

**TRACE-NORM REGULARISATION-BASED LEARNING
FRAMEWORK FOR BLIND IMAGE QUALITY ASSESSMENT
MODEL**



UNIVERSITI TEKNIKAL MALAYSIA MELAKA

**TRACE-NORM REGULARISATION-BASED LEARNING
FRAMEWORK FOR BLIND IMAGE QUALITY ASSESSMENT
MODEL**

RACHEL LYN JAHIRIN

**This report is submitted in partial fulfilment of the requirements
for the degree of Bachelor of Electronic/Computer Engineering
with Honours**

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

**Faculty of Electronic and Computer Technology and
Engineering**

Universiti Teknikal Malaysia Melaka

2024

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : Trace-norm Regularisation-based learning
Framework for Blind Image Quality Assessment
Model
Sesi Pengajian : 2023/2024

Saya RACHEL LYN JAHIRIN mengaku membenarkan laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (✓):

SULIT*

(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)

TERHAD*

(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan.)

TIDAK TERHAD

Disahkan oleh:

rachel

Dr. Redzuan Bin Abdul Manaf

(TANDATANGAN PENULIS)

(COP DAN TANDATANGAN PENYELIA)

Alamat Tetap: Kampung Bunga
Raya, Keningau
Sabah

DR. REDZUAN BIN ABDUL MANAF
Senior Lecturer
Faculty Of Electronic & Computer Engineering
Universiti Teknikal Malaysia Melaka (UTeM)
Hang Tuah Jaya
76100 Durian Tunggal, Melaka

Tarikh : 24 JANUARI 2024

Tarikh : 22 JANUARI 2024

DECLARATION

I declare that this report entitled “Trace-Norm Regularisation-Based Learning Framework for Blind Image Quality Assessment Model” is the result of my own work except for quotes as cited in the references.

 Signature : 
Author : RACHEL LYN JAHIRIN
Date : 24 JANUARI 2024


APPROVAL

I hereby declare that I have read this thesis and in my opinion this thesis is sufficient in terms of scope and quality for the award of Bachelor of Electronic Engineering with Honours.

Signature :  

Supervisor Name : REDZUAN BIN ABDUL MANAP

Date : 22 JANUARI 2024

DEDICATION

I dedicated this report to my supervisor Dr Redzuan bin Abdul Manap, my beloved family and all my friends.



ABSTRACT

The project's focus is on developing an image quality assessment (IQA) model that can accurately estimate an image's quality without the need of a reference image. The current blind IQA (BIQA) model typically trains their prediction separately for different image distortions, without considering the relationship between these learning tasks. Consequently, a BIQA model may perform well when tested on images affected by one type of distortion, but it may not be as effective when tested on other distortions. This project aims to overcome this limitation by simultaneously training a new BIQA model under different distortion conditions using the trace-norm regularisation-based learning framework. The model first extracts spatial domain BIQA features from a set of training images, and these features are then used as inputs to the trace-norm regularisation-based learning framework to learn prediction models for different distortion classes. The model then combines the predicted quality scores from each distortion present in the image to yield the overall image quality score.

ABSTRAK

Fokus projek ini adalah untuk membangunkan model penilaian kualiti imej (IQA) yang boleh menganggarkan kualiti imej dengan tepat tanpa memerlukan imej rujukan. Model IQA buta (BIQA) biasanya mempelajari ramalan mereka secara berasingan untuk herotan imej yang berbeza tanpa mempertimbangkan hubungan antara tugas pembelajaran. Ini menyebabkan model BIQA mungkin berprestasi baik apabila diuji pada imej yang dipengaruhi oleh satu jenis herotan, tetapi ia mungkin tidak begitu berkesan apabila diuji pada imej yang mengandungi herotan lain. Projek ini bertujuan untuk mengatasi had ini dengan melatih model BIQA baharu secara serentak di bawah keadaan herotan yang berbeza menggunakan rangka kerja berasaskan regularisasi jejak-norma. Pertama, model mengekstrak ciri domain spatial BIQA daripada set latihan imej, dan ciri-ciri ini kemudiannya digunakan sebagai input kepada rangka kerja pembelajaran berasaskan regularisasi jejak-norma untuk mempelajari model ramalan untuk kelas herotan yang berbeza. Skor kualiti yang diramalkan daripada setiap kelas herotan kemudian diberi pemberat iaitu anggaran keberangkalian setiap herotan yang terdapat dalam imej untuk menghasilkan skor kualiti yang terdapat dalam imej untuk menghasilkan skor kualiti keseluruhan imej tersebut.

ACKNOWLEDGEMENTS

I am profoundly grateful to my supervisor, Dr. Redzuan Bin Abdul Manap, whose unwavering support, patience, and invaluable guidance shaped every aspect of this project. His expertise and insightful feedback were pivotal in navigating the complexities of this project. I extend my sincere thanks to Universiti Teknikal Malaysia Melaka (UTeM) for providing the essential resources, access to facilities, and opportunities that facilitated the execution of this thesis. To my family, whose endless encouragement, understanding, and unwavering belief in my abilities sustained me throughout this journey-I owe you immeasurable gratitude. Your support has been my pillar of strength. Lastly, but by no means least, I express my heartfelt appreciation to all my friends for their continuous understanding, motivation, and patience during this demanding period. Each of these individuals and entities has played an indispensable role in the realization of this project. I am profoundly grateful to their contributions, without which this endeavor would not have been possible.

TABLE OF CONTENTS

Declaration	
Approval	
Dedication	
Abstract	i
Abstrak	ii
Acknowledgements	iii
List of Figures	vii
List of Tables	ix
List of Symbols and Abbreviations	x
List of Appendices	xii
chapter 1 INTRODUCTION	1
1.1 Introduction	1
1.2 Problem Statement	3
1.3 Objective	3
1.4 Scope of Work	4
CHAPTER 2 BACKGROUND STUDY	5
2.1 Introduction	5
2.2 Natural Scene Statistic (NSS) Based Model	6

2.2.1	Transform-Based Approach Model	v 6
2.2.2	Transform-Free Approach Model	7
2.3	Learning Based Models	8
2.3.1	General Learning Based Approach	8
2.3.2	Codebook Based Approach	11
CHAPTER 3 METHODOLOGY		13
3.1	Introduction	13
3.2	Flow Chart of Methodology	14
3.3	General Framework for The Proposed Model	16
3.4	Database Creation	16
3.5	Feature Extraction	17
3.5.1	Gradient of Magnitude (GM) and Laplacian of Gaussian (LOG)	18
3.5.2	Joint Adaptive Normalization	19
3.5.3	Statistical Feature Description	21
3.6	Quality Estimation	27
3.7	Training and Testing Procedure	30
3.8	Distortion Identification	30
3.9	Performance Analysis	31
3.9.1	Root Mean-Squared Error (RMSE)	31
3.9.2	Pearson Linear Correlation Coefficient (PLCC)	31

3.9.3 Spearman's Rank Order Correlation Coefficient (SROCC)	vi 32
CHAPTER 4 RESULTS AND DISCUSSION	33
4.1 Result Analysis	34
4.1.1 Database Construction	34
4.1.2 MATLAB Operation	36
4.1.3 Feature Extraction	37
4.1.4 Model Parameter	38
4.1.5 Train-test Partition	39
4.1.6 Performance Evaluation	40
4.1.7 Computational Complexity	47
4.2 Sustainable Design	48
CHAPTER 5 CONCLUSION AND FUTURE WORKS	50
5.1 Conclusion	50
5.2 Recommendation	51
REFERENCES	52
APPENDICES	56

LIST OF FIGURES

Figure 3.1: Flowchart of the project	15
Figure 3.2: The framework of the proposed model	16
Figure 3.3: Project database generated from LIVE database	17
Figure 3.4: The GM and LOG maps together with their marginal distribution before (middle column) and after (right column) joint adaptive normalization. (a) Houses; (b) Hats; and (c) Chessboard	21
Figure 3.5: Marginal probability function (P_G and P_L) of the distorted images produced at different DMOS values for one reference image	23
Figure 3.6: The independency distribution (Q_G and Q_L) represents the distribution of the independence among distorted images generated at various DMOS values concerning a single reference image	26
Figure 3.7: Training framework of trace-norm regularization	28
Figure 4.1: Database with 982 images	34
Figure 4.2: Database without reference image	35
Figure 4.3: Code to call image into MATLAB workspace	36
Figure 4.4: Example of image displayed in MATLAB	36
Figure 4.5: Coding for feature extraction	37
Figure 4.6: New database containing feature extraction data in column 5	37

Figure 4.7: Example of GMLOG feature data	38
Figure 4.8: Non-overlap random train-test partition based on the total number of references image contained in LIVE database	40
Figure 4.9: Median values SROCC and LCC across 1000 runs of the overall experiment.	41
Figure 4.10: Median values RMSE across 1000 runs of the overall experiment.	41
Figure 4.11: Median values for SROCC for different type of distortion on LIVE database	43
Figure 4.12: Median values for LCC across 1000 runs for different types of distortion.	45
Figure 4.13: Median values for RMSE across 1000 runs for different types of distortion	46
Figure 4.14: Sustainable development goals	49



LIST OF TABLES

Table 4.1: Median values across 1000 runs of the overall experiment	42
Table 4.2: Median for SROCC for different type of distortion on LIVE database	44
Table 4.3: Median LCC for different type of distortion in LIVE database	45
Table 4.4: Median value RMSE for different type of distortion in LIVE database	47
Table 4.5: Average Run-Time for different available BIQA	47

اونيورسيتي تيكنيكل مليسيا ملاك

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

LIST OF SYMBOLS AND ABBREVIATIONS

BIQA	:	Blind Image Quality Assessment
BIQI	:	Blind Image Quality Index
BRISQUE	:	Blind/Reference-less Image Spatial Quality Evaluator
BLINDS-11	:	Blind Image Integrity Notator utilizing DCT Statistic
CSIQ	:	Challenge Set for Image
CNN	:	Convolutional Neural Network
CGFA-CNN	:	Creative Gradient Feature Aggregation CNN
CORNIA	:	Codebook Representation for No-Reference Image Assessment
DMOS	:	Difference Mean Opinion Score
DS	:	Distortion Specific
DNN	:	Deep Neural Network
DI	:	Distortion Identification
FR-IQA	:	Full-Reference Image Quality Assessment
FF	:	Fast Fading
GB	:	Gaussian Blur
GM	:	Gradient Magnitude
IQA	:	Image Quality Assessment
JPEG	:	Joint Photographic Experts Group
JP2K	:	Joint Photographic 2000

LOG	:	Laplacian of Gaussian
LCC	:	Linear Correlation Coefficient
MOE	:	Mean Opinion Score
MSE	:	Mean Square Error
MEON	:	Multi-task End-to-End Optimized Network
MTL-IQ	:	Multi-task Learning Image Quality
NR-IQA	:	No-Reference Image Quality Assessment
NSS	:	Natural Scene Statistic
PSNR	:	Peak Signal-to-Noise Ration
PR-IQA	:	Pseudo-Reference Image Quality Assessment
PLCC	:	Pearson Linear Correlation Coefficient
QoE	:	Quality of Experience
QE	:	Quality Estimation
QS	:	Quality Score
ROI	:	Region of Interest
RMSE	:	Root Mean-Squared Error
SVM	:	Support Vector Machine
SVR	:	Support Vector Regression
STL	:	Single-Task Learning
SIANN	:	Space Invariant Artificial Neural Network
SROCC	:	Spearman Rank Correlation Coefficient
WN	:	White Noise

LIST OF APPENDICES

Appendix A: Coding for Creating Database of The Image	58
Appendix B: Coding to call images from database	60
Appendix C: Coding for Feature Extraction	61
Appendix D: Coding for MTL Train-Test Model Performance	62



CHAPTER 1

INTRODUCTION



1.1 Introduction

There are numerous ways to rate the quality of perceptual images. One popular way that has gained much research attention is the image quality assessment (IQA) model.

IQA is a process to evaluate how well a picture is presented. A real index is employed, and it must be extremely compatible with the subjective index used by humans. The highest quality in IQA is widely acknowledged to be human ratings.

These human ratings are often obtained through processes where people are asked to evaluate original images that are presented to them in accordance with the scores given. In order to determine the mean opinion score (MOS) or average opinion value difference (DMOS), the ratings across all participants are then averaged. The

measurement shows the evaluation techniques applied to the picture analysis. However, this method is time-consuming and cannot be used in practical situations. It is more practical to use objective IQA models that can automatically produce quality measures consistent with MOS/DMOS values.

There are two main categories of objective IQA, which are blind IQA (BIQA) and full-reference IQA (FR-IQA). When predicting the performance of a naturally distorted image, FR-IQA model assesses all the information between the reference image and its image. A reference image is a corresponding image that is free of distortion and of high quality. Peak signal-to-noise ratio (PSNR) and mean square error (MSE) are the two FR-IQA measurements that are easiest to understand. However, they have been shown to have a limited association with human perceptual measures.

Distortion-specific (DS) models and general-purpose models are the two categories of BIQA models [1]. Individual distortion models are implemented to make DS BIQA procedures work. This is made possible by assuming that the image's distortion is well-known from the start. For example, the method assumes the quality of JPEG compressed images in [2], while the quality of a motion-blurred raw image is tested blindly in [3]. Blockage and noise objects' effects are assessed in [4] and [5] respectively. In contrast, no prior understanding of the visual distortion is required for general-purpose BIQA models. Instead, the image's quality is evaluated to determine how much distortion it has, similar to how an image database is impacted. As visual examples, common IQA databases such as the LIVE [6] and the CSIQ [7] might be used. Using these database examples and their supplied MOS/DMOS values, the models are then trained to estimate the MOS/DMOS of the image.

1.2 Problem Statement

The current blind IQA (BIQA) models, both DS and general purpose, typically train their prediction separately for different image distortions, without considering the relationship between these learning tasks. Consequently, a BIQA model may perform well when tested on images affected by one type of distortion, but it may not be as effective when tested on other distortions. The main aim of this project is to develop a new model by studying a new learning framework that allows for the simultaneous training of a model under diverse distortion scenarios.

1.3 Objective

The main goal of this project is to develop a new model by studying a new learning framework that allows for the simultaneous training of a model under diverse distortion scenarios. The following project objectives must be fulfilled to achieve this goal:

- i. To extract spatial domain image features that are relevant for predicting image quality.
- ii. To utilize a trace-norm regularization-based learning technique to train the model simultaneously for different image distortion classes using Support Vector Machine (SVM).
- iii. Select appropriate model to predict quality score based on distortion identified in the image.
- iv. To evaluate the performance of the developed model by comparing it with various existing BIQA models in terms of its prediction accuracy, generalization capability, and computational requirements.

1.4 Scope of Work

The project focuses on BIQA model, thus will not cover the FR-IQA and models. This model proposes to utilize IQA features that are based on Natural Scene Statistic (NSS). The feature of the model is created by spatially extracting image data. These features are produced without the requirement for any kind of transformation by combining the normalized brightness of a picture with the statistical qualities of the GM and LOG operators. Then, these characteristics are applied to simultaneously learn regression models for various distortion scenarios. For the training, a trace-norm regularization-based learning methodology is utilized, and the model selects a specific regression technique for calculating the quality score of a picture given in a certain distortion. The model also uses a support vector machine (SVM) classifier to estimate separate images' different distortions with an unknown distortion. Probability estimates from the categorization model are used to weigh the picture evaluation scores from several regression models. The weighted scores are added together to determine the final quality score. The model is trained and tested using images from the LIVE databases in order to assess its performance.

CHAPTER 2

BACKGROUND STUDY



2.1 Introduction

The background study for this project draws upon various sources including research journals, conference papers, and previous reports to acquire comprehensive information and relevant knowledge on IQA. The theoretical foundations of IQA and previously developed model approaches serve as valuable references and guidance for this project. This chapter provides an overview of general-purpose models related to this project. These models can be categorized into two main groups which are NSS (Natural Scene Statistics) based models and learning-based models. Within the NSS based models, there are two major subgroups which are transform-based approach models and transform-free approach models. In addition, the learning-based models can be further classified into 2 categories which are general learning-based approach models and, codebook-based approach models.

2.2 Natural Scene Statistic (NSS) Based Model

The NSS-based model operates under the assumption that natural images possess distinct data analysis characteristics that undergo changes when distortion is introduced. By extracting features that indicate the extent to which these statistics deviate in an affected image, the model can estimate the image quality. The NSS approach involves designing handcrafted features based on prior knowledge, making it a manual process. The NSS-based models can be further divided into two categories which are transform-based models and transform-free models.

2.2.1 Transform-Based Approach Model

In [8], S. Gandhe and Omkar S suggested a new hybrid warping technique to assess the visual quality of stitched images, without relying on visual perception. Stitching images involves merging sequential photographs captured from a stationary camera with substantial overlap to achieve seamless panoramic view. In practical scenarios, it is challenging to obtain a clear and flawless panoramic image of a particular due to the inherent warping effect. To improve alignment of the images during the stitching process, a Hybrid Warping Technique has been developed. This technique incorporates two global warps and one local warp, aiming to refine the picture alignment step. The proposed method utilises this warping technique to enhance the overall quality of the stitched image. Homograph Screening is used to repair the problem of perspective distortion, and Edge Strength Similarity is employed to examine structural anomalies. To evaluate the objective quality of stitched images, various models are employed, including the Blind Image Quality Index (BIQI) [9], the Blind/Reference-less Image Spatial Quality Evaluator (BRISQUE) [10], and the Blind Image Integrity Notator utilizing DCT Statistics (BLINDS-11) [11]. These models are used to measure the quality of stitched images without relying on reference images. In

experimental tests, it was observed that the blind picture quality score obtained from the proposed method is significantly higher compared to existing methods. These findings indicate that the suggested approach performs exceptionally well in assessing the quality of stitched images.

2.2.2 Transform-Free Approach Model

In [13], Yan Fu and Shengchun Wang proposed a No Reference Image Quality Assessment Metric that is based on visual perception. The model introduces a general-purpose BIQA (Blind Image Quality Assessment) method that effectively integrates human vision qualities into the quality evaluation domain. One of the key contributions of the study is the description of a novel algorithm for extracting salient regions from images. This algorithm aims to identify the visually significant regions in an image, which play a crucial role in determining its overall quality. In the Itti model, two additional graphs depicting texture and edging characteristics were incorporated from the original image. The study utilized standardized brightness correlations of actual photographs, which conform to a simplified Gaussian probability distribution. This approach enabled the retrieval of quantitative properties in both regions of interest (ROI) and regions of non-interest, allowing for a comprehensive analysis of image regions based on their visual attributes. The gathered structures are subsequently merged to generate data that is used for training a Support Vector Regression (SVR) model. Finally, the image quality is assessed using the IQA (Image Quality Assessment) model developed during the training phase. Based on experimental results, this technique demonstrates a higher evaluation effectiveness compared to existing analytical algorithms. Additionally, the anticipated outcome is more closely aligned with human subjective experience, enabling accurate emulation of the human visual impression of image quality.

In [14], Wufeng Xue, Xuanqin Mou, et al. proposed an alternative approach for predicting image quality. They developed a BIQA (Blind Image Quality Assessment) model that is based on image local contrast features. The model considers significant structural information that closely aligns with human perception of images. The two local contrast features used in the model are gradient magnitude (GM) and Laplacian of Gaussian (LOG). These features are jointly normalized, and their distributions are utilized to estimate image quality without requiring a reference image. This approach enables the assessment of image quality in a blind manner, without the need for a comparison to a known high-quality image.

2.3 Learning Based Models

The learning-based models depend on many features that correspond to factors that influence the image quality captured compared to the NSS-based model.

2.3.1 General Learning Based Approach

The Convolutional Neural Network (CNN) is a powerful tool that can be utilized in various applications for visual imagery assessment. CNNs make use of feature maps, which are generated by convolutional kernels or filters sliding across input features. These feature maps contribute to the robust architecture of the CNN, aiding in the identification of distinct patterns and enabling effective partitioning solutions. CNNs are often referred to as shift invariant artificial neural networks or space invariant artificial neural networks (SIANN) due to their ability to recognize patterns regardless of their location within the image. Surprisingly, many deep neural networks exhibit equivariance rather than consistency when it comes to translation. However, in [12], the authors introduce a CNN-based architecture that can accurately estimate the quality of an image without requiring the original image. Unlike previous

approaches that relied on hand-crafted features, this architecture leverages image patches as feedback to capture the relevant structures within the image pixels. By utilizing image patches instead of pre-defined features, the proposed method aims to improve the accuracy and effectiveness of image quality estimation. The network architecture consists of a single convolutional layer with max and min pooling, followed by two fully connected layers and an output node. This design combines feature learning and regression into a single optimization process within the network structure, resulting in a more efficient model for evaluating image quality. The proposed method achieves improved performance on the LIVE dataset, surpassing current state-of-the-art approaches. Additionally, it demonstrates excellent adaptability when tested on different datasets, indicating its wide applicability. Furthermore, the CNN-based model shows promising results in assessing local quality distortions, which has been a relatively understudied aspect in previous research.

Weipeng Cai, Cien Fan, Lian Zou, Yifeng Liu, Yang Ma, and Minyuan Wu [15] proposed a CNN structure called CGFA-CNN (Creative Gradient Feature Aggregation CNN) for the purpose of Blind Image Quality Assessment (BIQA). The model employs a two-stage technique, where Sub-Network I is responsible for identifying the type of distortion present in an image, and Sub-Network II quantifies the severity of that distortion. To account for both realistic and artificial distortions, the hierarchical features are derived from two sources: a CNN trained on a self-built dataset specifically designed for BIQA, and a CNN trained on the large-scale ImageNet dataset. These combined features provide a comprehensive representation of the image's quality characteristics and enhance the overall performance of the CGFA-CNN model in assessing image quality.

In [17], Kede Ma, Zhengfang Duanmu, and Wangmeng Zuo propose a multi-task end-to-end optimized deep neural network called MEON (Multi-task End-to-End Optimized Network) for Blind Image Quality Assessment (BIQA). This model aims to predict the perceived quality of video frames without access to their corresponding reference frames. This is a significant challenge in computer vision that has yet to be effectively addressed. The MEON model tackles this challenge by leveraging the power of deep neural networks to learn and extract relevant features directly from video frames, enabling accurate quality assessment without relying on reference frames. A proposed deep learning perform is developed using ground truth data collected from various sources, such as subject-rated images. Usually, the learning and implementation of this function are planned independently. With the advent of deep neural network (DNN) techniques, a comprehensive end-to-end data driven BIQA explanation is now probable. The authors develop MEON from the beginning to finish, for quality prediction and distortion identification. Its performance attributes are shown to be cutting-edge. They also explain how to improve the efficiency of the system through various training techniques, such as multi-task learning and GDN.

In [19], S. Alireza Golestanch and Kris Kitani propose a simple yet effective framework for Blind Image Quality Assessment (BIQA). There are two types of BIQA methods: distortion-based and general-purpose. Distortion-based methods focus on specific types of image distortions, while general-purpose methods aim to assess image quality without knowing the specific distortions present. The proposed framework falls into the general-purpose category and uses multi-task learning techniques to capture and utilize diverse features for accurate quality assessment. It provides a versatile and adaptable approach to handle various image distortions and variations. Distortion-based techniques are limited to specific types of image

distortions, while general-purpose techniques can assess image quality regardless of the distortion category. General-purpose approaches use informative features that apply to various types of distortions. However, the effectiveness of these techniques depends on the careful design of multiple elements, which greatly affects their performance.

2.3.2 Codebook Based Approach

Several general-purpose BIQA models have been developed, which learn features directly from the pixels of the original image. One of the pioneering contributions in this approach can be found in [16]. The Codebook Representation for No-reference Image Assessment (CORNIA) model employs a method where unprocessed patches are randomly sampled from an image initially. In the CORNIA model, after sampling patches from the image, a codebook is used to compress and encode these patches. The encoded patches are then pooled together to form image-level features. CORNIA shares a similarity with CBIQ (another BIQA model) in that both models utilize a visual codebook in the feature extraction stage. However, unlike CBIQ, which employs features extracted from Gabor-filter responses, CORNIA constructs its codebook using raw image patches. This approach allows CORNIA to handle imbalanced datasets and effectively capture the diverse characteristics present in the image patches.

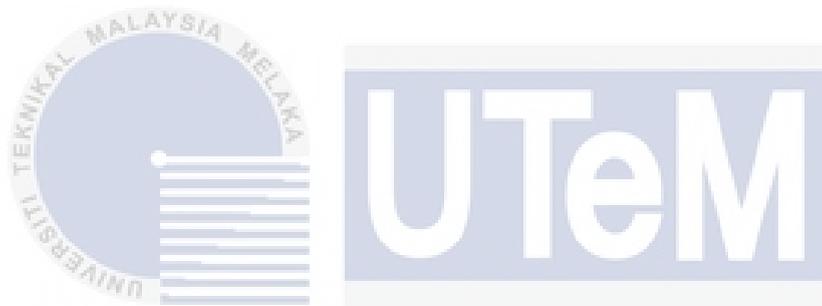
In [18], J. Xu, Peng Ye, Qiaohong Li, Haiqing Du, Yong Liu, and D. Doermarn propose a Blind Image Quality Assessment (BIQA) method based on High Order Statistics Aggregation. The primary objective of BIQA research is to develop a perceptual model that can effectively and accurately analyze image quality attributes in the absence of non-distorted reference images. The proposed method utilizes High

Order Statistics Aggregation to capture and aggregate complex statistical characteristics of images, enabling a comprehensive assessment of image quality. This approach addresses the challenge of evaluating image quality without direct access to reference images and contributes to the field of BIQA by providing a robust and efficient assessment method. The most recent general-purpose BIQA techniques can in fact be divided into two classes based on the sort of features used. The first is made up of evolved traits that are obtained from innate statistical regularities in images. Contrarily, these are improper for images that include text or fake graphics. The second group includes learning-based features, which usually demand a big codebook or supervised codebook update techniques, in order to obtain acceptable results. These take a lot of time to apply and are ineffectual.



CHAPTER 3

METHODOLOGY



3.1 Introduction

The flow among the procedure and the methods used throughout the project is presented in this chapter. This chapter also includes a detailed explanation about the software implementation, the features extraction, quality estimation and project performance testing.

3.2 Flow Chart of Methodology

To successfully accomplish this project, it is essential to address and execute certain methods and procedures. The project's flowchart is presented in Figure 3.1. The initial stage of this project entails conducting a background study and analyzing existing methods and approaches. Following the analysis, the GMLOG model [2] is chosen as the primary reference for developing the proposed model.

The second stage of this project is to identify proper image features for quality estimation. This feature extraction process is related to the first objective of the project. These features are obtained from the spatial domain. The adopted quality-predictive features are essentially GMLOG features. They are derived from the statistical characteristics of the image's Gradient Magnitude and Laplacian of Gaussian operators with the luminance of image operators within the spatial domain of the image.

The third stage of this project focuses on achieving the second objective, which involves developing a regression model algorithm. The goal is to train the quality prediction models for various types of distortions simultaneously using a technique called Trace-norm Regularization based Learning Framework utilizing the extracted features. During the training process, the framework learns to produce proper weight to each distortion types contained in the training images. These weight together with information obtained in the next stage (Distortion Identification) will be used to generate predicted quality score for any test image.

The fourth stage of this project is distortion identification. The proposed method begins by determining the different aspects of distortion present in an unspecified test image. The extracted feature data is then inputted into an SVM classifier during the

process. The decision to employ SVM is based on its proficiency in handling high-dimensional data spaces and classification tasks.

Next, the fifth stage of this project involves model testing, which is performed by evaluating the trained model using the LIVE database. The testing follows the standard train-test methodology commonly employed by previous Blind Image Quality Assessment (BIQA) models.

The final stage of this project is evaluating the performance of the model. A comprehensive assessment is conducted by comparing the model's outcomes with those of other models. The objective is to evaluate the accuracy, generalization, and speed of the model's performance.

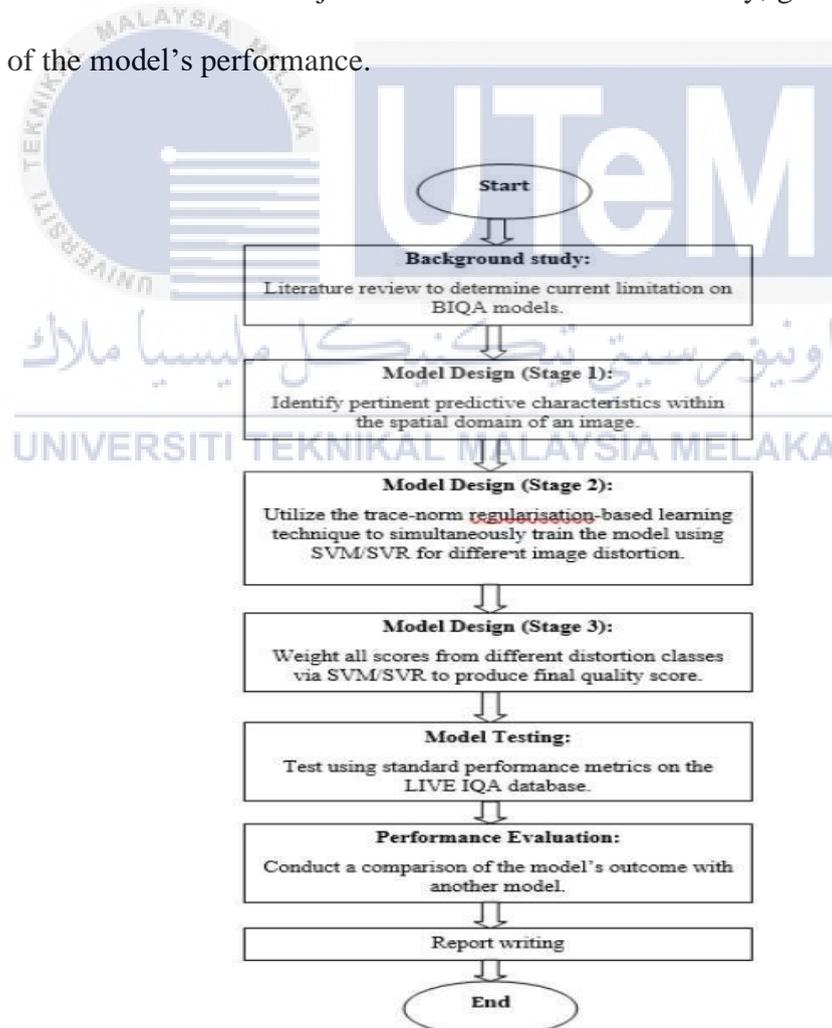


Figure 3.1: Flowchart of the project

3.3 General Framework for The Proposed Model

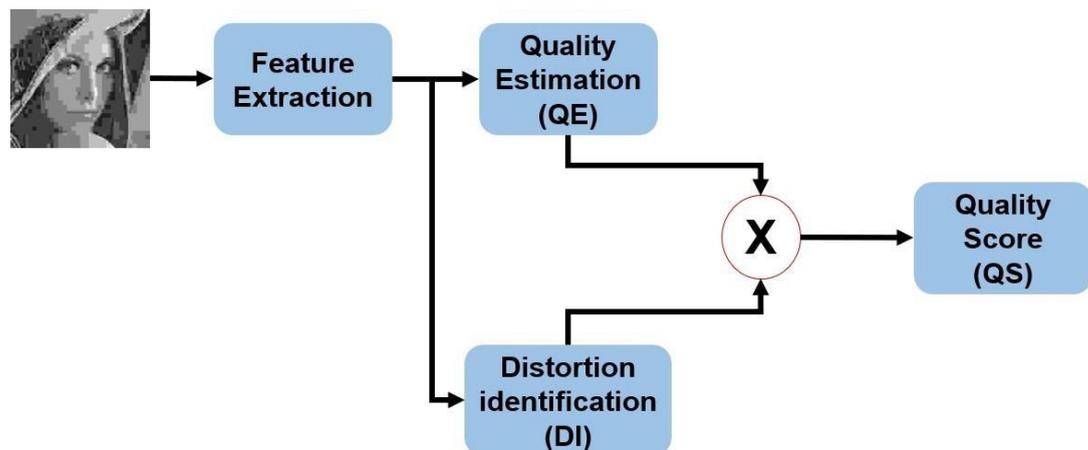


Figure 3.2: The framework of the proposed model

The process starts with Feature Extraction, where certain characteristics or data are extracted from the initial image. Following that, there's Quality Estimation (QE) which estimates the quality based on extracted features. In parallel, there's Distortion Identification (DI) represented by a folder icon and line connecting it to QE; it identifies distortions in images. Both QE and DI feed into a multiplication symbol labeled 'X' indicating some form of combination or interaction between these two processes. Finally, there's Quality Score (QS) as an output, represented as a blue rectangle; this is presumably calculated based on both QE and DI.

3.4 Database Creation

To facilitate easy access to the images, a new database is first created using MATLAB. The file used in MATLAB should be in the form of mat file. The database comprises a matrix cell with the following columns for 982 images: image size in column 1, DMOS (Distortion Mean Opinion Score) in column 2, type of distortion in column 3, reference image in column 4 and the features extraction in column 5 as shown in Figure 3.3. The main purpose of this database is to simplify the process of

retrieving the images during sampling, feature extraction, training, and testing. The code for creating this database can be found in the appendices section. Once the database is created, all the images within it are saved as image mat files, which allows for easier access. Prior to displaying the image and performing feature extraction, it is necessary to load the image mat file and then retrieve the image from the database.

	1	2	3	4	5	6
1	512x768 do...0		1	23	1x40 double	
2	505x632 do...28.0038		1	20	1x40 double	
3	512x768 do...34.0107		1	8	1x40 double	
4	453x618 do...65.1314		1	22	1x40 double	
5	505x634 do...68.9113		1	2	1x40 double	
6	453x618 do...65.1501		1	22	1x40 double	
7	505x634 do...54.3973		1	2	1x40 double	
8	512x768 do...44.3971		1	6	1x40 double	
9	482x627 do...0		1	24	1x40 double	
10	720x480 do...47.4300		1	9	1x40 double	
11	512x768 do...0		1	27	1x40 double	
12	720x480 do...41.4125		1	14	1x40 double	
13	512x768 do...49.5811		1	23	1x40 double	
14	512x768 do...48.8432		1	15	1x40 double	
15	512x768 do...26.1379		1	17	1x40 double	
16	512x768 do...0		1	7	1x40 double	
17	512x768 do...62.4787		1	6	1x40 double	
18	488x610 do...37.5989		1	4	1x40 double	
19	512x768 do...19.9666		1	29	1x40 double	
20	512x768 do...0		1	26	1x40 double	

Figure 3.3: Project database generated from LIVE database

3.5 Feature Extraction

The initial stage of the framework involves extracting image characteristics suitable for Blind Image Quality Assessment (BIQA). The chosen feature set for the project closely resembles the implementation of the GMLOG model. It consists of four statistical distributions generated by applying two contrast enhancement techniques to the images: the gradient magnitude (GM) and the Laplacian of Gaussians (LOG). The GMLOG model has shown that these distributions exhibit variations in distorted

images compared to the consistent distributions gradually change as the level of distortion increases, indicating their analytical relevance to image quality and their suitability as features for BIQA tasks. As to demonstrate that GM and LOG are effective predictors of local image quality, the subsequent section will elaborate on how the joint statistic of GM and LOG can be applied to the BIQA problem.

3.5.1 Gradient of Magnitude (GM) and Laplacian of Gaussian (LOG)

Local contrast features capture the structural aspects of an image. It is sensitive to changes in the image's structure, such as variations in shape and the number of edges. For a given image I , the corresponding Gradient Magnitude (GM) map $G1$ is as follows:

$$G1 = \sqrt{[I \otimes h_x]^2 + [I \otimes h_y]^2} \quad (3.1)$$

Where “ \otimes ” is the linear convolution operator and $h_d, d \in \{x, y\}$, the Gaussian second derived filter implemented horizontally (x) and vertically (y):

$$\begin{aligned} h_d\{x, y|\sigma\} &= \frac{\delta}{\delta} g\{x, y|\sigma\} \\ &= -\frac{1}{2\pi\sigma^2} \frac{d}{\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right) d \in \{x, y\} \end{aligned} \quad (3.2)$$

Where $g\{x, y|\sigma\} = -\frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2+y^2}{2\sigma^2}\right)$ is the scale controlled isotropic Gaussian function σ . Meanwhile, the LOG of image I is:

$$F_1 = I \otimes h_{LOG} \quad (3.3)$$

Where:

$$\begin{aligned}
h_{LOG}(x, y | \sigma) &= \frac{\delta^2}{\sigma^2} g(x, y | \sigma) + \frac{\delta^2}{\sigma^2} g(x, y | \sigma) \\
&= -\frac{1}{2\pi\sigma^2} \frac{x^2 + y^2 - 2\sigma^2}{\sigma^4} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right)
\end{aligned} \tag{3.4}$$

3.5.2 Joint Adaptive Normalization

Both the GM and LOG operators have the capability to reduce spatial discontinuities in an image while preserving certain connections between data points. Typically, these methods involve dividing an image into channels with different scales and orientations. Each channel is then smoothed by averaging the values within neighborhood centered on the current coefficient, weighted by the energy input. This divergence normalization technique effectively enhances the variables and eliminates local variations in contrast, resulting in a more consistent statistical characteristic. In this work, standardizing the GM and LOG coefficient is necessary to obtain a stable statistical representation of the image. The standardization of these operators can be expressed as follows:

$$F_1(i, j) = \sqrt{\frac{G^2(i, j) + A^2(i, j)}{1}} \tag{3.5}$$

In equation 3.5, F_1 represent a parameter that controls local homogenization of compatibility, while ε_{GMLOG} denotes a continuous reduction of arithmetic inconsistency. The data normalization factor for each location, in accordance with the GMLOG work at position (i, j) is determined as follows:

$$N_1(i, j) = \sqrt{\sum \sum_{(l, k) \in n_{i, j}} \omega(l, k) F_1^2(l, k)} \tag{3.6}$$

In equation 3.6, Ω_i indicates a temporary window centered at (i, j) , $\omega(l, k)$ are positive weight satisfying $\sum_{l,k} \omega(l, k) = 1$. In this project implementation, $\omega(l, k)$ is set to be a rescaled temporally compressed Gaussian kernel. The GM and LOG feature maps have been calibrated as follows:

$$\begin{aligned}\bar{G} &= G_1/(N_1 + \epsilon), \\ \bar{A} &= A_1/(N_1 + \epsilon),\end{aligned}\tag{3.7}$$

Where ϵ represents a small individual investor and serves as a constant to prevent numerical instabilities. The adjustment process that follows is referred to as joint adaptive adjustment (JAN). Since JAN solely impacts the local contrast scale, it does not alter the semantic structures within the image. The advantages of JAN include ensuring the reliability of the local contrast scale in GM and LOG maps throughout the image, eliminating uncertainties caused by variations in lighting, varying edge magnitudes, and other structural elements. Furthermore, JAN enhances the correlation between horizontal, vertical, and diagonal features in the image, while the normalization process stabilizes the profiles of these features.

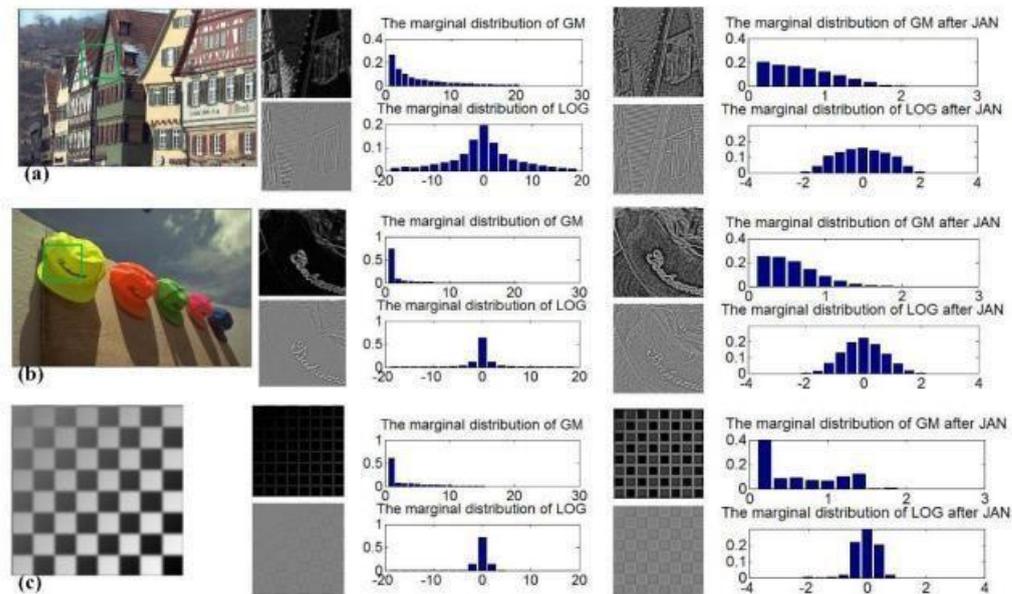


Figure 3.4: The GM and LOG maps together with their marginal distribution before (middle column) and after (right column) joint adaptive normalization.

(a) Houses; (b) Hats; and (c) Chessboard

Figure 3.4 illustrates the GM and LOG mappings of three images following the application of Joint Adaptive Normalization (JAN). Additionally, the figure displays the marginal distributions of the respective GM and LOG maps. With JAN applied, the GM and LOG maps remain constant across the entire image. The right column of the figure reveals that the GM and LOG distributions of the natural images, Houses and Hats, become nearly identical, despite their distinct image contents. However, for the fabricated image, the chessboard, the GM and LOG distributions appear significantly distinct from those of the natural images, Hats and Houses.

3.5.3 Statistical Feature Description

Following the application of JAN to the GM and LOG characteristics, the empirical distribution of $\overline{g} = (i, j)$ and $\overline{l} = (i, j)$ can be computed and utilized in the

development of a statistical method. Specifically, $\bar{G} = (i, j)$ is quantized into M levels $\{g_1, g_2, \dots, g_m\}$, while $\bar{L} = (i, j)$ is quantized into N levels $\{l_1, l_2, \dots, l_N\}$. To simplify notation, $\bar{G} = (i, j)$ is represented as G and $\bar{L} = (i, j)$ is represented as L .

the joint empirical probability function of G and L can be denoted as

(3.8)

$$K_{m,n} = P(G = g_m, L = l_n), m = 1, \dots, M; n = 1, \dots, N$$

To clarify, the bivariate histogram $K_{m,n}$ represents the calibration of G and L . While $K_{m,n}$ contains a substantial amount of statistical data within the region corresponding to $\bar{G} = (i, j)$ and $\bar{L} = (i, j)$, it possesses numerous characteristics ($M \times N$).

Instead of utilizing $K_{m,n}$ to develop the prediction model, it is preferable to extract a smaller set of performance features from $K_{m,n}$ for this task. Probabilistic reasoning suggests that the marginal probability functions of $\bar{G} = (i, j)$ denoted as P_G and $\bar{L} = (i, j)$ denoted as P_L , serve as simple alternatives:

$$P_{G_1}(G_1 = g_m) = \sum_{n=1}^N K_{m,n} \quad (3.9)$$

and,

$$P_{L_1}(\bar{L}_1 = l_n) = \sum_{m=1}^M K_{m,n} \quad (3.10)$$

Due to the application of the JAN process, the marginal probability functions P_G and P_L of typical images with different contents tend to exhibit similar structures. However, when a natural image is influenced by external factors, such as low quality or distortions, the forms of P_G and P_L may deviate from those observed in high-quality

natural images.



Figure 3.4 below shows the marginal probability functions of the LIVE database's distorted images of a reference image.

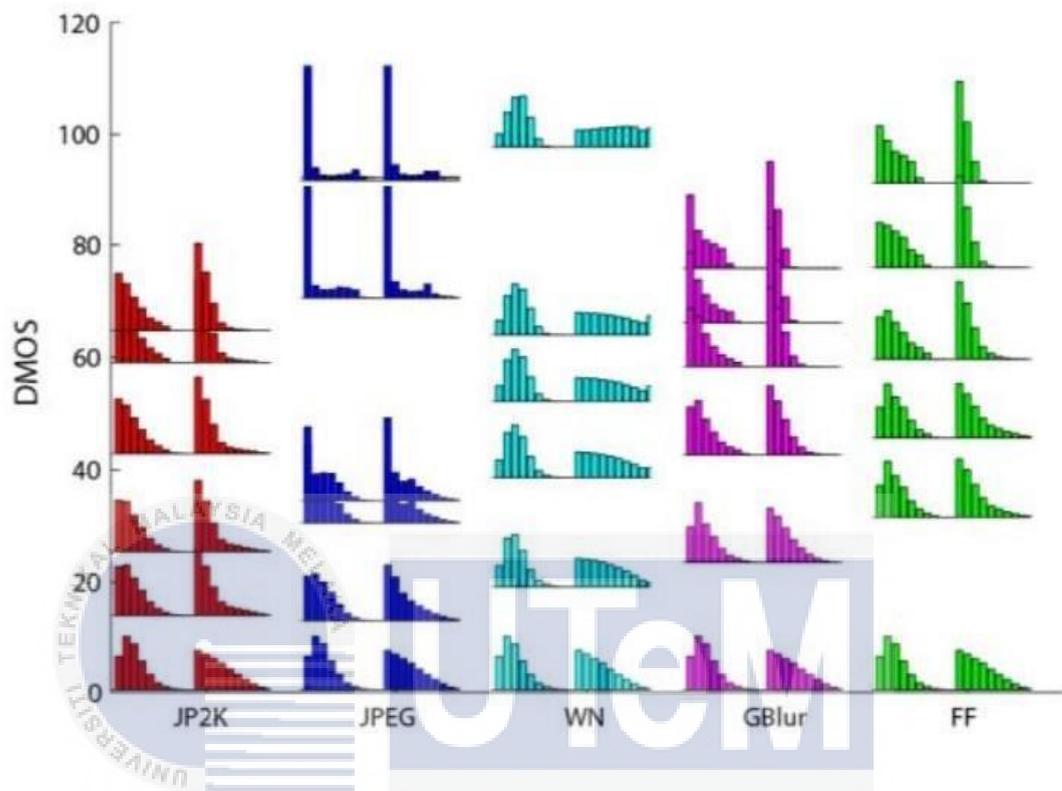


Figure 3.5: Marginal probability function (P_G and P_L) of the distorted images produced at different DMOS values for one reference image.

Based on Figure 3.5, the probabilities of the distorted picture's P_G and P_L at the edges can be observed. These probabilities are derived from the original reference images, which is subjected to various levels of DMOS related to quality scores determined by humans. The images used in this analysis are from the LIVE database and encompass five types of distortion, which are JP2K compression, JPEG compression, white noise (WN), gaussian blur (GB), and fast fading (FF). By examining the histograms of P_G and P_L at different DMOS levels to further explain how the marginal distribution change with the extent of degradation. The results

demonstrate that as the distortion level increases, both P_G and P_L undergo significant alternations.

As an illustration, let's us consider the case of JPEG compression. As the DMOS increases, the distribution's shape undergoes corresponding changes. The severity of distortion affects the pattern of the graph, enabling the computer to learn and identify scores based on the graph's pattern and its rate of change. Consequently, the information contained within the histogram can be leveraged for predictive learning. In the case of a test image, its features are extracted, and a distribution is generated. If this distribution resembles the patterns observed earlier, the computer can predict the quality score. This observation highlights that the pattern of P_G and P_L can effectively train the computer to estimate the quality score based on each specific type of distortion.

The marginal probability functions P_G and P_L do not represent the interconnections among an image's GM and LOG features. Consequently, when $K_{m,n} = P_G(G = g_m) \times P_L(L = l_n)$ holds true for all m and n, there's potential to explore the following index to evaluate the association between GM and LOG.

$$D_{m,n} = \frac{K_{m,n}}{P_G(G = g_m) \times P_L(L = l_n)} \quad (3.11)$$

Opting for a direct computation and utilization of $D_{m,n}$ is not the most optimal choice. Rather, it is more effective to compute the dependency of each individual value, $G = g_m$, against all potential values of L. By using the marginal probability $P(G = g_m)$ as a weighting factor, define the subsequent measure to gauge the overall dependency of $G = g_m$ on L:

$$Q_G(G = g_m) = P(G = g_m) \cdot \frac{1}{M} \sum_{m=1}^M D_{m,n} \quad (3.12)$$

Define a total dependency measure, $L = l_n$ on G using a similar approach as previously described for $G = g_m$ on L:

$$Q_L(L = l_n) = P(L = l_n) \cdot \frac{1}{N} \sum_{n=1}^N D_m, \quad (3.13)$$

It is straightforward to demonstrate that $Q_G \geq 0$ and $Q_L \geq 0$. Moreover, $\sum_n Q_L(L = l_n) = 1$. Consequently, it can be regarded as probability distributions in certain respects, known as independence distributions.

Then Q_G and Q_L can be re-written as:

$$Q_G(G = g_m) = \frac{1}{N} \sum_{n=1}^N \frac{P(G = g_m), (L = l_n)}{P(L = l_n)} \quad (3.14)$$

$$= \frac{1}{N} \sum_{n=1}^N P(G = g_m), (L = l_n)$$

$$Q_L(L = l_n) = \frac{1}{M} \sum_{m=1}^M \frac{P(G = g_m), (L = l_n)}{P(G = g_m)}$$

$$= \frac{1}{M} \sum_{m=1}^M P(L = l_n), (G = g_m) \quad (3.15)$$

By examining equations (3.14) and (3.15), it becomes evident that the proposed dependency measure can be seen as the sum of conditional probabilities for a particular value of G (or L) across the variable L (or G). This method utilizes arithmetic analysis that adjusts GM with the LOG characteristic.

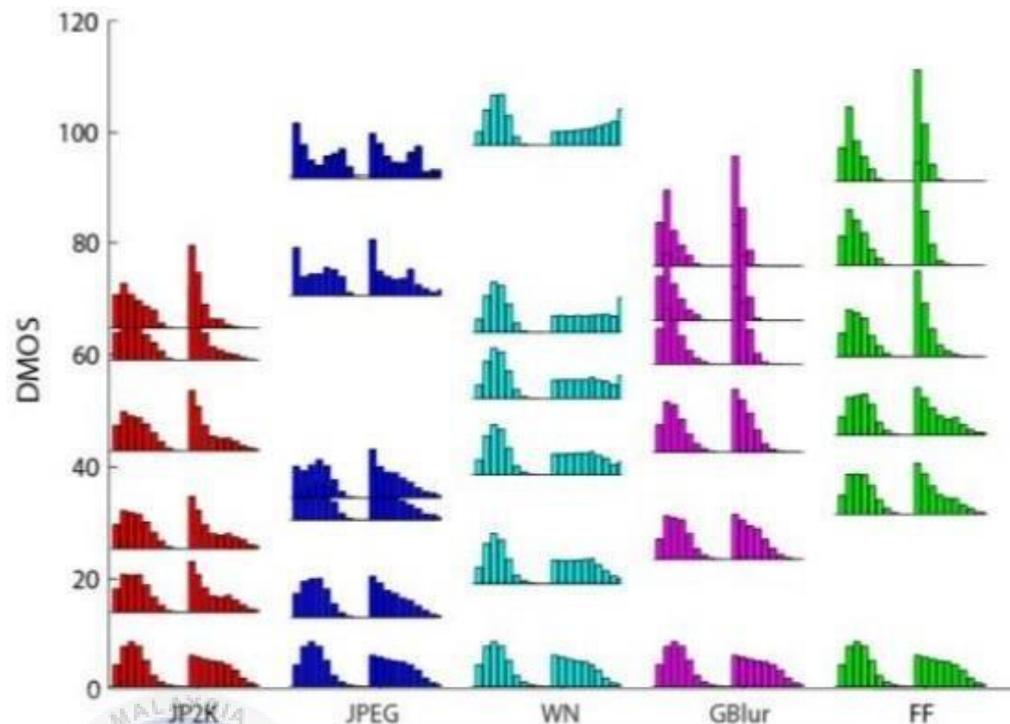


Figure 3.6: The independency distribution (Q_G and Q_L) represents the distribution of the independence among distorted images generated at various DMOS values concerning a single reference image.

The histogram, tracing the sequence of image feature data from a specific original image in the LIVE IQA database, exhibit the relationship between two properties (Q_G and Q_L), which serve as predictors of image quality. Figure 3.6 illustrates this plot. It is apparent that the histogram shape for each distortion type begins to alter with variations in the distortion's intensity.

Because of the connection between the GM and LOG operators, the model suggests deriving the next two BIQA feature properties from this relationship. MTL assesses the dependency of a specific variable by computing the statistical interaction between both operators $G = g_m$ across all potential combinations of L, and conversely. These computations can be formulated as follows:

$$Q_G(G = g_m) = \frac{1}{N} \sum_{n=1}^N P(G = g_m | L = l_n) \quad (3.16)$$

and,

$$Q_L(L = l_n) = \frac{1}{M} \sum_{m=1}^M P(L = l_m | G = g_n) \quad (3.17)$$

The distributions of P_G, P_L, Q_G and Q_L forms the image features set for the project.

3.6 Quality Estimation

The feature vector obtained is subsequently inputted into quality prediction models that have been trained to assess the quality of an image across different distortion scenarios. In contrast to previous methods in image quality assessment (BIQA), which employed single-task learning (STL) to train prediction models, this project adopts a trace-norm regularization-based learning approach to simultaneously learn its prediction models. In this trace-norm regularization-based learning technique, the prediction models for each distortion are considered as separate learning problems but being trained collectively.

The proposed model aims to minimize the following objective function given a set of training image and feature vectors:

$$\min_w E(W) = f(W) + \Omega(w), \quad (3.18)$$

Where $\Omega(W)$ is the linearization factor, which represents the correlation between the tasks, while $f(W)$ represents the empirical loss observed in the training set. In the case BIQA, $f(W)$ is expressed as a loss function $P(\cdot, \cdot)$ as:

$$(3.19)$$

$$f(W) = \sum_{j=1}^n \sum_{i=1}^{R_i} P(S_i, \omega_{ti} X_j^T)$$

Where n is the number of distortion classes, S_i denotes the number of samples in distortion. The variable x_j^i and y_j^i refer to the j th feature vector and the corresponding DMOS value in the i th distortion, respectively and $W = [\omega_1, \omega_2, \dots, \omega_n]$ where ω represents a parameter to be estimated from the training samples.

The suggested model is trained employing a trace-norm regularization technique, under the assumption that the distorted classes are interrelated, and the extracted features have a high dimensionality. To capture the correlation between tasks, a low-dimensional subspace learning approach was utilized, ensuring that models from different tasks share a common low-rank structure. Figure 3.7 visualizes the training structure of the proposal model, implementing trace-norm regularization.

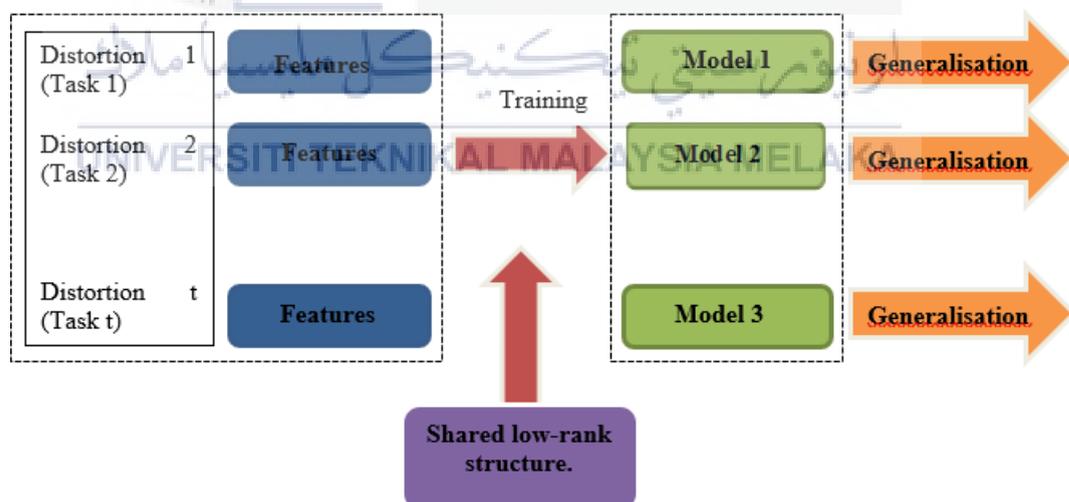


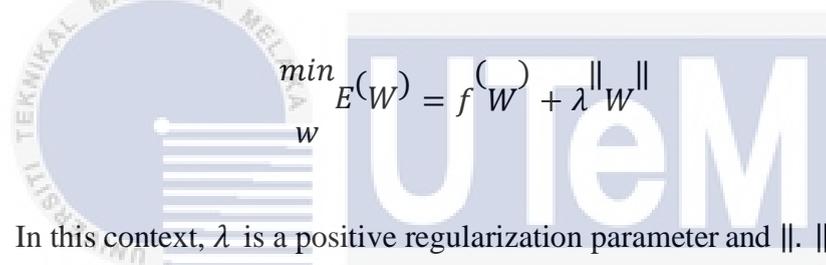
Figure 3.7: Training framework of trace-norm regularization

The objective function in trace-norm simplified technique is formulated as a problem minimizing the matrix rank. This formulation captures the share low-rank

structure among the tasks. The technique considers equation in (3.18) as a matrix rank minimization problem:

$$\min_w E(W) = f(W) + \lambda[\text{Rank}(W)] \quad (3.20)$$

Due to NP-hardness of the matrix rank minimization problem, a convex relaxation of the rank function $\text{Rank}(W)$ is commonly adopted. The trace-norm relaxation technique is widely used approximate $\text{Rank}(W)$ because it has been mathematically proven to provide a good approximation. Consequently, the problem can be reformulated as a trace-norm minimization problem, where equation (3.20) is written as follows:



$$\min_w E(W) = f(W) + \lambda \|W\| \quad (3.21)$$

In this context, λ is a positive regularization parameter and $\|\cdot\|$ * that denotes the trace-norm, defined as the sum of eigenvalues. To enhance convergence speed, the project utilizes an accelerated gradient method to solve equation (3.21) and determine the optimized values of W :

$$W = \arg \min_y \frac{1}{2} \|W - (S - \frac{1}{y}(W))\| + \lambda \|W\| \quad (3.22)$$

It represents an objective function that is being minimized with respect to the weight vector (W). The equation consists of a loss term and a regularization term which helps in preventing overfitting by penalizing large weights. The equation is solving for (W), which appears to be a weight vector in an optimization problem. Symbols like “arg min” indicate that this is an optimization problem to find the value of (W) that minimizes this expression.

3.7 Training and Testing Procedure

The LIVE database, widely utilized in previous BIQA models, serves as the training and testing dataset for this project. This database consists of images paired with corresponding quality scores that represent human perceptual measures. It contains five types of distortion, including JPEG2000 (227 images), JPEG (233 images), gaussian blur (174 images), white noise (174 images) and fast fading (174 images). Each image is associated with a difference mean opinion score (DMOS) value, which indicates its subjective quality scores. Furthermore, a training procedure is necessary for the proposed model to calibrate the regressor module. Therefore, the LIVE database is randomly divided into two subsets, namely the training test and the testing set.

The partitioning of the sets is based on the reference image numbers to ensure non-overlapping subsets. The experiments are conducted with different partition orders. Additionally, a random vector ranging from 1 to 29 is generated 1000 times. The purpose of the training set is to create the BIQA model, while the testing set is used to evaluate the model. The train-test procedure is repeated 1000 times, and then the median performance across these iterations is reported to mitigate performance biases.

3.8 Distortion Identification

After training, the models are utilized to estimate the quality score of a test image. The proposed approach initially assesses various types of distortion present in the test image with unknown distortion. The resulting feature vector is then fed into an SVM classifier. SVM was chosen due to its effectiveness in high-dimensional spaces and generalization employed. It is important to note that the objective is to evaluate each

distortion class in the image rather than perform complex classification. The classifier provides probabilities for these estimations. The predicted rating from trace-norm regularization models is weight using these probability values. Finally, the image's overall performance value is calculated by combining the weighted values.

3.9 Performance Analysis

To assess the model's performance, the Spearman Rank Correlation Coefficient (SROCC) and Pearson Correlation Coefficient (LCC) have been utilized to measure the relationship between the model's predictive score and MOS (Mean Opinion Score). SROCC determines the monotonicity of the prediction, where a value closer to 1 signifies strong alignment with human opinion. Likewise, a high LCC indicates good performance relative to human perception. Additionally, the project also employ the root mean square error (RMSE) to quantify the similarity between predicted scores and human scores. A value closer to 0 indicates superior model performance.

3.9.1 Root Mean-Squared Error (RMSE)

The root mean-square error (RMSE) is computed as the square root of the average squared difference between the estimated values from a model and the corresponding actual values. RMSE is often employed to assess the disparity between the model's predictions and the ground truth:

$$RMSE = \sqrt{\frac{1}{N} \sum_i (q_i - o_i)^2} \quad (3.23)$$

3.9.2 Pearson Linear Correlation Coefficient (PLCC)

The Pearson Linear Correlation Coefficient (PLCC) is utilized to evaluate the effectiveness of a linear relationship between two variables. The PLCC value ranges

from -1 to 1, indicating the strength and direction of the linear association. A value close to -1 to 1 signifies a strong linear relationship. The positive or negative sign of the PLCC indicates the direction of the linear relationship. The formula for calculating PLCC is as follows:

$$PLCC = \frac{\sum_i (q_i - \bar{q}) * (o_i - \bar{o})}{\sqrt{\sum_i (q_i - \bar{q})^2 * (o_i - \bar{o})^2}} \quad (3.24)$$

Where o_i is the DMOS between reference and distorted images, and q_i is a nonlinear function.

3.9.3 Spearman's Rank Order Correlation Coefficient (SROCC)

The Spearman's Rank Order Correlation Coefficient (SROCC) is utilized to assess the influence of the association between two sets of data. It is capable of accommodating both discrete and continuous data types. The SROCC is formally defined as follows:

$$SRCC = 1 - \frac{6 \sum_{i=1}^N d_i^2}{N(N^2 - 1)} \quad (3.25)$$

Where "d" represents the difference between the i -th ratings of images is conscious and unconscious evaluations.

CHAPTER 4

RESULTS AND DISCUSSION

This chapter initially outlines the experimental setup designed to test the proposed model. It is followed by observations of the findings and discussions regarding the model's performance. The results of the model are then compared with those of several previous IQA models to assess its accuracy and speed performance.

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

4.1 Result Analysis

4.1.1 Database Construction

The database is created by saving all 982 images from the LIVE database. This constructed database serves to streamline the image retrieval process. Illustrated in Figure 4.1, the completed database is structured with dimension [982x4], featuring four columns. The first column represents the image in grayscale value, while the second column represents the DMOS score, third column represents the type of distortion, and the fourth column represents the image in numerical form (label).

	1	2	3	4
1	512x768 do...	0	1	23
2	505x632 do...	28.0038	1	20
3	512x768 do...	34.0107	1	8
4	453x618 do...	65.1314	1	22
5	505x634 do...	68.9113	1	2
6	453x618 do...	65.1501	1	22
7	505x634 do...	54.3973	1	2
8	512x768 do...	44.3971	1	6
9	482x627 do...	0	1	24
10	720x480 do...	47.4300	1	9
11	512x768 do...	0	1	27
12	720x480 do...	41.4125	1	14
13	512x768 do...	49.5811	1	23
14	512x768 do...	48.8432	1	15
15	512x768 do...	26.1379	1	17
16	512x768 do...	0	1	7
17	512x768 do...	62.4787	1	6
18	488x610 do...	37.5989	1	4
19	512x768 do...	19.9666	1	29
20	512x768 do...	0	1	26

Figure 4.1: Database with 982 images

As for this project, all the reference images that have no DMOS value or 0 value, will be removed to get better prediction value of the image. After removing all the references image, then the new database is formed by storing the remaining 779 images only as shown in Figure 4.2. Note that the number in column 3 of the database is between 1 and 5. The number 1 until 5 represent the distinct image type, 1 for JPEG2000, 2 for JPEG, 3 for white noise (WN), 4 for gaussian blur (GB), and 5 for fast fading (FF) images. In the database's fourth column, labels ranging from 1 to 29 indicate reference images link to each image. This correlates with the 29 diverse reference images within the LIVE database.

779x5 cell

	1	2	3	4
1	505x632 do...	28.0038	1	20
2	512x768 do...	34.0107	1	8
3	453x618 do...	65.1314	1	22
4	505x634 do...	68.9113	1	2
5	453x618 do...	65.1501	1	22
6	505x634 do...	54.3973	1	2
7	512x768 do...	44.3971	1	6
8	720x480 do...	47.4300	1	9
9	720x480 do...	41.4125	1	14
10	512x768 do...	49.5811	1	23
11	512x768 do...	48.8432	1	15
12	512x768 do...	26.1379	1	17
13	512x768 do...	62.4787	1	6
14	488x610 do...	37.5989	1	4
15	512x768 do...	19.9666	1	29
16	512x640 do...	32.6947	1	25
17	720x480 do...	41.9223	1	9
18	438x634 do...	25.3410	1	19
19	512x768 do...	50.8396	1	26
20	512x640 do...	66.4619	1	25

Figure 4.2: Database without reference images.

4.1.2 MATLAB Operation

Once the image database was created, MATLAB performs image retrieval using the *imread* command. Figure 4.3 exhibits the code employed to access images from the established database. Following this, the *imshow* command is utilized within MATLAB to visually present the images. Figure 4.4 provides an instance displaying one of these images as an example.

```
1 w=imread('C:\Users\User\Desktop\PSM 1\LIVE database\databaserelease2\jpg2k\img10.bmp');  
2 A=rgb2gray(w);  
3 B=double(A);  
4 imshow(A);  
5
```

Figure 4.3: Code to call image into MATLAB workspace

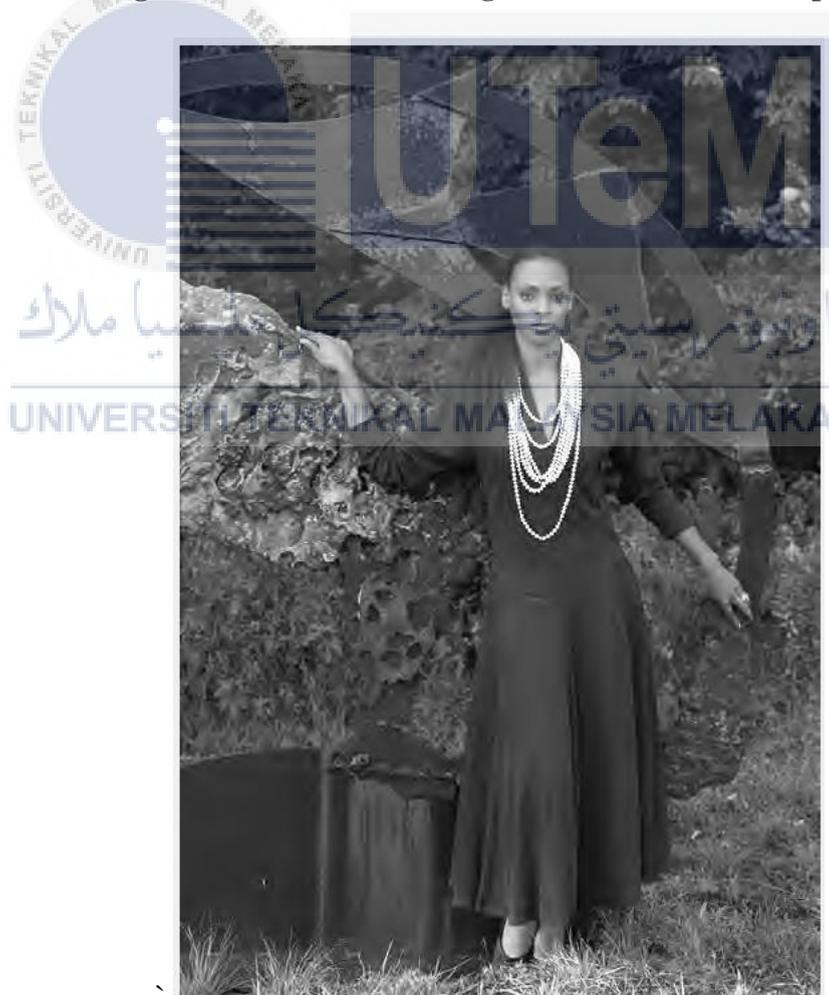
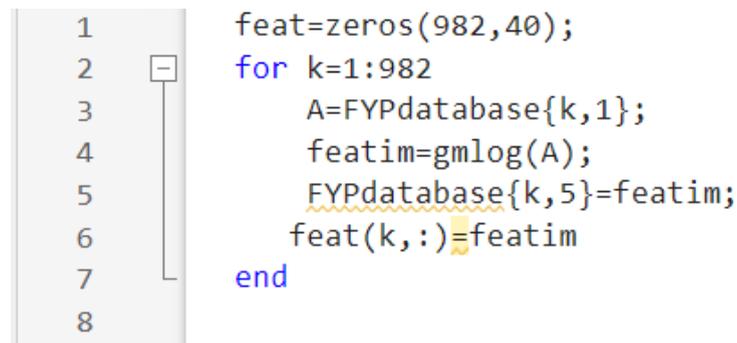


Figure 4.4: Example of image displayed in MATLAB

4.1.3 Feature Extraction

As previously stated, the model being proposed will exclusively rely on a singular set of features derived from the GMLOG algorithm. Each of feature's extraction will be performed in each image using the coding as shown in Figure 4.5.



```

1 feat=zeros(982,40);
2 for k=1:982
3     A=FYPdatabase{k,1};
4     featim=gmlog(A);
5     FYPdatabase{k,5}=featim;
6     feat(k,:)=featim
7 end
8

```

Figure 4.5: Coding for feature extraction

4.1.3.1 GMLOG Feature

In order to execute GMLOG feature extraction, the GMLOG feature command is employed, readily available online. 40 features are extracted from each image using the command as shown in Figure 4.6. Figure 4.7 illustrates the GMLOG-extracted features from each image.

	1	2	3	4	5
1	505x632 do...	28.0038	1	20	1x40 double
2	512x768 do...	34.0107	1	8	1x40 double
3	453x618 do...	65.1314	1	22	1x40 double
4	505x634 do...	68.9113	1	2	1x40 double
5	453x618 do...	65.1501	1	22	1x40 double
6	505x634 do...	54.3973	1	2	1x40 double
7	512x768 do...	44.3971	1	6	1x40 double
8	720x480 do...	47.4300	1	9	1x40 double
9	720x480 do...	41.4125	1	14	1x40 double
10	512x768 do...	49.5811	1	23	1x40 double
11	512x768 do...	48.8432	1	15	1x40 double
12	512x768 do...	26.1379	1	17	1x40 double
13	512x768 do...	62.4787	1	6	1x40 double
14	488x610 do...	37.5989	1	4	1x40 double
15	512x768 do...	19.9666	1	29	1x40 double
16	512x640 do...	32.6947	1	25	1x40 double
17	720x480 do...	41.9223	1	9	1x40 double
18	438x634 do...	25.3410	1	19	1x40 double
19	512x768 do...	50.8396	1	26	1x40 double
20	512x640 do...	66.4619	1	25	1x40 double

Figure 4.6: New database containing feature extraction data in column 5

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0.0642	0.1479	0.1843	0.1932	0.1800	0.1379	0.0738	0.0180	6.4521e-04	0	0.1602	0.1541	0.1418
2	0.0725	0.1229	0.1840	0.1901	0.1871	0.1487	0.0782	0.0160	6.0034e-04	0	0.2193	0.1947	0.1663
3	0.0598	0.1085	0.1548	0.1748	0.1893	0.1700	0.1165	0.0260	2.6117e-04	0	0.2343	0.2095	0.1790
4	0.1139	0.0769	0.1769	0.1774	0.1720	0.1485	0.1104	0.0241	9.5048e-05	0	0.2893	0.2234	0.1919
5	0.0552	0.0962	0.1470	0.1689	0.1895	0.1807	0.1325	0.0298	2.3940e-04	0	0.2457	0.2139	0.1804
6	0.1196	0.1151	0.1984	0.1873	0.1672	0.1256	0.0677	0.0185	6.3365e-04	0	0.2531	0.2030	0.1784
7	0.0888	0.0980	0.1839	0.1880	0.1919	0.1578	0.0809	0.0106	1.9365e-04	0	0.2459	0.2045	0.1752
8	0.1868	0.1223	0.2220	0.1814	0.1256	0.0932	0.0563	0.0120	3.5796e-04	0	0.2787	0.2063	0.1910
9	0.0920	0.1255	0.1761	0.1744	0.1668	0.1388	0.0949	0.0312	3.0404e-04	0	0.2185	0.1919	0.1663
10	0.0950	0.1104	0.1846	0.1877	0.1814	0.1458	0.0832	0.0117	2.6539e-04	0	0.2453	0.2036	0.1766
11	0.0846	0.1573	0.1804	0.1798	0.1681	0.1315	0.0739	0.0229	0.0015	0	0.1783	0.1662	0.1492
12	0.1996	0.1213	0.2081	0.1659	0.1308	0.0979	0.0571	0.0183	0.0010	0	0.2612	0.1814	0.1790
13	0.1790	0.1567	0.2163	0.1777	0.1361	0.0859	0.0374	0.0107	1.9324e-04	0	0.2051	0.1435	0.1377
14	0.0558	0.1118	0.1738	0.1895	0.1911	0.1617	0.0925	0.0227	0.0011	0	0.2206	0.2024	0.1714
15	0.0943	0.1086	0.1903	0.1864	0.1850	0.1490	0.0754	0.0107	2.6114e-04	0	0.2341	0.1950	0.1722
16	0.1315	0.1680	0.1960	0.1697	0.1478	0.1080	0.0584	0.0197	7.1685e-04	0	0.1813	0.1562	0.1456
17	0.0400	0.0499	0.1205	0.1552	0.2131	0.2363	0.1632	0.0219	3.0951e-05	0	0.3371	0.2558	0.1780
18	0.0825	0.0993	0.1608	0.1661	0.1707	0.1634	0.1234	0.0336	2.2932e-04	0	0.2552	0.2107	0.1760
19	0.0371	0.0663	0.1126	0.1469	0.1937	0.2284	0.1857	0.0293	8.3569e-05	0	0.2716	0.2274	0.1768
20	0.0636	0.0589	0.1526	0.1745	0.2067	0.2032	0.1280	0.0124	3.3495e-05	0	0.3222	0.2455	0.1856

Figure 4.7: Example of GMLOG feature data

4.1.4 Model Parameter

During the feature extraction phase, the setting of the model parameter was configured in accordance with GMLOG implementation. The scale parameter used for the GM and LOG operators was designated as 0.5, while the quantization level, denoted as $M=N$, was established at 10.

The database was divided into two distinct sections for model training purposes, the training set encompassed 80% of the original images along with their altered versions, while the testing set comprised the remaining 20% of the original images. Notably, there was no overlap between these two sets. For the training of the multitask learning framework, the MALSAR open-source package was utilized, implementing the trace-norm regularisation technique. Within this package, the loss function $\ell(\cdot, \cdot)$ was specifically configured as least square function. Moreover, to train the SVM for the DI stage, the open-source LIBSVM software was employed as a requisite tool.

Two types of experiment were conducted to assess the performance of the proposed method, the average performance experiment and the distortion-specific (DS) performance experiment. In the overall performance trial, termed the average

performance experiment, the train-test evaluation encompassed all images irrespective of their distortion type. This comprehensive approach gauged the effectiveness of the BIQA model across diverse types of distortion, providing insight into its overall performance. On the other hand, the DS performance experiment focused on evaluating the model's efficacy with a particular distortion class. This involved a train-test run restricted solely to photos from a single distortion category. Such targeted experimentation aimed to analyze how well the BIQA model handles specific distortions. It is worth noting that the proposed structure comprises numerous trained models tailored to different distortion classes. In the DS performance experiment, where the distortion type is known, a dedicated trained model can be directly applied for the Quality Estimation (QE) stage without the need for the Distortion Identification (DI) stage.

4.1.5 Train-test Partition

The experiment was iterated 1000 times, each time employing different sets for training and testing the model. This iterative process produced a matrix with dimensions of [1000x29]. Here, the number 1000 signifies the total experiments conducted, while 29 represents the count of reference images stored within the database. In each experiment, the 'randperm' command was utilized to generate 29 random numbers ranging from 1 to 29. This matrix layout is depicted in Figure 4.8. For every row, signifying an individual experiment, the initial 80% of the matrix data was allocated for the training set, while the remaining 20% was designated for testing purposes.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
1	22	6	3	16	11	7	28	17	14	8	5	29	21	25	27
2	3	11	5	6	26	25	1	16	22	4	9	29	13	12	18
3	10	11	19	2	16	20	14	4	12	28	18	23	22	5	9
4	12	5	4	25	9	28	14	18	26	20	16	13	2	24	11
5	7	3	11	5	20	24	19	17	13	18	27	12	29	15	4
6	29	7	25	1	20	15	3	21	4	26	2	5	18	12	27
7	1	16	2	12	10	8	22	13	28	20	21	6	19	23	24
8	25	15	21	13	8	14	16	11	26	29	9	18	2	27	20
9	12	29	9	4	3	22	17	10	5	6	27	26	7	18	15
10	20	4	10	8	19	5	17	6	23	12	22	29	25	26	11
11	4	13	11	14	29	16	20	6	1	2	27	26	10	5	9
12	14	24	6	17	1	16	2	19	26	12	21	18	3	7	9
13	13	28	10	21	12	9	17	25	3	18	24	1	26	19	27
14	19	25	13	22	23	1	4	17	7	5	21	11	2	24	20
15	2	3	17	13	19	29	18	20	4	12	7	16	21	23	25
16	17	13	25	5	29	26	7	6	2	22	10	15	19	20	18
17	11	24	9	3	21	23	5	7	10	1	15	26	27	14	12
18	27	25	26	13	24	6	11	12	1	28	3	29	16	22	9
19	23	10	8	20	7	25	22	19	28	1	15	26	14	18	11
20	5	27	9	19	8	22	6	18	13	1	15	7	11	24	4

Figure 4.8: Non-overlap random train-test partition based on the total number of references image contained in LIVE Database

4.1.6 Performance Evaluation

The project model was compared to three other BIQA models: BRISQUE [20], BIQI, and CORNIA [21]. The assessment of these models' effectiveness involved the use of three metrics to determine the consistency between predicted quality scores and subjective DMOS value: the linear correlation coefficient (LCC), the spearman rank order correlation coefficient (SROCC), and the root mean squared error (RMSE). The LCC and RMSE metrics served as indicators of a model's prediction accuracy, while the SROCC metric evaluates the prediction monotonicity of the model. When the LCC and SROCC values approach 1 or the RMSE values trend toward 0, it indicates a strong correlation between the model's predictions and human subjective evaluations, highlighting a high level of alignment with human perception.

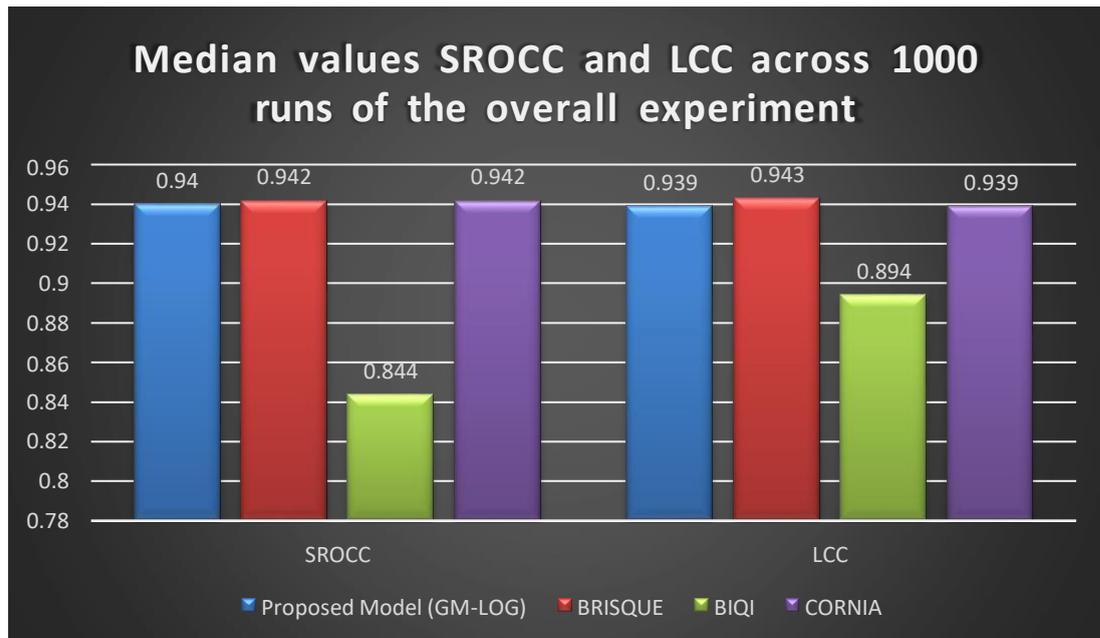


Figure 4.9: Median values for SROCC and LCC across 1000 runs of the overall experiment

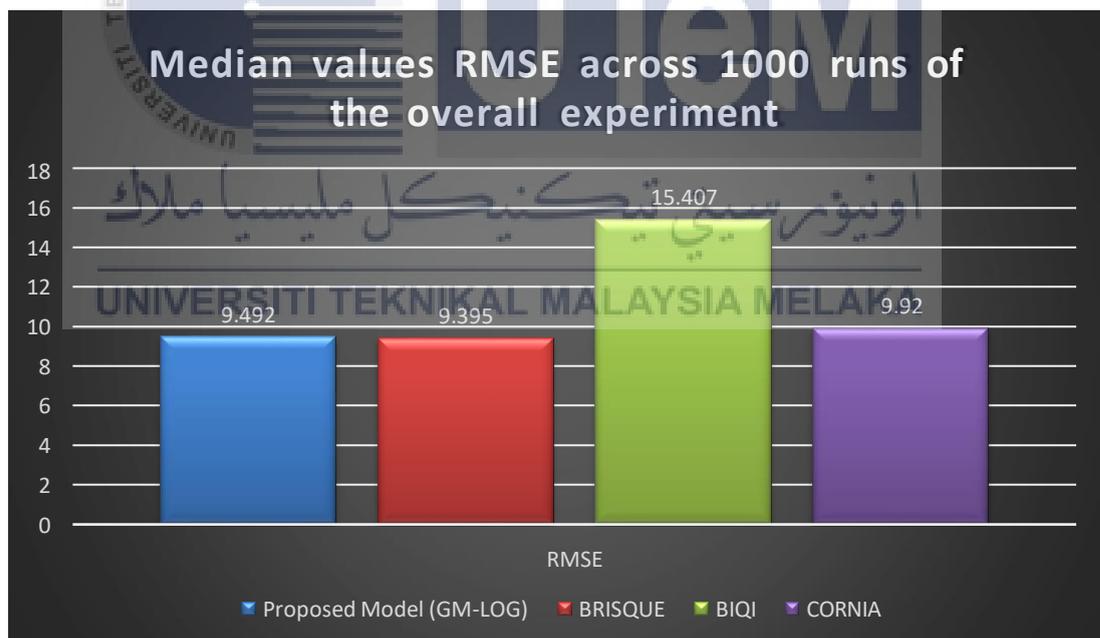


Figure 4.10: Median values for RMSE across 1000 runs of the overall experiment

Table 4.1: Median values across 1000 runs of the overall experiment

Algorithm	LIVE DATABASE		
	SROCC	LCC	RMSE
Proposed Model (GM-LOG)	0.940	0.939	9.492
BRISQUE	0.942	0.943	9.395
BIQI	0.844	0.894	15.407
CORNIA	0.942	0.939	9.920

The median outcomes from 1,000 runs for the comprehensive performance trial are displayed in Figure 4.9 and Figure 4.10 and summarized in Table 4.1. The top two BIQA models are highlighted in bold. Both the proposed model and BRISQUE rank among the leading models within the LIVE database. According to the table, the BIQI model produced an SROCC value of 0.844 and an LCC value of 0.894, suggesting a relatively weak correlation with human perceptual assessments.

In Figure 4.10, the median RMSE value for the proposed model is 9.492, the second lowest among all values depicted. A low RMSE signifies minimal variance between predicted and human-assigned scores, reflecting high accuracy in the model's performance. It's important to note that a lower RMSE, closer to 0, is preferred. The RMSE for the proposed model significantly outperforms BIQI, and CORNIA, underscoring that the suggested model exhibits superior estimation performance.

The model underwent additional validation by testing it on various individual distortions. Table 4.2 and Figure 4.11 display the median SROCC for each distortion. Remarkably, the model exhibited its strongest performance when tested on images affected by FF. Specifically, it achieved a median SROCC of 0.9006 for FF, highlighting its robust performance on these particular distortions.

Among the various distortions, different models showcased their strengths. CORNIA notably excelled with a top SROCC value of 0.952 for GB images. BRISQUE, on the other hand, demonstrated its prowess by producing the highest SROCC values of 0.964 for JP2K images and an impressive 0.979 for WN images. These results underscore the diverse capabilities of different models in addressing specific types of distortions.

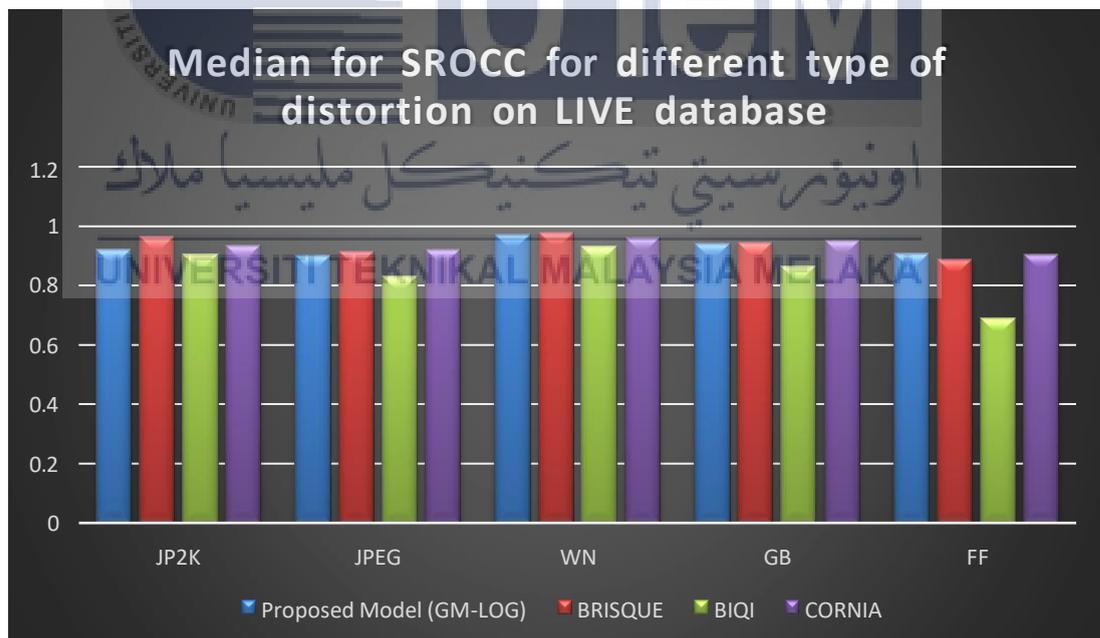


Figure 4.11: Median values for SROCC for different type of distortion on LIVE database

Table 4.2: Median for SROCC for different type of distortion on LIVE database

Algorithm	SROCC				
	JP2K	JPEG	WN	GB	FF
Proposed Model (GM-LOG)	0.9218	0.9003	0.9742	0.9479	0.9066
BRISQUE	0.964	0.916	0.979	0.945	0.887
BIQI	0.906	0.830	0.933	0.866	0.689
CORNIA	0.936	0.921	0.961	0.952	0.905

Table 4.3 and Figure 4.12 showcase the model's performance in specific distortion types, focusing on median LCC values. Among the examined BIQA models, BRISQUE emerges as the top performer across various distortions: achieving LCC values of 0.972 for JP2K, 0.993 for WN, and 0.949 for GB. Interestingly, the proposed model showcased its strength by attaining the highest LCC value of 0.9386 when tested on JPEG images, and value of 0.9417 on FF images.

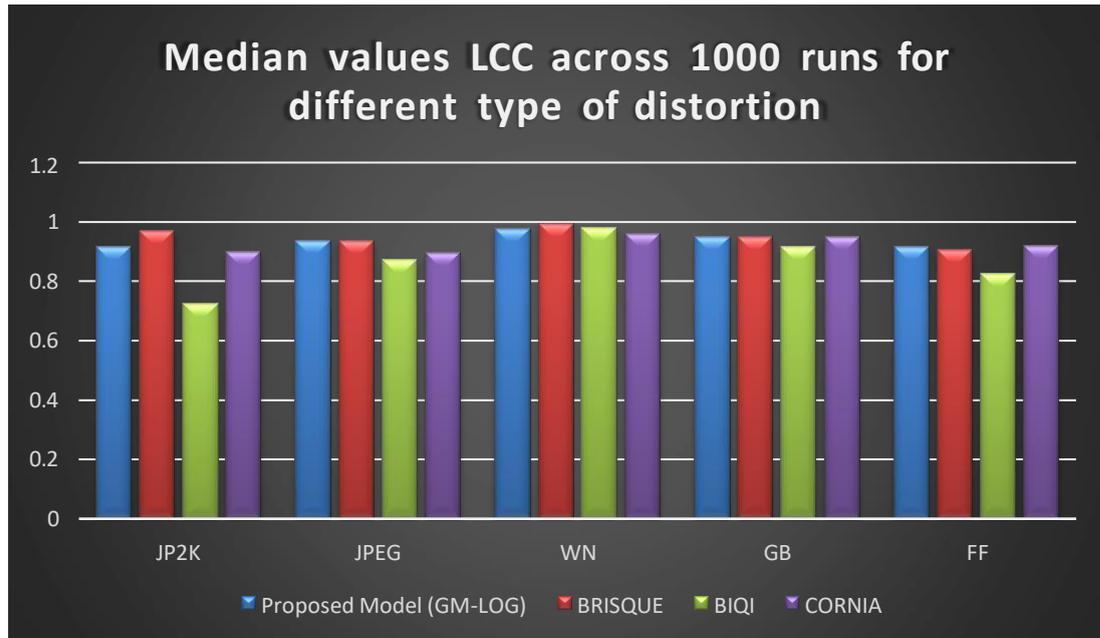


Figure 4.12: Median values for LCC across 1000 runs for different type of distortion

Table 4.3: Median LCC for different type of distortion in LIVE database

Algorithm	LCC				
	JP2K	JPEG	WN	GB	FF
Proposed Model (GM-LOG)	0.9176	0.9386	0.9763	0.9479	0.9417
BRISQUE	0.972	0.938	0.993	0.949	0.905
BIQI	0.724	0.874	0.981	0.917	0.827
CORNIA	0.899	0.894	0.960	0.948	0.920

In Table 4.4 and Figure 4.13, the model's performance within specific distortion types is depicted in terms of median RMSE values. Notably, among the five BIQA models examined, the proposed model demonstrated exceptional performance across

JPEG2K, JPEG, WN, GB, and FF cases. A smaller RMSE indicates a model's better fit to a dataset.

Additionally, it's evident that BRISQUE exhibited improvement compared to CORNIA showcased enhancements compared to BIQI, highlighting the iterative advancements and refinements made among these models within their respective distortion types.

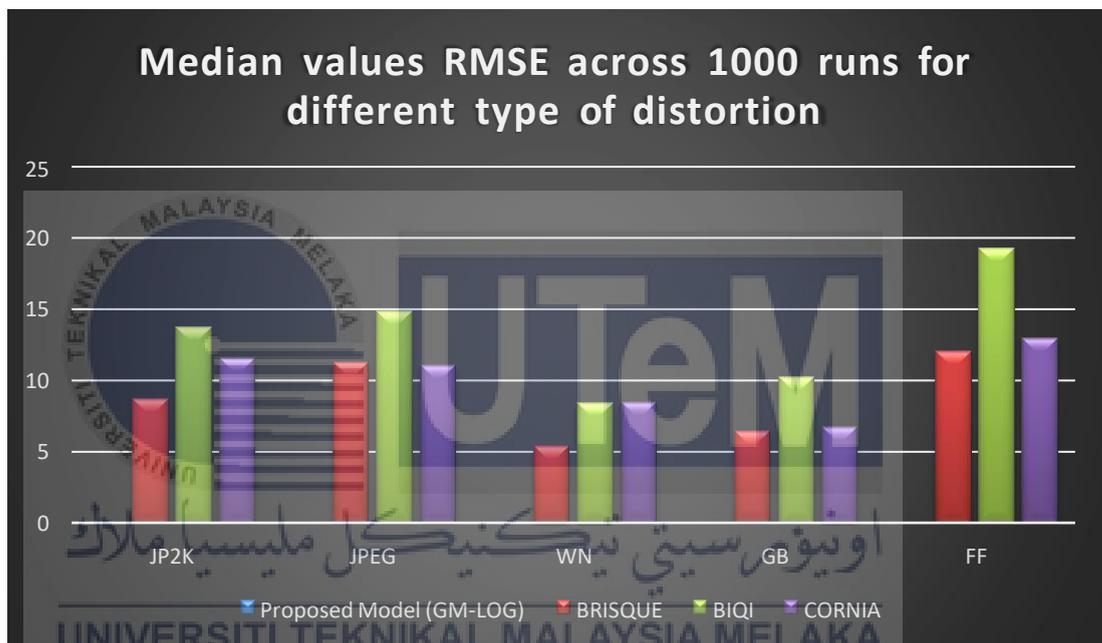


Figure 4.13: Median values for RMSE across 1000 runs for different type of distortion

Table 4.4: Median value RMSE for different type of distortion in LIVE database

Algorithm	RMSE				
	JP2K	JPEG	WN	GB	FF
Proposed Model (GM-LOG)	0.0254	0.0252	0.0697	0.0272	0.0311
BRISQUE	8.7204	11.3164	5.3847	6.4584	12.0683
BIQI	13.7552	14.8427	8.4094	10.2347	19.2911
CORNIA	11.5645	11.1013	8.4741	6.7449	12.9975

4.1.7 Computational Complexity

Another crucial aspect to assess in evaluating any BIQA model involves its computational speed. This entails analyzing the processing time needed to execute the proposed models. Specifically, comparing the average run-time of these models against other BIQA models under consideration is conducted using a standard image size of 512x768. These outcomes were derived from running unoptimized MATLAB R2023a code on a PC equipped with 8GB of RAM and powered by an AMD Ryzen 5 5500U with Radeon Graphics processor clocked at 2.10GHz.

Table 4.5: Average Run-Time for different available BIQA

BIQA MODEL	Proposed Model (GM-LOG)	BRISQUE	BIQI	CORNIA
RUN TIMES	0.9	0.10	0.05	2.43

Table 4.5 displays the comparison of average run times among various available BIQA models. It is evident that BIQI and the proposed model emerge as the top two in terms of speed. However, it is important to note that while BIQI exhibits higher speed, its predictive performance is significantly inferior to that of the proposed model.

4.2 Sustainable Design

This project aligns with the ninth of the United Nations' 17 Sustainable Development Goals, focusing on Industry, Innovation, and Infrastructure. It emphasizes the application of sustainable design principles aimed at establishing resilient infrastructure, fostering sustainable industrial practices, and driving innovation.

Utilizing machine learning for the classification of scientific and patent documents holds promise as a complementary approach to human-driven classifications. Neural networks, Support Vector Machines, Logistic Regression, and Supervised Fuzzy Algorithms have been frequently employed for the classification of scientific and technological literature. These algorithms stand out as some of the most utilized methods, contributing significantly to fostering innovation in this domain.

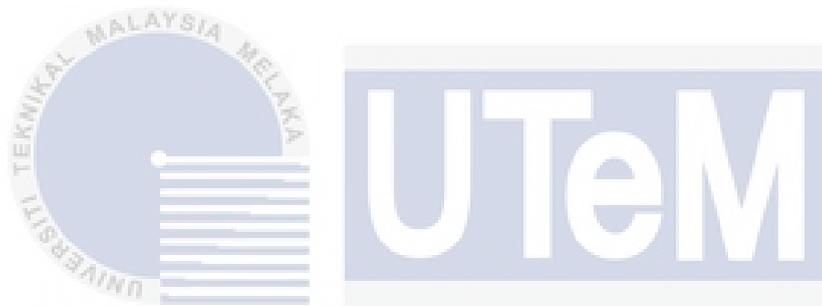


Figure 4.14: Sustainable development goals



CHAPTER 5

CONCLUSION AND FUTURE WORKS



5.1 Conclusion

To sum up, this project has successfully met all the outlined objectives. During the first semester, the creation of the LIVE database in MATLAB was accomplished. Image information was extracted across various distortions including JP2K, JPEG, White Noise, Gaussian Blur, and Fast Fading. In the second semester, all four project objectives were fulfilled: extracting pertinent quality-predictive spatial domain image features, implementing the MTL technique to concurrently train the model across different image distortion classes, weighting scores from diverse distortion classes via SVM/SVR to derive a final quality score. Subsequently, a comparative analysis was conducted between the model and several other BIQA models in terms of correlation with human perceptual measures.

In summary, this project introduces a straightforward, yet efficient BIQA model that incorporates a trace-norm regularized MTL technique within its learning framework. The proposed model leverages a shared representation across various distortion training samples, enabling the simultaneous learning of prediction models for each distortion class.

5.2 Recommendation

There are several recommendations to enhance the model's performance. First, exploring diverse features and datasets would validate and potentially improve the proposed model's efficacy. Second, investigating alternative MTL methods for expedited processing could be beneficial. Additionally, considering the use of the MTL approach itself to identify unknown distortion types, instead of relying solely on SVM as done in this project, could be advantageous. Lastly, comparing the performance of the proposed model with the latest BIQA models generated by different learning approaches like neural networks, deep learning, or nearest neighbor techniques would provide valuable insights.

REFERENCES

- [1] R. A. Manap, L. Shao and A. F. Frangi, "A non-parametric framework for no-reference image quality assessment," in Proc. of IEEE Conference on Signal and Information Processing, Orlando, Florida, Dec. 2015, pp. 562-566.
- [2] S. A. Golestaneh and D. M. Chandler, "No-reference quality assessment of JPEG images via a quality relevance map," IEEE Signal Processing Letters, vol. 21, no. 2, pp. 155-158, Feb. 2014.
- [3] E. Kurimo, L. Lepistö, J. Nikkanen, J. Grén, I. Kunttu, and J. Laaksonen, "The effect of motion blur and signal noise on image quality in low light imaging," Image Analysis, pp. 81-90, 2009.
- [4] G. Yammine, E. Wige, and A. Kaup, "A no-reference blocking artefacts visibility estimator in images," in Proc. of IEEE Conference on Image Processing, Hong Kong, Sept. 2010, pp. 2497-2500.
- [5] Singh, D. Singh, and M. Uddin, "Detection methods for blocking artefacts in transform coded images," IET Image Processing, vol. 8, no. 8, pp. 435-444, 2014.

- [6] ‘Live Image Quality Assessment Database," QUALINET Databases, 01-Jan-2020.[Online].Available:https://qualinet.github.io/database/image/live_image_qualityassessment_database/.
- [7] 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. 1 – 21, 2010.
- [8] S. T. Gandhe and O. S. "Blind Image Quality Evaluation of Stitched Image using Novel Hybrid Warping Technique." International Journal of Advanced Computer Science and Applications, vol. 10, no. 6, 2019
- [9] A.K. Moorthy and A. C. Bovik, "A two-step framework for constructing blind image quality indices," IEEE Signal Processing Letters, vol. 17, no. 5. pp. 513-516, 2010.
- [10] 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
- [11] M. A. Saad. A. C Bovik, and C. Charrier, "ADCT statistics-based blind image quality index," IEEE Signal Processing Letters, vol. 17, no. 6, pp. 583-586, 2010
- [12] V. Pang, M.Sun, X. Jiang, and X. Li, "Convolution in convolution for network in Network," IEEE Transactions on Neural Networks and Learning Systems, vol. 29, no. 5, pp. 1587-1597,2018.

- [13] Y. Fu and S. Wang. "A no reference image quality assessment metric based on visual perception." *Algorithms*, vol. 9, no. 4, p. 87, 2016.
- [14] W. Xue, X. Mou, L. Zhang, A. C. Bovik, and X. Feng. "Blind Image Quality Assessment using joint statistics of gradient magnitude and Laplacian features," *IEEE Transactions on Image Processing*, vol. 23, no 11, pp. 4850-4862, 2014.
- [15] W P. Cai, C. Fan, L. Zou, Y. Liu, Y. Ma, and M. Y. Wu, "Blind Image Quality Assessment based on classification guidance and feature aggregation." 2020
- [16] S. Wang, M. Kim, G. Wu, and D. Shen, "Scalable high performance image registration framework by unsupervised deep feature representations learning." *Deep Learning for Medical Image Analysis*, pp. 245-269, 2017
- [17] K. Ma, W. Liu, K. Zhang, Z. Duanmu, Z. Wang, and W. Zun, "End-to-End Blind Image Quality Assessment Using Deep Neural Networks." *IEEE Transactions on Image Processing*, vol. 27, no. 3, pp. 1202-1213, 2018,
- [18] J. Xu, P. Ye, Q. Li, H. Du, Y. Liu, and D. Doermann, "Blind Image Quality Assessment based on high order statistics aggregation," *IEEE Transactions on Image Processing*, vol. 25, no. 9, pp. 4444-4457, 2016.
- [19] S. Alireza Golestanch and Kris Kitani. "No-Reference Image Quality Assessment via Feature Fusion and arxiv.org/pdf/2006.03783,2020. Multi-Task Learning."

- [20] A. Mittal, A. K. Moorthy and A. C. Bovik, "Blind/Referenceless Image Spatial Quality Evaluator," *2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)*, Pacific Grove, CA, USA, 2011, pp. 723-727, doi: 10.1109/ACSSC.2011.6190099.
- [21] P. Ye, J. Kumar, L. Kang and D. Doermann, "Unsupervised feature learning framework for no-reference image quality assessment," *2012 IEEE Conference on Computer Vision and Pattern Recognition*, Providence, RI, USA, 2012, pp. 1098-1105, doi: 10.1109/CVPR.2012.6247789.



APPENDICES

APPENDIX A: Coding for Creating Database of The Image

```

load('dmos.mat');
FYPdatabase=cell(numel(dmos),4);
FYPdatabase(:,2)=num2cell(dmos);
feat=zeros(982,40);
for k=1:982
    A=FYPdatabase{k,1};
    featim=gmlog(A);
    FYPdatabase{k,5}=featim;
    feat(k,:)=featim
end
addpath('C:\Users\User\Desktop\LIVE database\dataserelease2\jp2k');
for i=1:227
    A=imread(sprintf('img%d.bmp',i));
    A1=rgb2gray(A);
    A2=double(A1);
    FYPdatabase{i,1}=A2;
    FYPdatabase{i,3}=1;
end

addpath('C:\Users\User\Desktop\LIVE database\dataserelease2\jpeg');
for i=1:233
    A=imread(sprintf('img%d.bmp',i));
    A1=rgb2gray(A);
    A2=double(A1);
    FYPdatabase{227+i,1}=A2;
    FYPdatabase{227+i,3}=2;
end

addpath('C:\Users\User\Desktop\LIVE database\dataserelease2\wn');
for i=1:174
    A=imread(sprintf('img%d.bmp',i));
    A1=rgb2gray(A);
    A2=double(A1);
    FYPdatabase{460+i,1}=A2;
    FYPdatabase{460+i,3}=3;

```

end

```
addpath('C:\Users\User\Desktop\LIVE database\dataserelease2\gblur');
```

```
for i=1:174
```

```
    A=imread(sprintf('img%d.bmp',i));
```

```
    A1=rgb2gray(A);
```

```
    A2=double(A1);
```

```
    FYPdatabase{634+i,1}=A2;
```

```
    FYPdatabase{634+i,3}=4;
```

end

```
addpath('C:\Users\User\Desktop\LIVE database\dataserelease2\fastfading');
```

```
for i=1:174
```

```
    A=imread(sprintf('img%d.bmp',i));
```

```
    A1=rgb2gray(A);
```

```
    A2=double(A1);
```

```
    FYPdatabase{808+i,1}=A2;
```

```
    FYPdatabase{808+i,3}=5;
```

end



APPENDIX B: Coding to call images from database

```
w=imread('C:\Users\User\Desktop\PSM 1\LIVE  
database\databaserelease2\jp2k\img10.bmp');  
A=rgb2gray(w);  
B=double(A);  
imshow(A);
```



APPENDIX C: Coding for Feature Extraction

```
feat=zeros(982,40);  
for k=1:982  
    A=FYPdatabase{k,1};  
    featim=gmlog(A);  
    FYPdatabase{k,5}=featim;  
    feat(k,:)=featim  
end
```



APPENDIX D: Coding for MTL Train-Test Model

Performance

```

load('FYPdatabase (2).mat');

% Loading SVM library
libsvmpath='C:\Users\User\Desktop\PSM1\MALSAR\libsvm-mat-3.0-1';
addpath(libsvmpath);

%remove reference image in database
idx = cellfun(@(x) ~x, FYPdatabase(:,2));
FYPdatabase(idx, :) = [];

% Loading MALSAR library
addpath('MALSAR');
addpath('MALSAR/utills');

% Loading train-test partition data
load('run280415.mat');
%% MTLBIQG model
B=cell2mat(FYPdatabase(:,4));
N = 1000;
SROCC_ALL=zeros(N,1);
LCC_ALL = zeros(N,1);
RMSE_ALL=zeros(N,1);
SROCC_JP2K=zeros(N,1);
LCC_JP2K=zeros(N,1);
RMSE_JP2K=zeros(N,1);
SROCC_JPEG=zeros(N,1);
LCC_JPEG=zeros(N,1);
RMSE_JPEG=zeros(N,1);
SROCC_WN=zeros(N,1);
LCC_WN=zeros(N,1);
RMSE_WN=zeros(N,1);
SROCC_GB=zeros(N,1);
LCC_GB=zeros(N,1);
RMSE_GB=zeros(N,1);
SROCC_FF=zeros(N,1);
LCC_FF=zeros(N,1);
RMSE_FF=zeros(N,1);

for i=1:N
    % Train-test partition
    train=ismember(B,A(i,:));
    test=~train;
    Trainset=FYPdatabase(train,:);
    Testset=FYPdatabase(test,:);

```

```

%% DI training (SVM)
% Training
feattrain=cell2mat(Trainset(:,5));
labeltrain=cell2mat(Trainset(:,3));
model1 = fitcecoc(feattrain,labeltrain,'Coding','onevsone');

%% QE training (Trace-norm MTL)
Xtr=cell(1,5);
Ytr=cell(1,5);
Xte=cell(1,5);
Yte=cell(1,5);
% Splitting training and test sets for each task
% Splitting database info training and testing set for each task
Ctr=cell2mat(Trainset(:,3));
Cte=cell2mat(Testset(:,3));
% Task 1: JP2K
idxtr=ismember(Ctr,1);
idxte=ismember(Cte,1);
Xtr{1,1}=cell2mat(Trainset(idxtr,5));
Ytr{1,1}=cell2mat(Trainset(idxtr,2));
Xte{1,1}=cell2mat(Testset(idxte,5));
Yte{1,1}=cell2mat(Testset(idxte,2));
clear idxtr idxte;
% Task 2: JPEG
idxtr=ismember(Ctr,2);
idxte=ismember(Cte,2);
Xtr{1,2}=cell2mat(Trainset(idxtr,5));
Ytr{1,2}=cell2mat(Trainset(idxtr,2));
Xte{1,2}=cell2mat(Testset(idxte,5));
Yte{1,2}=cell2mat(Testset(idxte,2));
clear idxtr idxte;
% Task 3: WN
idxtr=ismember(Ctr,3);
idxte=ismember(Cte,3);
Xtr{1,3}=cell2mat(Trainset(idxtr,5));
Ytr{1,3}=cell2mat(Trainset(idxtr,2));
Xte{1,3}=cell2mat(Testset(idxte,5));
Yte{1,3}=cell2mat(Testset(idxte,2));
clear idxtr idxte;
% Task 4: GB
idxtr=ismember(Ctr,4);
idxte=ismember(Cte,4);
Xtr{1,4}=cell2mat(Trainset(idxtr,5));
Ytr{1,4}=cell2mat(Trainset(idxtr,2));
Xte{1,4}=cell2mat(Testset(idxte,5));
Yte{1,4}=cell2mat(Testset(idxte,2));
clear idxtr idxte;
% Task 5: FF
idxtr=ismember(Ctr,5);

```

```

idxte=ismember(Cte,5);
Xtr{1,5}=cell2mat(Trainset(idxtr,5));
Ytr{1,5}=cell2mat(Trainset(idxtr,2));
Xte{1,5}=cell2mat(Testset(idxte,5));
Yte{1,5}=cell2mat(Testset(idxte,2));
clear idxtr idxte;
clear Ctr Cte;

% Adding the bias term for each task to learn the bias
for t=1:size(Xtr,2)
    Xtr{t}=[Xtr{t} ones(size(Xtr{t},1),1)];
end
clear t;
for t=1:size(Xte,2)
    Xte{t}=[Xte{t} ones(size(Xte{t},1),1)];
end
% Estimating the best regularization parameter
% Function used for evaluation
eval_func_str='eval_MTL_mse';
higher_better=false;
% Cross validation fold
cv_fold=5;
% Optimization options
opts=[];
opts.maxIter=100;
% Model parameter range
param_range=[0.001 0.01 0.1 1 10 100 1000 10000];
% Cross validation
best_param1=CrossValidation1Param(Xtr, Ytr, 'Least_Trace',opts,param_range,...
cv_fold, eval_func_str, higher_better);
% Model selection
W=Least_Trace(Xte,Yte,best_param1,opts);

%% Testing stage (DS Experiment)
% JP2K
Score_JP2K=(cell2mat(Xte(:,1)))*W(:,1);
% JPEG
Score_JPEG=(cell2mat(Xte(:,2)))*W(:,2);
% WN
Score_WN=(cell2mat(Xte(:,3)))*W(:,3);
% GB
Score_GB=(cell2mat(Xte(:,4)))*W(:,4);
% FF
Score_FF=(cell2mat(Xte(:,5)))*W(:,5);
% Combine all together
Scoreall = [Score_JP2K;Score_JPEG;Score_WN;Score_GB;Score_FF];
%% Testing stage (Overall Experiment)
% DI
feattest=cell2mat(Testset(:,5));
predict_label=predict(model1,feattest);

```

```

% QE
feattest1=[feattest ones(size(feattest,1),1)];
PreScore=zeros(size(feattest1,1),1);
for i1=1:size(feattest1,1)
% Predicted score
    if predict_label(i1)==1
        score=feattest1(i1,:)*W(:,1);
    elseif predict_label(i1)==2
        score=feattest1(i1,:)*W(:,2);
    elseif predict_label(i1)==3
        score=feattest1(i1,:)*W(:,3);
    elseif predict_label(i1)==4
        score=feattest1(i1,:)*W(:,4);
    else
        score=feattest1(i1,:)*W(:,5);
    end
    PreScore(i1,1)=score;
    clear score;
end
%JP2K
ActScore1=cell2mat(Yte(:,1));
SROCC_JP2K(i,1)=corr(ActScore1,Score_JP2K,'type','Spearman');
LCC_JP2K(i,1)=corr(ActScore1,Score_JP2K,'type','Pearson');
RMSE_JP2K(i,1)=sqrt(sum(ActScore1(:)-
Score_JP2K(:)).^2/numel(ActScore1));
% JPEG
ActScore2=cell2mat(Yte(:,2));
SROCC_JPEG(i,1)=corr(ActScore2,Score_JPEG,'type','Spearman');
LCC_JPEG(i,1)=corr(ActScore2,Score_JPEG,'type','Pearson');
RMSE_JPEG(i,1)=sqrt(sum(ActScore2(:)-
Score_JPEG(:)).^2/numel(ActScore2));
% WN
ActScore3=cell2mat(Yte(:,3));
SROCC_WN(i,1)=corr(ActScore3,Score_WN,'type','Spearman');
LCC_WN(i,1)=corr(ActScore3,Score_WN,'type','Pearson');
RMSE_WN(i,1)=sqrt(sum(ActScore3(:)-Score_WN(:)).^2/numel(ActScore3));
% GB
ActScore4=cell2mat(Yte(:,4));
SROCC_GB(i,1)=corr(ActScore4,Score_GB,'type','Spearman');
LCC_GB(i,1)=corr(ActScore4,Score_GB,'type','Pearson');
RMSE_GB(i,1)=sqrt(sum(ActScore4(:)-Score_GB(:)).^2/numel(ActScore4));
% FF
ActScore5=cell2mat(Yte(:,5));
SROCC_FF(i,1)=corr(ActScore5,Score_FF,'type','Spearman');
LCC_FF(i,1)=corr(ActScore5,Score_FF,'type','Pearson');
RMSE_FF(i,1)=sqrt(sum(ActScore5(:)-Score_FF(:)).^2/numel(ActScore5));
% Overall
ActScore=cell2mat(Testset(:,2));
SROCC_ALL(i,1)=corr(ActScore,PreScore,'type','Spearman');
LCC_ALL(i,1)=corr(ActScore,PreScore,'type','Pearson');

```

```

RMSE_ALL(i,1)=sqrt(sum(ActScore(:)-PreScore(:)).^2/numel(ActScore));
% SROCC_ALL1(i,1)=corr(ActScore,Scoreall,'type','Spearman');
% LCC_ALL1(i,1)=corr(ActScore,Scoreall,'type','Pearson');
% RMSE_ALL1(i,1)=sqrt(sum(ActScore(:)-Scoreall(:)).^2/numel(ActScore));
clear i1 feattest feattest1;
clear modell1;
end
median_LCC_JP2K=median(LCC_JP2K)
median_SROCC_JP2K=median(SROCC_JP2K)
median_RMSE_JP2K=median(RMSE_JP2K)

median_LCC_JPEG=median(LCC_JPEG)
median_SROCC_JPEG=median(SROCC_JPEG)
median_RMSE_JPEG=median(RMSE_JPEG)

median_LCC_WN=median(LCC_WN)
median_SROCC_WN=median(SROCC_WN)
median_RMSE_WN=median(RMSE_WN)

median_LCC_GB=median(LCC_GB)
median_SROCC_GB=median(SROCC_GB)
median_RMSE_GB=median(RMSE_GB)

median_LCC_FF=median(LCC_FF)
median_SROCC_FF=median(SROCC_FF)
median_RMSE_FF=median(RMSE_FF)

median_LCC_ALL=median(LCC_ALL)
median_SROCC_ALL=median(SROCC_ALL)
median_RMSE_ALL=median(RMSE_ALL)

```