this QE mapping is only applied as a reference to generate the distortion images. This may not influence the result because in total, 50 leveled images are enough to generate representable impaired images with different distortion levels. One group of sample images are shown below with certain distortion levels.

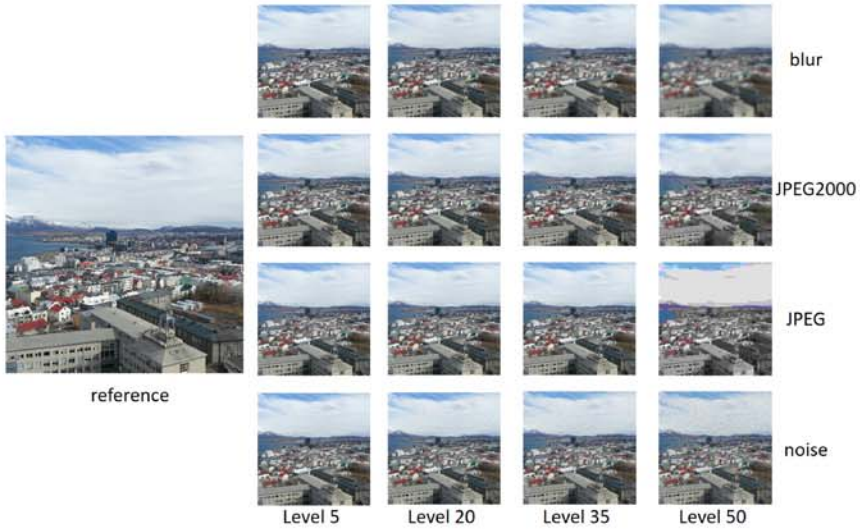


Figure 3.4. 512 sized image, with certain level of distortion levels

It can be seen from Figure 3.4 that the distortion increases as the level rises. In addition, some low level images do not have too much obvious distortion. This happens for a certain distortion type such as JPEG. Some distortion level ranges have clear separation between each other and others do not. For further analysis, the level ranges of clear distinguishable images are more studied.

3.2.3 Different Sized Images

All the distortion techniques above are applied for 512*512 sized images only. For larger sized images, distortion type noise and JPEG compression still use the same impairment technique, while the knob values of blur and JPEG 2000 compression increase with image sizes. This increment ratio is determined by using the sum of the width and height of input image sizes divided by the sum of those in 512 sized images (i.e. 512+512). Figure 3.5 shows one example reference image and its heaviest distorted images for four distortion types across 3 sizes. Three sized images with resolution 512*512, 1024*1024 and 2048*2048, are shown with same viewing angle and all the distorted images are level 50. Comparing these three sized images, it can be observed that there is little difference in blur distortion, but for other 3 typed distortion, larger sized images present better qualities. This means the distortion technique for blur is close to make equal viewing angle while the noise and JPEG are more close to equal pixel level. JPEG 2000 is designed to have same quality experience for different sized images but it is still possible to see the quality differences. This results from the mapping method of the knob values (bit rate) of JPEG 2000 distortion.

3.3 QE Testing Experiment

This section is about the QE testing experiment using STIQE software with the images generated in section 3.2. We will first talk about the experiment procedures and background information of testing QEs. Then we provide the results of three

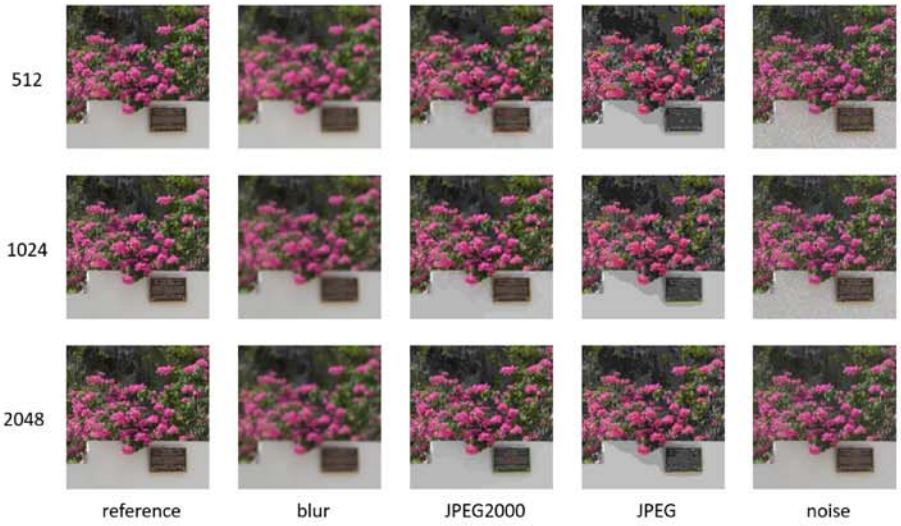


Figure 3.5. Example reference image and its most distorted (level 50) images

different testing aspects: separability test in section 3.3.2, invariance test in section 3.3.3 and monotonicity test in section 3.3.4 and 3.3.5. Part of the results shown in this section are also presented in the paper of STIQE software package [66], but with four extra FR QEs added in this work: GSM, RFSIM and SRSIM.

3.3.1 Preparation

The objective QE values analysis is based on 60 reference images and each reference image has 3 different sizes. Each sized reference image is impaired by 4 distortion types and each distortion type has 50 distortion levels. This ends up with 200 distortion images. In total, there are 12000 distorted images for each sized images prepared for objective QE test. 14 QEs are selected to be analyzed in this test, which are shown in Table 3.1, together with their run time for each sized images. The theoretical best and worst quality image score is also presented. Notice the symbol '*' represents that the best or worst value of that QE is not determined, and the value is the maximum or minimum value that observed in the test.

Table 3.1.
Summary of FR and NR QEs

QE	Type	Runtime (sec/image)			Best	Worst
		512	1024	2048		
ADM	FR	0.146	.062	2.7	1	0
FSIM	FR	0.34	0.54	1.44	1	0
GSM	FR	0.05	0.17	0.65	1	0.8921*
MAD	FR	1.52	6.3	29.2	0	184.6603*
PSNR	FR	0.036	0.108	0.4	Inf	0
PSNR-HVS-M	FR	2.17	8.9	35	Inf	0
RFSIM	FR	0.09	0.18	0.65	1	0
SRSIM	FR	0.05	0.16	0.64	1	0
SSIM	FR	0.046	0.14	0.66	1	0
BIQI	NR	0.68	1.06	1.58	0	100
BRISQUE	NR	0.24	0.49	1.5	0	100
CORNIA	NR	3.5	4.2	7.8	-14.8456*	113.5498*
IL-NIQE	NR	9.8	9.8	9.85	0	145.215*
NIQE	NR	0.3	1.1	5.02	0	22.9973*

The run time of each QE is tested on the machine and these values may vary based on other machines. It can be observed that some QEs such as MAD and PSNR-HVS-M are very time-consuming, especially for 2048 sized images.

3.3.2 Undistorted & Badly Distorted Images Test

For each QE in the table, the software computes the QE values for every reference image (undistorted images) and its level 50 image (badly distorted images) for in each distortion type. Then all QE values from 60 reference images generate CDFs and the CDFs are plotted on the same figure with 3 results from different sized images. One example plot with IL-NIQE is shown in Figure 3.6.

In Figure 3.6, the black lines show the CDF of the undistorted images which are on the left hand side of the plot. The thicker lines represent larger sized images. The line markers of Blue squares, red +, yellow 'o' and green triangle represent JPEG, blur, noise and JPEG-2000 respectively. The largest marker shows 2048 sized, middle sized is 1024 and smallest is 512. Comparing undistorted images and badly distorted CDFs, IL-NIQE has a good separation of 512 sized images but it has an overlapping

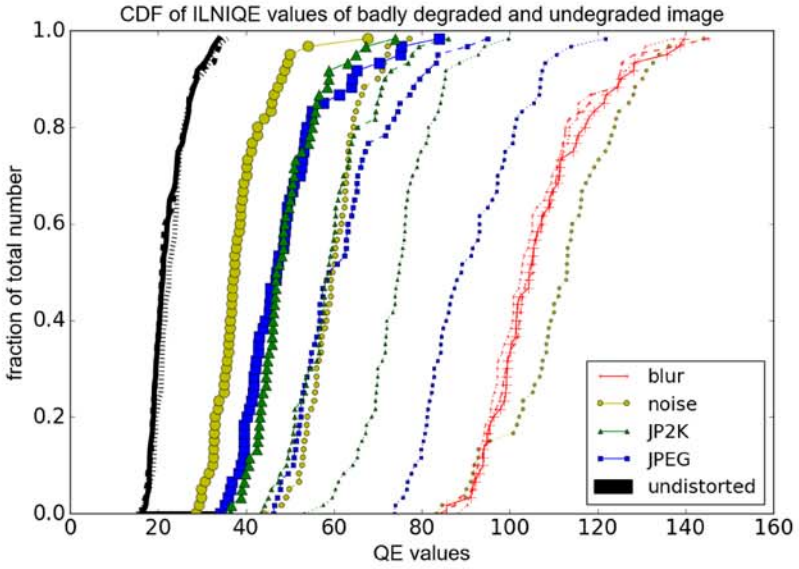


Figure 3.6. Undistorted & badly distorted image test with IL-NIQE

region for large sized images. Among four distortion types, it is interesting to see noise, JPEG and JPEG 2000 distortions have very clear separated with different sized images while all 3 sized blur images share almost the same CDF. This results from the distortion generation scheme, which provides different sized blur images with same visual quality at same viewing angles. JPEG and noise are designed to make the images have same quality at same pixel levels, which lead to a different quality levels for the three sizes. JPEG 2000 is designed to be same at same viewing angle like blur, but the size ratio does not perform well in this case and make the result not obvious.

For further exploration of this analysis, the software is designed to compute an overlapping rate which is defined in equation (3.1). In the equation, A represents the undistorted image scores, B is the badly distorted image scores. If the QE uses larger value to represent good quality images, A and B need to be switched. The overlap value is basically the overlapping range's percentage of the whole QE range. Positive overlap value means no overlap between high quality and low quality images. The absolute value of overlap value means the distance of two distributions. A well

behaved QE should have a large positive overlap value. While a smaller value shows a worse performance. The number of images in overlapping region is also counted.

Apart from the separation test, Kolmogorov-Smirnov (KS) statistical test [67] is applied to determine the similarities of two distributions. A larger P value generated from the KS test means that two distributions are more similar. If the QE has more similar CDF of blur images, it is possible to believe the QE performs on viewing angle level. If the QE has more similar CDF on noisy image, then the QE is more likely to be designed based on pixel level.

$$\text{overlap} = \frac{B_{min} - A_{max}}{B_{max} - A_{min}} \tag{3.1}$$

The results of overlap values and KS-test is shown in Table 3.2. It can be seen from the table that the overlap of high and low quality images (negative values) all happens in NR QEs since they do not have a reference for comparison. Among 5 NR QEs, BRISQUE has the best separation and BIQI has the worst. MAD has the largest separation among FR QEs. Based on the analysis of KS analysis, it can be concluded that among the FR QEs, ADM, FSIM, GSM, SRSIM and SSIM are all more effective at comparing different-sized images with identical viewing angle, while PSNR and PSNR-HVS-M are more effective for constant pixel size. Similarly, among the NR QEs, IL-NIQE is unique, in that it is more effective for identical viewing angle, while all other NR QEs are more effective for constant pixel size.

3.3.3 Invariance QE Test

Based on the invariance test function, the QE values of all cropped images are computed and the software finds the maximum differences in every 9 pair of images. This maximum QE difference value is called the 'resolution', which means any two images with QE difference smaller than this value are considered to have the same quality level. A good QE should have a small resolution, which means more accurate. Two distortion JPEG and JPEG 2000 are tested for each reference image, the 95 % of

Table 3.2.
QE statistics for high & low quality images.

QE name	Overlap range percentage	Percent of images in overlap region	KS-blur	KS-noise
ADM	+9.6	0.0	0.0	0.0
FSIM	+3.5	0.0	0.477	0.0
GSM	+2.2	0.0	0.629	0.0
MAD	+42.6	0.0	0.0	0.0
PSNR	–	0.0	0.345	0.911
PSNRHVSM	–	0.0	0.0	0.16
RFSIM	+24.7	0.0	0.239	0.0
SRSIM	+3.0	0.0	0.784	0.0
SSIM	+2.9	0.0	0.629	0.0
BIQI	-48.2	64.2	0.0	0.0
BRISQUE	-3.1	1.4	0.0	0.784
CORNIA	-10.0	3.6	0.0	0.0
ILNIQE	-5.5	4.1	0.629	0.0
NIQE	-5.9	4.0	0.0	0.477

maximum QE difference is shown in Table 3.3. The table also includes the theoretical maximum and minimum of that QE. If the theoretical value is unknown, the software finds the maximum and minimum among all possible QE values. In Table 3.3, it can be concluded that for FR QEs, ADM is one QE has a relatively high percentile of this resolution which reaches almost 20% of its total range. Other FR QEs have relatively small resolution, which are all smaller than 5%. About NR QEs, the resolution values are generally larger than FR, but BRISQUE and NIQE have percentages less than 5%. In addition, CORNIA shows negative values in the test, which maybe out of its original defined range.

3.3.4 Monotonicity QE Test

In this test, the QE values of every distortion level image are computed and among 50 leveled QE scores, the maximum non-monotonic ΔQE and Δd_level [27] are computed. Then the software counts the number of images with monotonicity behaviors and plot the ΔQE vs Δd_level for non-monotonic images. A well behaved QE should have all images with monotonic behaviors. The number of monotonicity

Table 3.3.
QE statistics for invariance test. (Observed best and worst values are in parentheses when the paper does not indicate best/worst.)

QE name	Best	Worst	JPEG 95%tile of ΔQE_{max}	JPEG-2000 95%tile of ΔQE_{max}
ADM	1	0	0.0280	0.1954
FSIM	1	0	0.0031	0.0054
GSM	1	0.8921*	0.0003	0.0003
MAD	0	184.6603*	5.4169	4.6657
PSNR	inf	0	0.2552	0.2521
PSNRHVSM	inf	0	2.7049	0.3124
RFSIM	1	0	0.0754	0.0273
SRSIM	1	0	0.0016	0.0018
SSIM	1	0	0.0121	0.0321
BIQI	0	100	23.567	44.404
BRISQUE	0	100	14.202	3.9741
CORNIA	-14.8456*	113.5498*	19.780	17.810
ILNIQE	0	145.215*	5.7695	9.0333
NIQE	0	22.9973*	0.6052	1.4668

images of 14 QEs are shown in Table 3.4. In this table, the maximum image number of every cell is 60. It can be observed that almost all FR QEs have very good monotonicity behavior for noise distortion. However some QEs such ADM and PSNR-HVS-M perform badly for blur distortion. ADM also does not have good performance for other two distortion types. NR QEs also have the best monotonicity behavior in noise distortion with 512 sized images. As size increases, BRISQUE and IL-NIQE become worse. Apart from noise, BRISQUE and NIQE also have some monotonic behavior for blur. All QEs does not have good performance on JPEG and JPEG 2000 distortions. CORNIA behaves poorly in this test, since all reference images show non-monotonic behavior in every case.

For further analysis of monotonicity of NR QEs, the software provides the plots of ΔQE vs Δd_level. The non-monotonicity plot is shown in Figure 3.7. Two lines in this figure represent the 80% separation of points vertically and horizontally. It can be observed that the non-monotonic images of blur lies horizontally with QE values while JPEG points falls more vertically with distortion levels. Blur images in this cases have larger QE value difference within small distortion levels, which is more

Table 3.4.
Statistics for monotonicity test. Number of reference images for which QE demonstrates fully monotonic behavior.

QE name	Noise			Blur			JPEG			JP2K		
Size	512	1024	2048	512	1024	2048	512	1024	2048	512	1024	2048
ADM	59	60	60	0	0	19	20	20	28	25	42	54
FSIM	60	60	60	60	60	51	60	58	55	59	59	60
GSM	60	60	60	60	47	13	59	58	54	60	60	60
MAD	60	60	60	58	51	46	50	47	45	40	51	53
PSNR	60	60	60	38	1	0	60	60	60	48	59	60
PSNR-HVS-M	60	60	60	3	2	0	51	53	46	39	59	60
RFSIM	60	60	60	60	59	59	58	59	54	54	60	59
SRSIM	60	60	60	60	60	45	57	54	56	56	60	60
SSIM	60	60	59	60	60	44	59	57	56	60	59	60
BIQI	19	19	20	0	0	0	4	2	1	0	0	0
BRISQUE	51	34	15	14	30	6	0	1	1	0	1	5
CORNIA	0	0	0	0	0	0	0	0	0	0	0	0
IL-NIQE	38	14	0	0	0	0	0	0	0	0	0	0
NIQE	36	37	36	2	24	16	0	0	0	0	0	0

likely to produce potential False Difference (FD). JPEG images, in contrast, have small difference QE values within a large range of distortion levels, which is more likely to produce potential False Ties (FT).

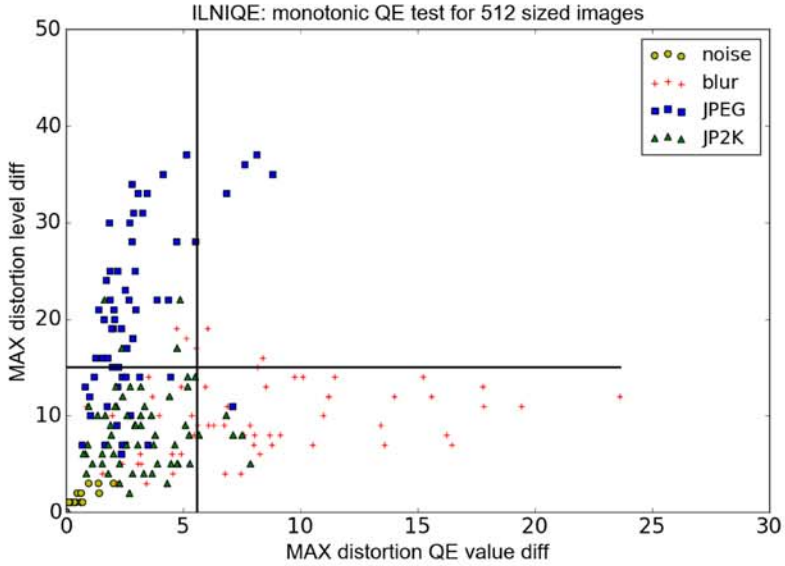


Figure 3.7. Non-Monotonicity results for IL-NIQE with 512-size images

3.3.5 QE Pairwise Comparison Test

In this test, each QE saved in the P file are compared with all other QEs, in order to view their preferences in pair of images. For example, given a pair of distorted images A and B, let QE_1 and QE_2 make judgments for this pair and see if both QEs agree that A is better than B. The more pairs two QEs agree to each other, they are more likely to have the same behavior. If one QE does not agree with most of other QEs with very large percentages, there may be some problems with that QE itself.

Four different kinds of pairing methods are used, same reference image and same distortion type ('11' case), same reference image but different distortion type ('10' case), different reference image but same distortion type ('01' case), different reference image and different distortion type ('00' case). These four cases are designed for specific analysis in the scope of distortion types or reference image contents. Each case generates different number of image pairs, and the result for all pairs of images can be computed based on the results of these four cases by weights. Table 3.5 shows the pairwise comparing percentage for all possible pairs of images. In this table, each value shows the disagreement percentage between that QE with all others. It can be observed that most FR QEs agree with each other more than NR QEs. FSIM, GSM and SRSIM have the least disagreement percentage, while BIQI and CORNIA have percentage over 20%. In FR QEs, BRISQUE and ILNIQE have good performances but IL-NIQE fails as size increases.

For more detailed analysis, the CDF plot of disagreement percentage based on different distortion type and different reference image (case 00) for 512 sized images is shown in Figure 3.8. Each line in the figure shows the CDF of disagreement percentage of one QE. If the line is on the left side of the plot, this means the QE that line represents agrees more to other QEs. The end points of all lines gathered with others since every two QE are compared and the disagreement contributes to the CDF line. It can be observed that all FR QEs (solid lines) are on the left side and NR

Table 3.5.
QE pairwise comparing disagreement percentages for all possible pairs

QE_name	512 size	1024 size	2048 size
ADM	15.88	11.92	14.86
FSIM	11.0	10.01	11.09
GSM	11.0	10.0	11.01
MAD	12.96	11.98	13.08
PSNR	16.91	17.98	20.01
PSNR-HVS-M	16.1	13.13	15.13
RFSIM	13.96	12.04	12.24
SRSIM	11.0	10.01	11.01
SSIM	12.08	11.08	12.08
BIQI	22.97	19.16	21.89
BRISQUE	17.07	15.06	19.08
CORNIA	20.01	17.19	18.37
IL-NIQE	16.06	18.22	23.18
NIQE	20.93	17.99	20.06

QEs (dashed lines) are right side. Among FR QEs, PSNR and PSNR-HVS-M have the most disagreement with others, while GSM, SSIM and SRSIM have the highest agreement rate. For NR type, IL-NIQE and BRISQUE have some overlapping with FR CDF lines. But BIQI and NIQE are more likely to be separated on the right size, which means they do not agree with the most of other QEs.

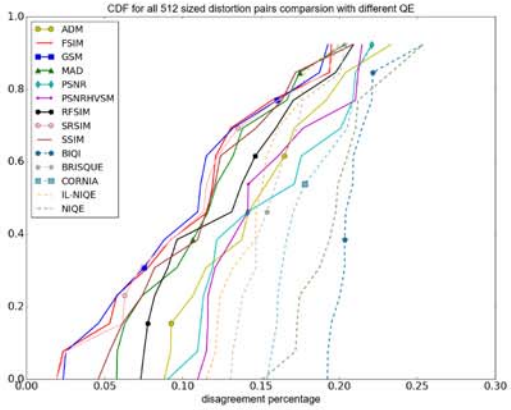


Figure 3.8. QE pairwise comparing disagreement percentage for 512 sized image with different reference image and different distortion type (00 case)

4. SUBJECTIVE IMAGE DATABASE

Using the software designed in chapter 2 and 3, an image database is generated in order to provide subjective data for QE analysis. This chapter mainly focuses on the designing and implementing subjective tests, together with the raw data processing. First in section 4.1, we will discuss the procedure of subjective test: database design, image preparation, subjective test and raw data analysis. Then the detailed information of the online test is explained in section 4.2. Then the in-lab test is also introduced in section 4.3. After that we are going to talk about the method and results of analyzing raw subjective data in section 4.4. Finally we make some discussions and suggestions about the subjective test in section 4.5.

4.1 Subjective Test Preparation

This section covers the preparation work of subjective test. The first section 4.1.1 discusses about the design of the database. Then the four main test phases are introduced in section 4.1.2.

4.1.1 Database Design

When designing a database, there are several aspects that researchers need to consider, such as image distortion types, distortion levels and the content of reference images. Many widely used databases are limited by the number of distortion levels and the resolution of reference images. For this new database, we focused on more distortion levels and high resolution reference images. This novel database is designed to have 60 reference images, 4 distortion types, and each distortion type is divided into 50 distortion levels. The four distortion types are Additive White Gaussian Noise

(AWGN), Gaussian Blur (GB), JPEG compression (JPEG) and JPEG2000 compression (JP2K). The reference images will have both 1024*1024 resolution version and 512*512 version. In total there are 24000 distorted images in the database.

Every distorted image in this database can be described by 4 parameters, Reference image number (R), Distortion type (D), distortion Level (L) and image Size (S).

$$Img_i = (r, d, l, s)$$

$$i \in \big[1, 2, 3, ...24000\big]$$

$$r \in \big[1, 2, 3, ...60\big]$$

$$d \in \{1, 2, 3, 4\} = \{AWNG, GB, JPEG, JP2K\}$$

$$l \in \big[1, 2, 3, ...50\big]$$

$$s \in \{1, 2\}$$

In the expression above, i represents the index of all distortion images. r is the index of reference image and d represents number 1, 2, 3 and 4, which represent four distortion types, AWGN, GB, JPG and JP2 respectively. l represents the distortion level, with level 50 indicating the most heavily distorted and 1 representing the least. Two image size 1024 * 1024 and 512 * 512 are represented in s with numbers 1 and 2 respectively. The image index i can be calculated by four parameters in equation (4.1):

$$i = (s-1) * 12000 + (r-1) * 60 + (d-1) * 50 + l - 1 \tag{4.1}$$

The test is processed in a paired comparison method, which requires human viewers to select a better quality image from a pair of images. Since paired comparison method needs to test each image with all other images, it is not possible to compare every two images in the database. And because the number of images is large, we first select representative images and then divide them into 4 testing groups.

4.1.2 Test Steps

The whole testing processes is designed with 4 phases, with each focusing on one different image parameter.

In phase 1, parameter D, R and S are fixed for each pair and only the L changes. This phase only uses 1024 by 1024 sized images. 11 out of 50 different distortion level images are selected per impair type per reference image. These 11 images have distortion levels of 1 (lowest distorted), 5, 10, 15, 20, 25, 30, 35, 40, 45 and 50 (heavily distorted). So there are 11*4*60 (2640) images in this testing image pool. Because the population of this image pool is still large, it is not possible to compare every two images in this pool. The images are paired if they share the same reference image and same distortion type. And among each set of 11 images, only images with similar distortion levels are tested, e.g. level 1 with level 5, level 5 with level 10, level 10 with level 15 and so on. This stage contains 10 pairs. Also images with level 1 and level 10, level 10 and level 20, till level 40 and level 50 are also included. This stage gives another 5 pairs. So there are 15 pairs from the same distortion type and the same reference image. Then in total there are 15*4*60 (3600) pairs of images that are tested for this phase.

In phase 2, only S and R are fixed, and the Distortion types (D) vary. All images tested for this phase are 1024 by 1024 sized. We select different distortion Levels for images in one pair so that they are comparable. For example, we can compare JPEG level 20 image with Blur level 40 image. The selection of images depend on the result from phase 1 and their objective QE scores. The number of selected images from each reference image can vary, but no more than 11, which is the number of selected levels from phase 1. The paired comparison procedure of this phase is the same as phase 1.

In phase 3, S, L and D are fixed, and Reference images (R) vary, so that viewers compare images with different content. Similar to previous phases, we only select images with resolution of 1024. The distortion level needs to be relatively large,

so that the distortion can be clearly observed by participants. Since there are 60 reference images in total and each reference image includes 4 distortion types, the number of these candidate images is quite large. As a result, the image pairs are constructed, taking into account of the subjective data from the previous phases, to obtain the most information from the viewers' inputs. The paired comparison procedure of this phase is the same as phase 1.

In phase 4, D and R are fixed, and the image size (S) varies. Because there are only two image sizes in the database, 512 size and 1024 size images are compared in every pair. The images' content and distortion type are the same in every pair. The distortion levels are largely different because for the same distortion level, participants normally prefer a high resolution image. To make two different sized image comparable, lightly degraded 512 image are selected with a heavily degraded 1024 image. The comparison method is the same as in phase 1 and the actual number of pairs are determined later.

After the test, not all of the images are tested because the 1024 sized images are the focus of the test. 512 sized images are supporting data and build a connection from 1024 sized to 512 sized images. This also helps to give a scope of comparing this new database to currently widely used databases. The overall test plan is ambitions, therefore in this work, only phase 1 of the test is implemented because of time issues.

4.2 Online Subjective Test

This part mainly discusses about the works of the online subjective test. We first talk about the platform in section 4.2.1. Then the testing interface is introduced in section 4.2.2. Finally we summarize the procedures of this online subjective test in section 4.2.3.

4.2.1 Online Platform

The preparation of subjective test begins when the images in the database are generated. The test is performed on Amazon Mechanical Turk (AMT) platform, so it is required to generate required testing interfaces based on its requirement. Every test generated on the platform is formed of Human Intelligence Test (HIT) and each HIT can be done by different participants. Researchers or requesters give the participants financial rewards based on the number of HITs they finished. Requesters have the right to reject any results which do not meet the qualification and refuse to pay.

4.2.2 Test Interface

The testing format is based on static web pages and each page corresponds to one HIT. All these web pages are stored in research lab server, together with all distorted images. For example in phase 1, 3600 pairs are tested and they are separated into 360 groups randomly and one group has 10 pairs of images. Each group is converted into one static web page as one HIT. As a result, 360 web pages need be generated for phase one. To generate these web pages, a template is designed in an HTML file and the program is used to substitute the image links in every HTML file.

Considering the platform we used in the test, 1024 sized image are presented by online participants through their own displaying devices. Due to the fact that two high resolution images cannot be fully displayed on some monitors, one image is display at a time. The participants cannot view two 1024 sized images at the same time, however, they are allowed to view a pair of images, by switching back and forth before they make judgments. There is a one second delay when an image is displayed. This delay prevents users from switching two images quickly and make unreasonable choices. Still, a minimum screen resolution of 1440*900 is required participants to join the test. It is suggested that both images in a pair should be view around five seconds. Therefore, participants have around 15 to 20 seconds to make judgment for each pair. An example testing interface is shown in Figure 4.1.

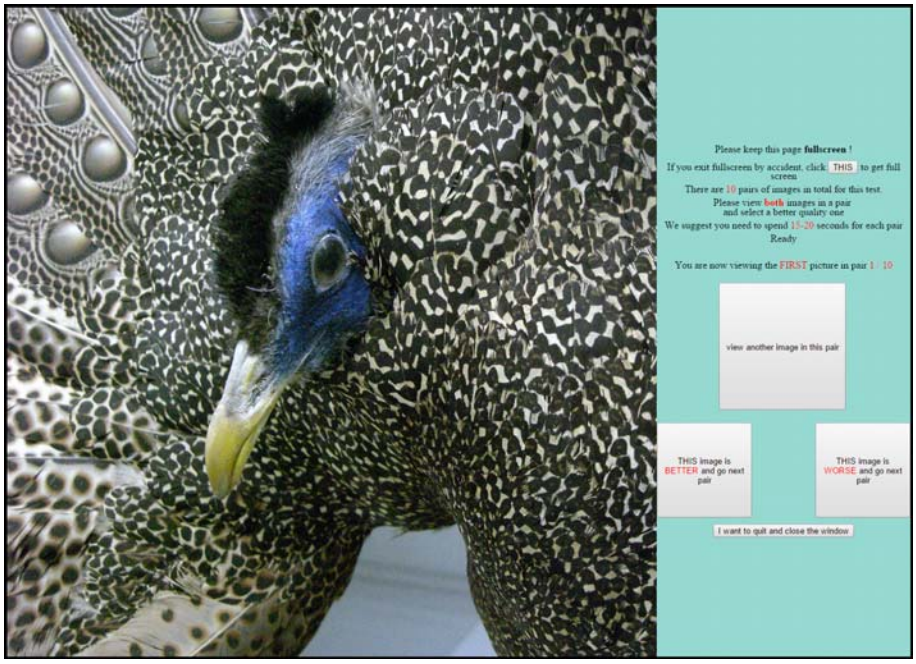


Figure 4.1. Example testing interface on the web page

It can be seen that one image is shown at a time on the left side, and there are buttons on the right side. The main button in the center is viewing another image in the pair, which can be used to switch two images, back and forth. Human viewers can show preferences by clicking the button: this current image is better or worse than the other. This testing interface is displayed in full-screen. There is a limit on the number of participants to view two images in a pair; they can not switch images more than 8 times. If this happens, images disappears and participants are forced to choose a better quality image.

Every HIT is designed to include 10 pairs of images for participants to compare since people can easily get tired after 10 minutes. For ten pairs, every HIT only requires 3 to 4 minutes. In addition, each HIT is viewed by five different human subjects and this number does not include the rejected results. The financial reward of every HIT is set to be $0.04.

4.2.3 Test Procedures

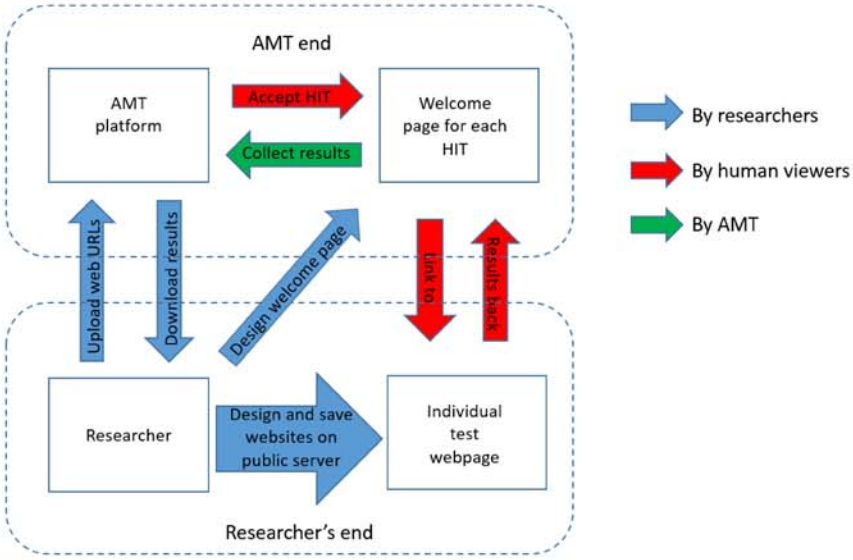


Figure 4.2. The flow chart of the cloud based subjective test

The all testing procedure is shown in Figure 4.2. In this Figure, apart from designing every testing pages, researchers also need to provide web page links to AMT and generate the welcome page. The welcome pages are designed by researchers but stored and displayed on AMT. For participants, they accept HITs first on AMT and get to the welcome page. From there they are given the link to the subjective test web pages on lab server. After they finish, they are required to copy the 'raw results' back to welcome page on AMT as a proof that they finished the test. Researches can download the collected 'raw results' from AMT and validate the results as acceptable or not. If five valid results are received from one HIT, that HIT will be closed, but if some results are rejected, the procedure repeats. This procedure ends when every HIT has five valid results.

One example welcome page is shown in Figure 4.3. Every welcome page includes one HIT and provide the link to the testing page. The welcome page also records the 'raw result' from the test page. Since this process requires human viewer to copy

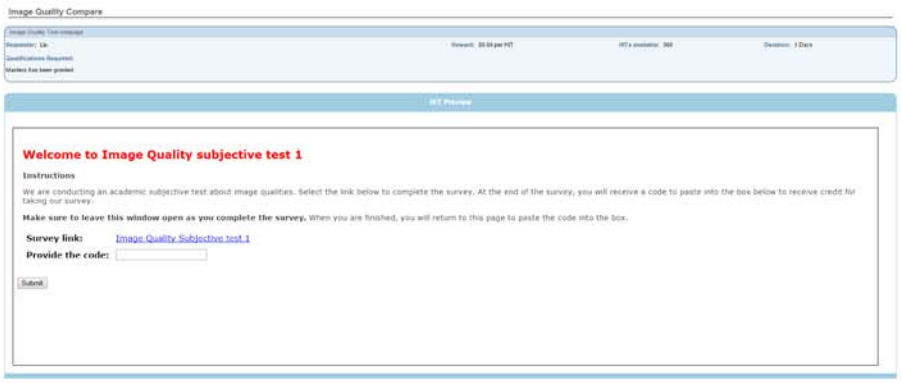


Figure 4.3. Welcome page on AMT

'raw results', it is important to encrypt the results. The method of encryption used in this test is mapping the numbers to random letters. The code book is unique and saved on the server. Instead of the viewers preferences, the 'raw result' also includes viewers' monitor resolution, color depth, actually web page resolution, and the time they spent on the test. This information is used to judge whether the results are accepted.

After the raw data is filtered, a program is developed to map each choice to a result matrix, which records the sum of votes of all the pairs which are pairwise compared. For 1024*1024 sized images, the result matrix is 12000*12000. Based on the results from phase 1, there are almost 3600*2 cells filled with values in this matrix.

4.3 In-Lab Subjective Test

This section covers the in-lab subjective test. The design of the test is described in section 4.3.1 and the testing condition is introduced in section 4.3.2.

4.3.1 Purpose and Design

The In-Lab subjective test is a complementary test and it is performed after the online test. The purpose is to gather more data in a shorter time period in a different testing environment.

The testing images are same as the images used in online subjective test, but only first 15 reference images are selected for the test, which provides 900 pairs per view in total. Each pair of images requires 10 seconds at most and after 10 seconds the images disappear and the participant should make a choice. The waiting time between two pairs is set to be 1 seconds. On average, each pair costs around 5 to 8 seconds. Based on the concerns of time for each test, 900 pairs are divided into 5 lists of pairs, with each list consists of 180 pairs, which costs around 15 - 20 minutes. Each pair of images is set to be compared by 5 people so this test requires 25 people/HIT in total.

4.3.2 Test Interface

The In-Lab test is based on a two screen testing environment which is shown in Figure 4.4. The testing software used for this test is PsychoPy [68], which is able to present two images on two screens. For each time, two 1024*1024 sized images are displayed simultaneously and the participants are allowed to directly compare the differences of two images. Two images are positioned in the middle part of the screens and viewing distance is set to be around 1 meter. The user is asked to press key '1' and '0' from the keyboard to select the left or right image.

Based on this testing environment, the display difference of the two screens may influence the participant's choices. The screen is calibrated using Spyder5 [69] before all the test started. Comparing to online test, the confidence level of in-lab tests are higher [70].

Figure 4.4. Testing environment of In-Lab subjective test

4.4 Data Analysis

This section talks about the process of the raw results generated from the test. Both online and in-lab tests generate data matrices separately and these matrices include all the preference of human subjects between pair of images. A merged matrix is generated by summing the online and in-lab matrices together. We will first talk about the theory of the computation in section 4.4.1. Then we will discuss about the necessary conditions of this computation method in section 4.4.2. After that, some example subjective scores are presented in section 4.4.3. In addition, we compare the generated subjective scores to some QE scores in section 4.4.4.

4.4.1 Computation Theory

When the paired comparison matrix are available, models are applied to compute the scores for each image. Bradley Terry model is a linear model [71] used in paired comparisons. This model is described in equation (4.2).

$$P\left(i > j\right) = \frac{\pi_i}{\pi_i + \pi_j} \tag{4.2}$$

In equation (4.2), the left side shows the probability of object i beats j. On the right side, π_i represents the true ratings of the ith score. If there is a list of objects numbered from 1 to m, and we want to to compute the parameters π. Then the Maximum Likelihood Estimation can be applied with the objective function shown in equation (4.3).

$$L\left(\overrightarrow{\pi}\right) = \sum_{i,j}^{m} w\left(i,j\right) \log\left(\frac{\pi_i}{\pi_i + \pi_j}\right) \tag{4.3}$$

Equation (4.3) shows the log-likelihood function, and $w\left(i,j\right)$ represents the number of times that object i beats j.

Theoretically, the image quality scores can be computed follow by this model. In phase 1, only images with same reference image (R) and same distortion type (D) are compared. So all 11 images (level 1, 5 ...50) with same R and D generates one block for an image quality score computation process. The detailed computation algorithm is below.

1. Grab one block (11*11) out of the whole result matrix.
2. Generate an initial guess $\overrightarrow{q}$ vector
3. Optimize the objective function (4.3) and get optimized quality scores $\overrightarrow{q_{opt}}$
4. Compute the error $|\overrightarrow{q} - \overrightarrow{q}_{opt}|$ and check if the norm is larger than threshold value (set by 0.1).

5. If the error is larger than threshold, use $\overrightarrow{q_{opt}}$ as initial guess and go back to step 3.

6. If the error is smaller than threshold, the result is $log\,(\overrightarrow{q_{opt}})$ and the algorithm terminates.

However, when applying the Bradley Terry model with MLE method, there is a constrain: every partition of the objects into two nonempty sets, must have some objects in the second subset that has been preferred to at least once to some objects in subset one [32]. This condition is also explained in [72] using the concept of graphs: suppose all objects in a list are the nodes of a graph and the $w\,(i,j)$ represents a directed edge from node i to j. Then in this graph, that constrain can be expressed that there is always a path from any node to other nodes in the graph. The matrix block is calculable only if it satisfies this requirement.

4.4.2 Matrix Block Analysis

The result matrix generated from the subjective tests is 12000 by 12000. Inside this matrix, only cells representing images with same reference images and same distortion type have non-zero entrees. So before doing the computation, the first step is to separate and analyze the block of matrix. The block of matrix example is shown in Figure 4.5.

In Figure 4.5, two axises are the distortion levels that shows the distortion levels of images that are compared in phase 1. The cell of row i and column j stores the number of how many people prefer the image with distortion level i than level j. This value is denoted as $w\,(i,j)$ in equation (4.3). Only cells in green and yellow store the preference votes and all other values are 0. To check if this block can be computed by Bradley Terry model and MLE, the block must satisfy the requirement mentioned in section 4.4.1. From Figure 4.5, it can be observed that one possible way for the matrix to fit the algorithm is that all green cells have non zero entrees. Another way to satisfy the requirement is to have one green cell 0 but its adjacent

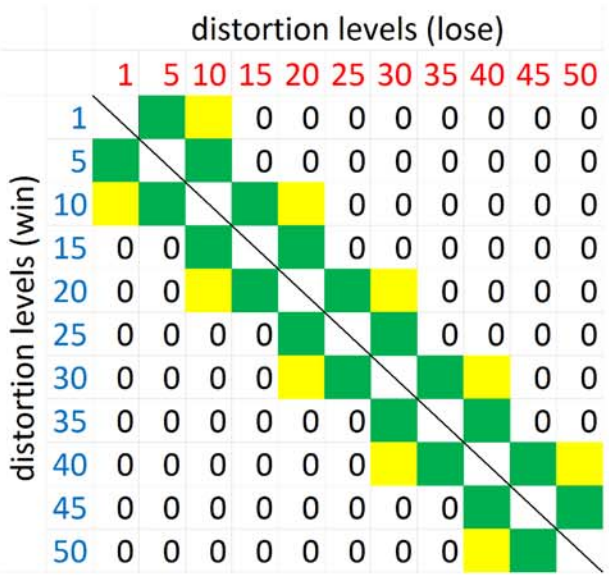


Figure 4.5. One block of matrix example

yellow matrix should be non-zero. All matrix blocks are checked based on these two features. The number and percentage of blocks from different distortion types that satisfy the requirements are shown in Table 4.1.

Table 4.1.
The number and percentage of satisfied blocks

	first 15 images		last 45 images		all images	
distortion	number	%	number	%	number	%
noise	13	86%	19	45%	32	53%
blur	6	40%	14	31%	20	33%
JPEG	10	66%	23	51%	33	55%
JP2K	12	80%	23	51&	35	58%

In Table 4.1, it can be observed that the first 15 reference images have higher percentage of satisfied blocks because the first 15 images are tested in both online and in-lab subjective test and have more viewers. On average, every pair of images

from the 15 are viewed by 11 people while the pair of last 45 reference images are only viewed by 5 people. It can be concluded that a block matrix with more data is more likely to be computed. However, comparing the four distortion types, blur distortion has the least number of satisfied blocks. This indicates that participants can clearly distinguish blur images with different blur levels, so that all people prefer one image in a pair than another.

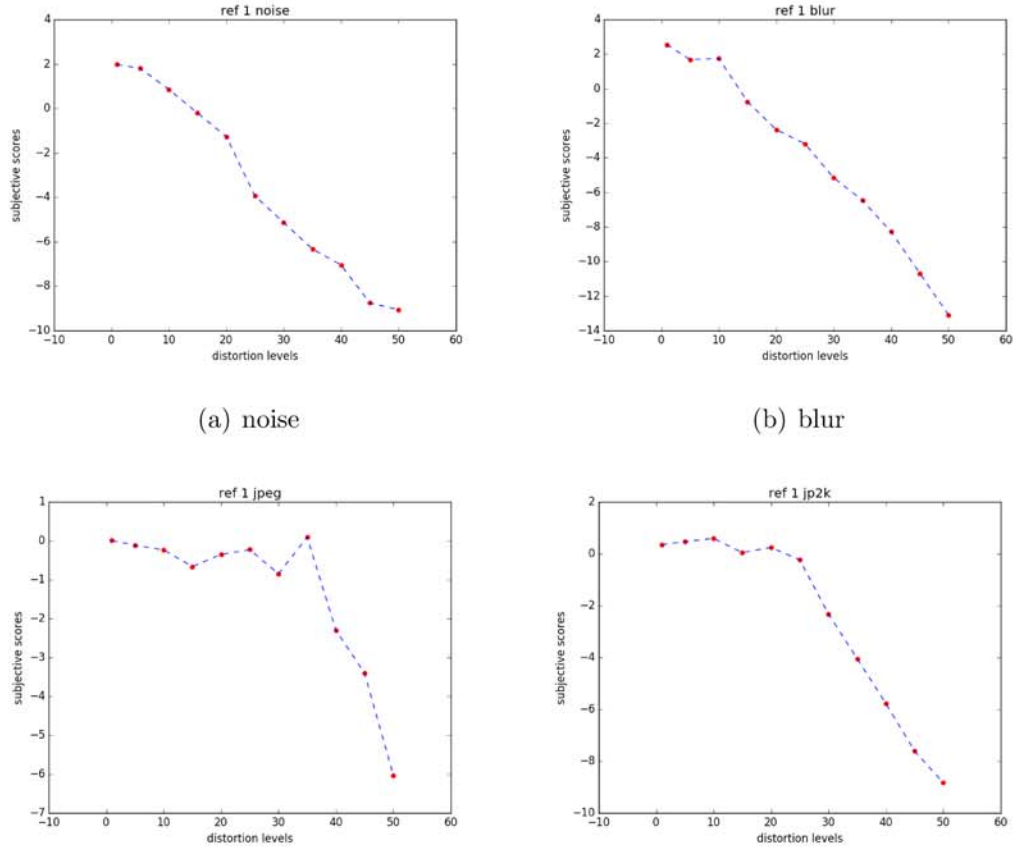
4.4.3 Ranking Data Result

The blocks that are satisfied with the Bradley Terry model requirement can be computed. Each block will generate individual subjective quality scores for every image. The algorithm described in section 4.4.1 is applied for this computation. The subjective data of reference image 1 is plotted in Figure 4.6. The block matrices of four distortion types from reference image 1 all satisfy the computation requirement.

In Figure 4.6, the X axis shows the distortion level and Y axis represents the subjective scores. For ideal case, the subjective scores should have a linear relationship of the distortion levels. From these figures, the noise and blur distortion have plots that are basic linearly decreasing. While for JPEG and JPEG 2000 distortions, their lines have a flat start and then dramatically decrease after level 30. Especially for JPEG distortion, the points before level 30 fluctuates and cause non-monotonicity. This is because the images in low level ranges of the JPEG and JPEG 2000 are too close for the participants to tell the difference. The scores of every distortion type are relative rankings, which only represents the difference between each other. Since no different reference images or different distortion images are compared, the scores are only meaningful in their own graphs.

4.4.4 Subjective Scores vs QE Values

The goal of generating subjective database is to benchmark the objective QE performances. In this section, we use the available subjective scores to test some



(a) noise

(b) blur

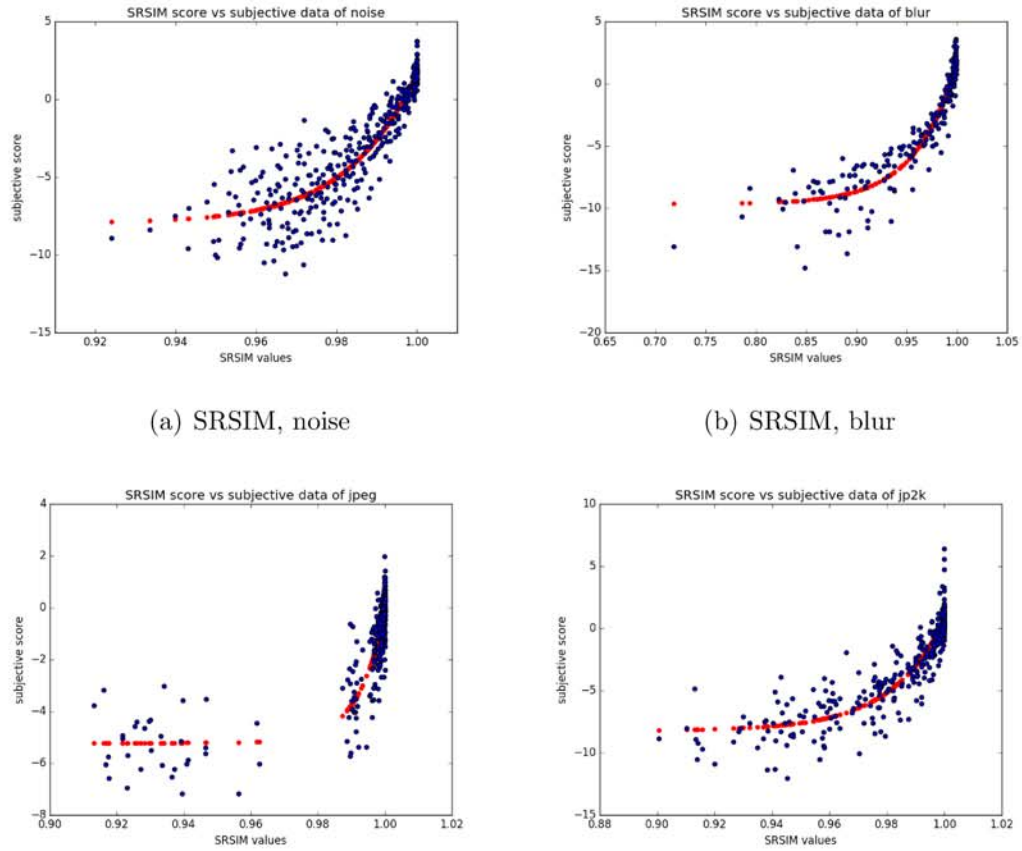
(c) JPEG

(d) JP2K

Figure 4.6. Plots of subjective scores of reference image 1

QEs. All the subjective scores are chosen from the result matrix blocks which meet the requirement of Bradley Terry and MLE method. Still, we present the QE scores versus subjective scores of two QEs: SRSIM and CORNIA in Figures 4.7 4.8.

In Figures 4.7 and 4.8, the red dots represent the non-linear mapping of two group of data using equation (2.4). It can be observed that all four distortions generally present monotonicity relations between QE values and subjective scores. Comparing these subplots, JPEG distortion has a very dense distribution in high subjective score region (low distortion level images). In this dense region, CORNIA has relative large vertical range, which means this QE cannot tell the difference of these images. Apart



(a) SRSIM, noise

(b) SRSIM, blur

(c) SRSIM, JPEG

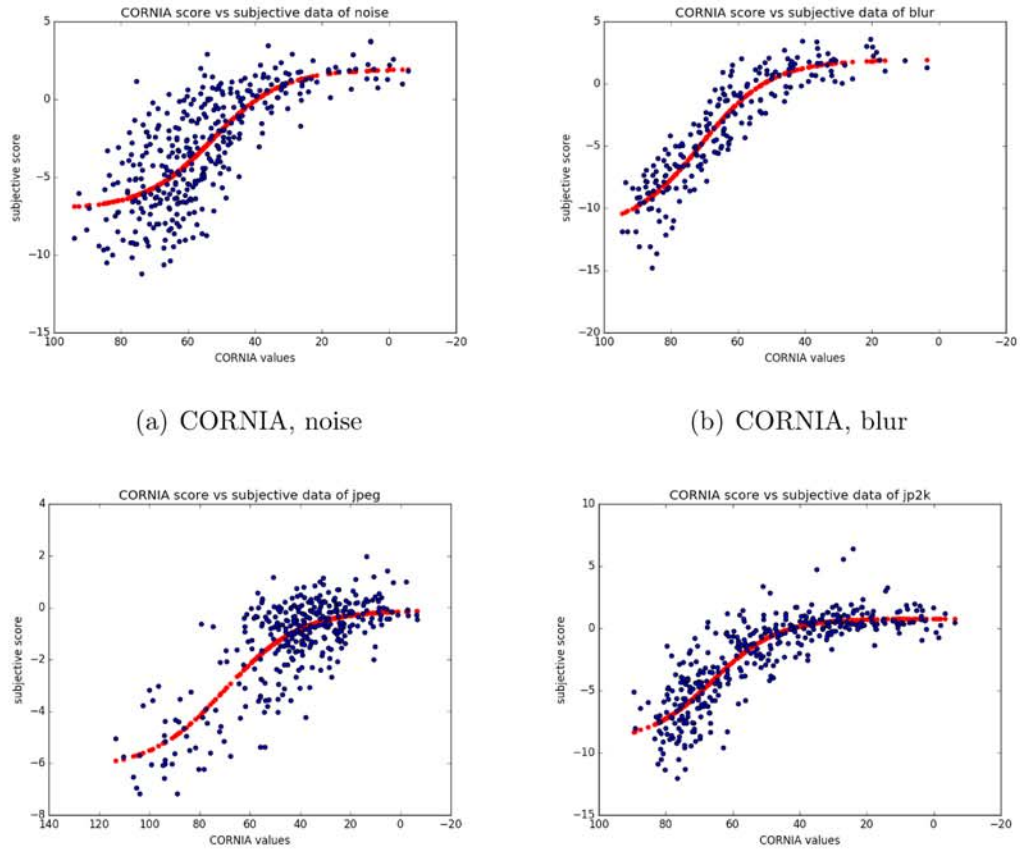
(d) SRSIM, JP2K

Figure 4.7. SRSIM values vs subjective scores

from JPEG distortion, this dense region also happens in the low subjective score region (high distortion level images) in noise distortion.

Apart from the plots of QE values versus subjective scores, corrections can be computed to check how close these two distribution are. The Pearson correction is computed for QE performance check. The correlations are shown in Table 4.2.

In Table 4.2, JPEG distortion has relatively the lowest correction among all four distortion types. This is because the low distortion leveled JPEG images are not that subjectively distinguishable. Noise distortion has a relatively high correction, which may result from the fact that noise has the largest the number of calculable matrix shown in Table 4.1. Comparing all FR QEs, ADM and MAD have relatively higher



(a) CORNIA, noise

(b) CORNIA, blur

(c) CORNIA, JPEG

(d) CORNIA, JP2K

Figure 4.8. CORNIA values vs subjective scores

correlation of all four distortions. While PSRN has the lowest correlation. BRISQUE has the highest correlation in NR QEs.

4.5 Discussion

In this discussion section, we will talk about some further analysis and provide some suggestions for further subjective tests. Firstly, the features of matrix blocks which cannot result in ranking scores are further explored in section 4.5.1. Then we will cover the problems of image distortion levels in section 4.5.2. Then we give some comments and suggestions for the design of online subjective test in section 4.5.3.

Table 4.2.
Pearson Correction of QEs with subjective data

QE	noise	blur	JPEG	JP2K
ADM	0.9	0.955	0.809	0.88
FSIM	0.826	0.889	0.755	0.865
GSM	0.849	0.789	0.77	0.829
MAD	0.888	0.919	0.867	0.921
MIQE	0.861	0.939	0.856	0.896
MSSSIM	0.859	0.865	0.794	0.85
PSNR	0.833	0.744	0.615	0.723
PSNRHVSM	0.837	0.84	0.711	0.811
RFSIM	0.879	0.898	0.86	0.901
SRSIM	0.841	0.863	0.746	0.851
SSIM	0.763	0.866	0.754	0.828
VIF	0.851	0.9	0.848	0.837
VSI	0.849	0.789	0.77	0.829
BIQI	0.83	0.81	0.757	0.698
BRISQUE	0.826	0.926	0.818	0.781
CORNIA	0.727	0.869	0.747	0.791
ILNIQE	0.837	0.879	0.78	0.825
NIQE	0.849	0.939	0.759	0.774

4.5.1 Incalculable Matrix Block

The matrix block checking process is important before computing the subjective data. The unsatisfied matrices are not able to use the Bradley Terry model with MLE. To avoid this problem, Tsukida and Gupta discussed about this problem [30]. Instead of directly process the input matrix, they proposed a related solution by changing 0 entrees into 0.5 and their opposite entrees to $m - 0.5$, where m is the number of total number of people who view this pair.

Apart from this, we also find some blocks satisfying the requirements still cannot converge in this computation process. Two paired comparison matrix blocks are found with this problem and one of them is shown in Figure 4.9.

The matrix in Figure 4.9 satisfies the computation requirements but the optimization process does not converge. Checking the pair of images with level 15 and level 10, 4 people choose level 15 is better than level 10 among 6 viewers. This pair causes a problem because it disagree with other adjacent pairs and the optimization function terminates at the maximum number of iteration. This situation also happens on

distortion levels (win) \ distortion levels (lose)	1	5	10	15	20	25	30	35	40	45	50
1		5	4	0	0	0	0	0	0	0	0
5	1		5	0	0	0	0	0	0	0	0
10	2	1		2	5	0	0	0	0	0	0
15	0	0	4		4	0	0	0	0	0	0
20	0	0	1	2		4	5	0	0	0	0
25	0	0	0	0	2		4	0	0	0	0
30	0	0	0	0	1	2		5	5	0	0
35	0	0	0	0	0	0	1		4	0	0
40	0	0	0	0	0	0	1	2		4	6
45	0	0	0	0	0	0	0	0	2		4
50	0	0	0	0	0	0	0	0	0	2	

Figure 4.9. Paired comparison matrix of outlier points in blur distortion

some blocks in JPEG distortion. But unlike blur blocks, in the low distorted level regions of JPEG, more than one pair of comparison do not agree with each other, which generate a 'messy' matrix. In this case, the optimization process of the 'messy' matrix converges. Back to the non-converged matrix in blur distortion, all pairs show the same result that lower level image is better except that only one exception. So it can be concluded that in this 'diagonal only' setup of paired comparison test, one opposite pair against all other pairs may lead to non-converge problem.

To avoid generating incalculable matrix blocks systematically, one direct way is to gather more viewers for the test. This steps can eliminate the 0 entrees in some degree. Also testing more off-diagonals positions in the matrix helps to provide more connections between images.

The most important step is to design test images with better distortion levels. By comparing images with clear difference is a direct reason of generating 0 entrees. In this test, because of there are 50 distortion levels, only representative leveled images are selected for comparison. These images works for noise, JPEG and JPEG 2000 distortions because these images are not easy for participants to select a better one.

But for blur distortion, the participants are able to find the difference by searching detailed areas on both images for every pair we tested. As a result, when testing blur distortion, images with closer distortion level should be paired together.

4.5.2 Image Distortion Levels

The image distortion levels are important and hard to control in generating the image databases. The ideal distortion levels should almost agree with result from human viewing experiences. This ideal case presents the linear relationship of subjective scores versus distortion levels. Besides, the worst levels of different distortion types should provide the same viewing experience for human subjects. However, because different human viewers may have different opinions, this goal is hard to achieve.

In this test, no subjective viewers are involved in the image generation processes, because 50 levels are designed and informal subjective test is expensive. Objectively, the distortion levels are determined with respect to the one widely used QE VIF [61]. The distortion parameters are mapped to levels with carefully computed non-linear equations to make image with same distortion levels have the same VIF scores. Because VIF algorithm mainly compares the amount of information difference between reference image and distorted image, the distortion levels generated using VIF causes problems for JPEG and JPEG 2000 images. These two compression methods take advantage of HVS and reduce the information that human eyes are not sensitive to. This causes the fact that low leveled JPEG and JPEG 2000 images provides very high viewing experience for human subjects, and when the level reaches the threshold of human eyes, the viewing experience dramatically decreases. One example is shown 4.10.

As Figure 4.10 shows, before level 35, the subjective scores fluctuates around a high level, but decreases immediately after level beyond 35. This problem may be solved by non-linear mapping. The mapping of distortion parameters can reduce the range of low distortion levels and generate more levels in dramatic decreasing regions.

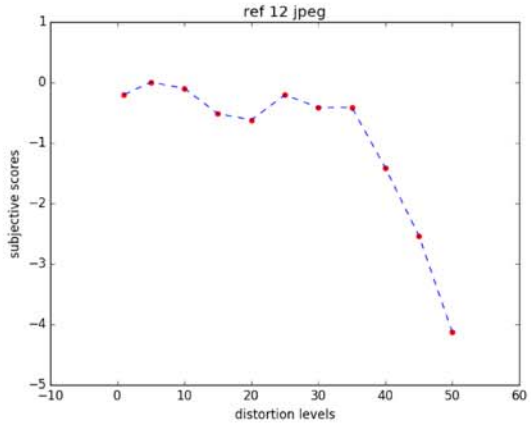


Figure 4.10. Subjective scores versus distortion levels for Reference image 12 with JPEG distortion

Instead, purely applying objective QE as reference is problematic, informal subjective test is useful when designing different distortion level images.

4.5.3 Online Subjective Test Design

In this test, images with resolution of 1024*1024 are tested through online testing platform. This resolution problem is solved by displaying one image each time. The participants should switch images back and forth to view both images before making a judgment. This testing method may not be that effective when displaying two images at the same time.

The online subjective test is the main data sources in this database. The data reliability may be a problem and causes error when analyzing the data. In this test, several testing details are collected from the subjects to prove the data reliability, such as the seconds they spent on the test and their screen resolution. More detailed information can be collected such as the time that people views every pair and the number of times the images are switched back and forth. Longer time they spent on each pair and more times they switching images back and forth, indicates the image pairs have closer viewing experiences, and the preference is less reliable.

The testing time is another problem of online subjective test. In this test, 1800 HIT are posted on AMT and each HIT costs 3 minutes on average. The reward for each accepted HIT is $0.04. With this testing condition, the process lasts for 1 month which is beyond our expected testing period. This is mainly based on the financial reward, but by increasing the testing amount for each HIT may help to decrease the whole testing period. In addition, the online testing web page should be able to reject the result automatically if it is unaccepted. This process will help to increase the raw data processing efficiency.

5. SUMMARY

This thesis mainly focuses on the analysis of current image Quality Estimators (QE) and building new subjective data base. A number of programs are implemented and saved in several functional modules. Some of the modules are combined to a published software STIQE. The subjective database is generated based on the data collected from both the online and in-lab subjective tests.

The QE analysis software includes five main modules, image impair, subjective database access, QE calculation, objective QE score mapping and statistical analysis. This software can load subjective databases or generate distortion images as test cases to evaluate the performance of QEs. The published software STIQE is a purely objective analysis software, which can use any images as input to test QEs. STIQE mainly focuses on three testing aspects, the good bad quality separability, pixel shift invariance and monotonicity of different distortion levels. In software testing experiment, 60 reference images with 4 distortion types are used as input to test 14 QEs.

The subjective database is designed based on the 60 reference images used in the software testing part. These images are processed to 1024 * 1024 sized and each reference image is distorted with four distortion types. For each distortion type, the one reference image is processed to generate 50 distortion leveled images. Pair comparison method is applied for the subjective test and the whole test is divided into four phases, but only phase 1 is covered in this thesis. The subjective test is firstly performed through the online platform and followed by another in-lab test. The result shows the blur and noise distortion have the most linear ranking scores while JPEG distortion and JPEG 2000 images do not have a clear separation in low distortion region of their subjective scores.

REFERENCES

REFERENCES

[1] S. S. Hemami and A. R. Reibman, "No-reference image and video quality estimation: Applications and human-motivated design," *Signal processing: Image communication*, vol. 25, no. 7, pp. 469–481, 2010.

[2] 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. IEEE, 2002, pp. IV–3313.

[3] Z. Wang and A. C. Bovik, "Modern image quality assessment," *Synthesis Lectures on Image, Video, and Multimedia Processing*, vol. 2, no. 1, pp. 1–156, 2006.

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

[5] H. R. Sheikh, A. C. Bovik, and G. De Veciana, "An information fidelity criterion for image quality assessment using natural scene statistics," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2117–2128, 2005.

[6] D. M. Chandler and S. S. Hemami, "VSNR: A wavelet-based visual signal-to-noise ratio for natural images," *IEEE Transactions on Image Processing*, vol. 16, no. 9, pp. 2284–2298, 2007.

[7] A. M. Demirtas, A. R. Reibman, and H. Jafarkhani, "Multiscale image quality estimation," 2013.

[8] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: a feature similarity index for image quality assessment," *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2378–2386, 2011.

[9] P. Gupta, P. Srivastava, S. Bharadwaj, and V. Bhateja, "A HVS based perceptual quality estimation measure for color images," *ACEEE International Journal on Signal & Image Processing (IJSIP)*, vol. 3, no. 1, pp. 63–68, 2012.

[10] Z. A. Seghir and F. Hachouf, "Full-reference image quality assessment measure based on color distortion," in *Computer Science and Its Applications*. Springer, 2015, pp. 66–77.

[11] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *Conference Record of the Thirty-Seventh Asilomar Conference on Signals, Systems and Computers, 2004*, vol. 2. Ieee, 2003, pp. 1398–1402.

[12] M. Hassan and C. Bhagvati, "Structural similarity measure for color images," *International Journal of Computer Applications*, vol. 43, no. 14, pp. 7–12, 2012.

[13] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on Image Processing*, vol. 21, no. 12, pp. 4695–4708, 2012.

[14] A. M. et al., "Making a "completely blind" image quality analyzer," *IEEE Signal Processing Letters*, vol. 20, no. 3, pp. 209–212, 2013.

[15] P. Ye, J. Kumar, L. Kang, and D. Doermann, "Real-time no-reference image quality assessment based on filter learning," in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 987–994.

[16] P. Ye and D. Doermann, "No-reference image quality assessment based on visual codebook," in *2011 18th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2011, pp. 3089–3092.

[17] Y. Zhang and D. M. Chandler, "An algorithm for no-reference image quality assessment based on log-derivative statistics of natural scenes," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2013, pp. 86 530J–86 530J.

[18] W. Xue and X. Mou, "Reduced reference image quality assessment based on weibull statistics," in *2010 Second International Workshop on Quality of Multimedia Experience (QoMEX)*. IEEE, 2010, pp. 1–6.

[19] Z. Wang and E. P. Simoncelli, "Reduced-reference image quality assessment using a wavelet-domain natural image statistic model," in *Electronic Imaging 2005*. International Society for Optics and Photonics, 2005, pp. 149–159.

[20] A. A. Abdelouahad, M. E. Hassouni, H. Cherifi, and D. Aboutajdine, "A reduced reference image quality measure using bessel k forms model for tetrolet coefficients," *arXiv preprint arXiv:1112.4135*, 2011.

[21] A. Rehman and Z. Wang, "Reduced-reference image quality assessment by structural similarity estimation," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3378–3389, 2012.

[22] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on image processing*, vol. 15, no. 11, pp. 3440–3451, 2006.

[23] Y. Han, Y. Cai, Y. Cao, and X. Xu, "Monotonic regression: A new way for correlating subjective and objective ratings in image quality research," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 2309–2313, 2012.

[24] A. R. Reibman, "A strategy to jointly test image quality estimators subjectively," in *2012 19th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2012, pp. 1501–1504.

[25] I. T. Union, "Method for specifying accuracy and cross-calibration of video quality metrics(VQM)," 2004.

[26] W. Xue, L. Zhang, and X. Mou, "Learning without human scores for blind image quality assessment," in *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2013, pp. 995–1002.

[27] F. M. Ciaramello and A. R. Reibman, "Systematic stress testing of image quality estimators," in *2011 18th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2011, pp. 3101–3104.

[28] C. Keimel, J. Habigt, C. Horch, and K. Diepold, "Qualitycrowd—a framework for crowd-based quality evaluation," in *Picture Coding Symposium (PCS), 2012*. IEEE, 2012, pp. 245–248.

[29] F. Ribeiro, D. Florencio, and V. Nascimento, "Crowdsourcing subjective image quality evaluation," in *18th IEEE International Conference on Image Processing (ICIP), 2011*. IEEE, 2011, pp. 3097–3100.

[30] K. Tsukida and M. R. Gupta, "How to analyze paired comparison data," DTIC Document, Tech. Rep., 2011.

[31] J. Redi, H. Liu, H. Alers, R. Zunino, and I. Heynderickx, "Comparing subjective image quality measurement methods for the creation of public databases," in *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics, 2010, pp. 752 903–752 903.

[32] J. C. Handley, "Comparative analysis of Bradley-Terry and thurstone-mosteller paired comparison models for image quality assessment," in *PICS*, vol. 1, 2001, pp. 108–112.

[33] L. H.R.Sheikh, Z.Wang and A.C.Bovik, "Live image quality assessment database release 2." [Online]. Available: http://live.ece.utexas.edu/research/quality

[34] C. T. Vu, T. D. Phan, P. S. Banga, and D. M. Chandler, "On the quality assessment of enhanced images: A database, analysis, and strategies for augmenting existing methods," in *2012 IEEE Southwest Symposium on Image Analysis and Interpretation (SSIAI)*. IEEE, 2012, pp. 181–184.

[35] N. Ponomarenko, V. Lukin, A. Zelensky, K. Egiazarian, M. Carli, and F. Battisti, "TID2008-a database for evaluation of full-reference visual quality assessment metrics," *Advances of Modern Radioelectronics*, vol. 10, no. 4, pp. 30–45, 2009.

[36] T. Hobfeld, R. Schatz, and S. Egger, "SOS: The MOS is not enough!" in *2011 Third International Workshop on Quality of Multimedia Experience (QoMEX)*. IEEE, 2011, pp. 131–136.

[37] A. C. Bovik, "Automatic prediction of perceptual image and video quality," *Proceedings of the IEEE*, vol. 101, no. 9, pp. 2008–2024, 2013.

[38] A. Torralba and A. Oliva, "Statistics of natural image categories," *Network: computation in neural systems*, vol. 14, no. 3, pp. 391–412, 2003.

[39] C. J. van den Branden Lambrecht, "A working spatio-temporal model of the human visual system for image restoration and quality assessment applications," in *Acoustics, Speech, and Signal Processing, 1996. ICASSP-96. Conference Proceedings., 1996 IEEE International Conference on*, vol. 4. IEEE, 1996, pp. 2291–2294.

[40] M. Carandini, J. B. Demb, V. Mante, D. J. Tolhurst, Y. Dan, B. A. Olshausen, J. L. Gallant, and N. C. Rust, "Do we know what the early visual system does?" *The Journal of neuroscience*, vol. 25, no. 46, pp. 10 577–10 597, 2005.

[41] A. V. Murthy and L. J. Karam, "IVQUEST – Image and Video QUality Evaluation SofTware," https://ivulab.asu.edu/software/quality/ivquest.

[42] "Python Imaging Library (PIL) 1.1.7 for python 2.7 (windows only)," http://www.pythonware.com/products/pil/#pil117.

[43] "opencv version 2.4.12 for windows," http://opencv.org/.

[44] "Kakadu Software version 7.8 for win32," http://kakadusoftware.com/.

[45] L. Yuan, J. Sun, L. Quan, and H.-Y. Shum, "Image deblurring with blurred/noisy image pairs," in *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3. ACM, 2007, p. 1.

[46] J. Shen, Q. Li, and G. Erlebacher, "Hybrid no-reference natural image quality assessment of noisy, blurry, jpeg2000, and jpeg images," *IEEE Transactions on Image Processing*, vol. 20, no. 8, pp. 2089–2098, 2011.

[47] E. C. Larson and D. M. Chandler, "Most apparent distortion: full-reference image quality assessment and the role of strategy," *Journal of Electronic Imaging*, vol. 19, no. 1, pp. 011 006–011 006, 2010.

[48] N. Ponomarenko, L. Jin, O. Ieremeiev, V. Lukin, K. Egiazarian, J. Astola, B. Vozel, K. Chehdi, M. Carli, F. Battisti *et al.*, "Image database TID2013: Peculiarities, results and perspectives," *Signal Processing: Image Communication*, vol. 30, pp. 57–77, 2015.

[49] S. L. et al., "Image quality assessment by separately evaluating detail losses and additive impairments," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 935–949, 2011.

[50] A. K. Moorthy and A. C. Bovik, "A two-step framework for constructing blind image quality indices," *IEEE Signal Processing Letters*, 2010.

[51] P. Ye et al., "Unsupervised Feature Learning Framework for No-reference Image Quality Assessment," in *Intl. Conf. on Computer Vision and Pattern Recognition (CVPR 2012)*, 2012, pp. 1098–1105.

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

[53] A. Liu, W. Lin, and M. Narwaria, "Image quality assessment based on gradient similarity," *IEEE Transactions on Image Processing*, vol. 21, no. 4, pp. 1500–1512, 2012.

[54] L. Z. et al., "A feature-enriched completely blind image quality evaluator," *IEEE Trans. on Image Processing*, vol. 24, no. 8, pp. 2579–2591, 2015.

[55] Z. Wang and Q. Li, "Information content weighting for perceptual image quality assessment," *IEEE Transactions on Image Processing*, vol. 20, no. 5, pp. 1185–1198, 2011.

[56] A. M. Demirtas, A. R. Reibman, and H. Jafarkhani, "Full-reference quality estimation for images with different spatial resolutions," *IEEE Transactions on Image Processing*, vol. 23, no. 5, pp. 2069–2080, 2014.

[57] N. P. et al. "On between-coefficient contrast masking of DCT basis functions," in *VPQM*, 2007.

[58] L. Zhang, L. Zhang, and X. Mou, "RFSIM: A feature based image quality assessment metric using riesz transforms," in *2010 17th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2010, pp. 321–324.

[59] L. Zhang and H. Li, "SR-SIM: a fast and high performance IQA index based on spectral residual," in *2012 19th IEEE International Conference on Image Processing (ICIP)*. IEEE, 2012, pp. 1473–1476.

[60] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE signal processing letters*, vol. 9, no. 3, pp. 81–84, 2002.

[61] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 430–444, 2006.

[62] L. Zhang, Y. Shen, and H. Li, "Vsi: A visual saliency-induced index for perceptual image quality assessment," *IEEE Transactions on Image Processing*, vol. 23, no. 10, pp. 4270–4281, 2014.

[63] K. Seshadrinathan, T. N. Pappas, R. J. Safranek, J. Chen, Z. Wang, H. R. Sheikh, and A. C. Bovik, "Image quality assessment," *The essential guide to image processing*, 2009.

[64] "scipy version 0.17.1," https://www.scipy.org/.

[65] M. H. Brill, J. Lubin, P. Costa, S. Wolf, and J. Pearson, "Accuracy and cross-calibration of video quality metrics: new methods from ATIS/T1A1," *Signal Processing: Image Communication*, vol. 19, no. 2, pp. 101–107, 2004.

[66] H. Liu and A. R. Reibman, "Software to stress test image quality estimators," 2016.

[67] H. W. Lilliefors, "On the Kolmogorov-Smirnov test for normality with mean and variance unknown," *Journal of the American Statistical Association*, vol. 62, no. 318, pp. 399–402, 1967.

[68] "PsychoPy software, version 1.8 in Python," http://www.psychopy.org/.

[69] "Spyder5 ELITE," http://spyder.datacolor.com/portfolio-view/spyder5elite/.

[70] S. Bosse, M. Siekmann, J. Rasch, T. Wiegand, and W. Samek, "Quality assessment of image patches distorted by image compression using crowdsourcing," in *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME)*, 2016.

[71] R. A. Bradley and M. E. Terry, "Rank analysis of incomplete block designs: I. the method of paired comparisons," *Biometrika*, vol. 39, no. 3/4, pp. 324–345, 1952.

[72] D. R. Hunter, "Mm algorithms for generalized Bradley-Terry models," *Annals of Statistics*, pp. 384–406, 2004.

APPENDIX

A. APPENDIX FIGURES

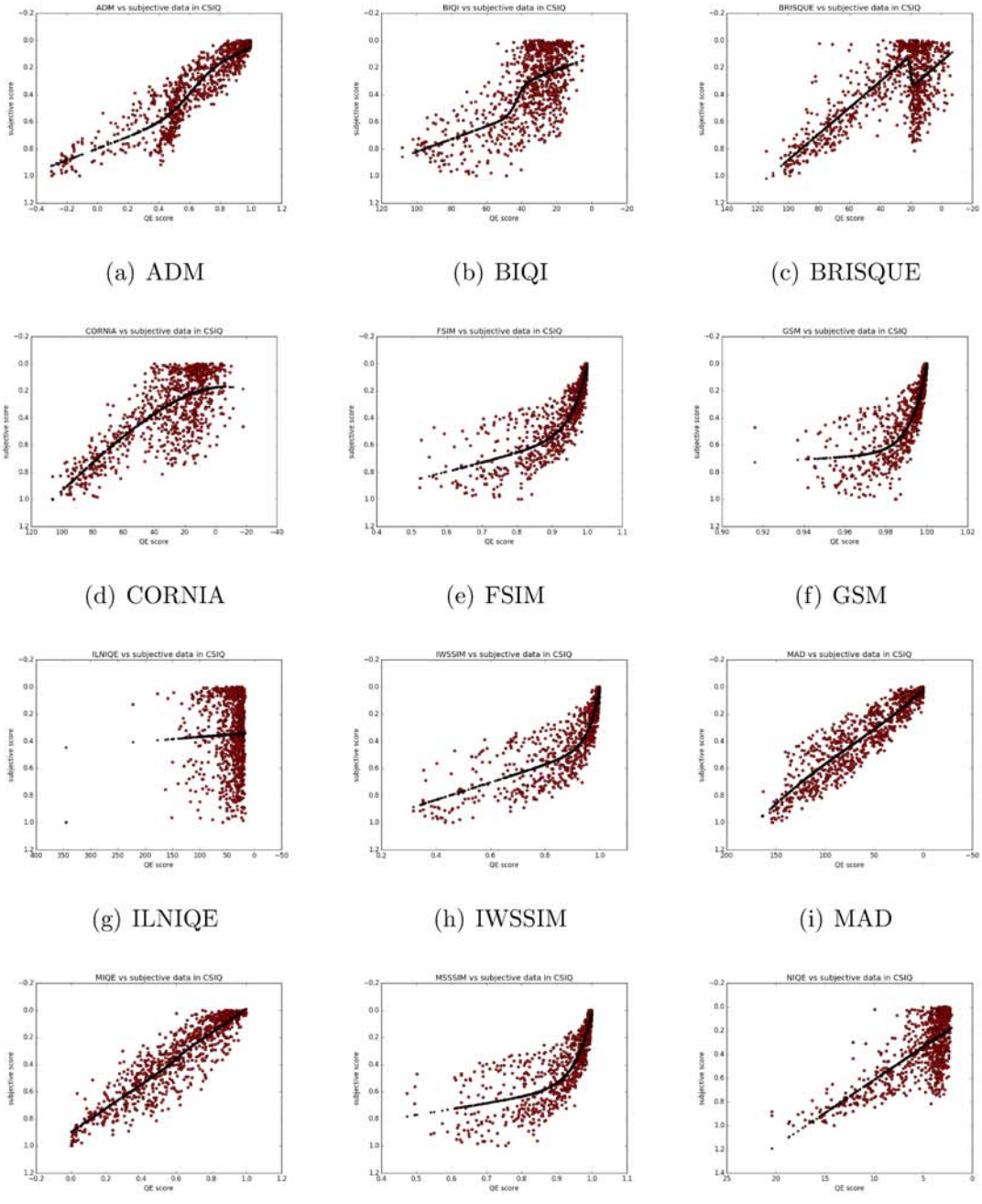
A.1 Reference Images

The Thumbnails of all 60 reference images used in the database is shown A.1.

A.2 Subjective Database & QE mapping

The software is able to load information from subjective databases, such as LIVE, CSIQ and TID2013. The data results will be saved in specific structure and available for further operations. The current QE metrics are also capable to run by this software, with any input images. If the QE is written in MATLAB code, the software will be able to call MATLAB software for QE computing. With subjective database, the software can plot the objective QE score vs subjective data, with non-linear fitting function. Figures below show all the objective QE scores vs subjective Mean Opinion Scores (MOS) from CSIQ database, together with the non-linear fitting functions.

Figure A.1. Reference images in the database



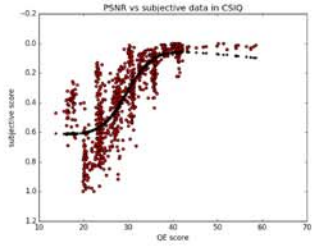
(a) ADM (b) BIQI (c) BRISQUE

(d) CORNIA (e) FSIM (f) GSM

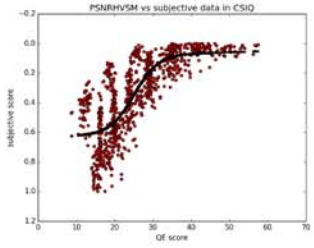
(g) ILNIQE (h) IWSSIM (i) MAD

(j) MIQE (k) MSSSIM (l) NIQE

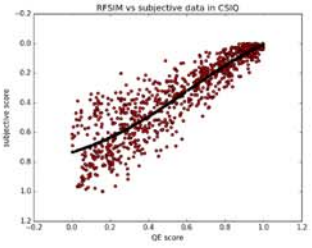
Figure A.2. The non-linear mapping plots of QE scores and CSIQ subjective data (1)



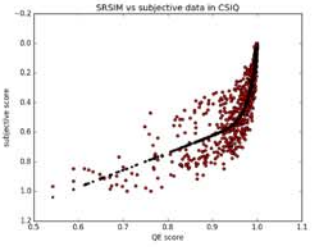
(a) PSNR



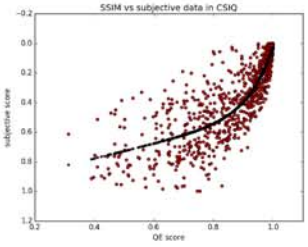
(b) PSNRHVSM



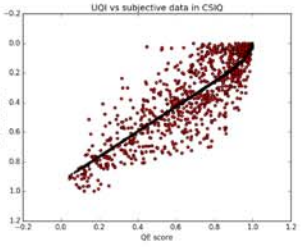
(c) RFSIM



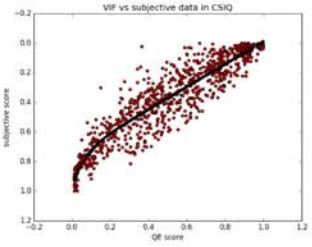
(d) SRSIM



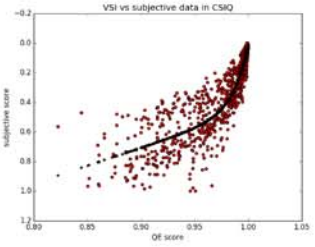
(e) SSIM



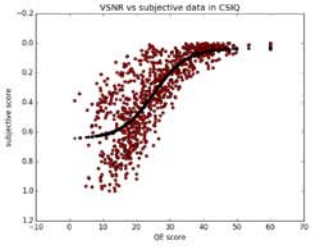
(f) UQI



(g) VIF



(h) VSI



(i) VSNR

Figure A.3. The non-linear mapping plots of QE scores and SCIQ subjective data (2)