COPPE
UFRJ
Instituto Alberto Luiz Coimbra de
Pós-Graduação e Pesquisa de Engenharia

MACHINE LEARNING AND DEEP LEARNING MODELS FOR TONE-MAPPED IMAGE QUALITY ASSESSMENT

Gustavo Martins da Silva Nunes

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Elétrica.

Orientadores: José Gabriel Rodríguez Carneiro Gomes
Fernanda Duarte Vilela Reis de Oliveira
Mylène Christine Queiroz Farias

Rio de Janeiro
Julho de 2022

MACHINE LEARNING AND DEEP LEARNING MODELS FOR TONE-MAPPED IMAGE QUALITY ASSESSMENT

Gustavo Martins da Silva Nunes

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Orientadores: José Gabriel Rodríguez Carneiro Gomes
Fernanda Duarte Vilela Reis de Oliveira
Mylène Christine Queiroz Farias

Aprovada por: Prof. José Gabriel Rodríguez Carneiro Gomes
Prof$^{\underline{a}}$. Fernanda Duarte Vilela Reis de Oliveira
Prof. Carlos Alexandre Barros de Mello
Prof. Eduardo Antônio Barros da Silva
Prof. Frederic Dufaux

RIO DE JANEIRO, RJ – BRASIL
JULHO DE 2022

Nunes, Gustavo Martins da Silva

Machine Learning and Deep Learning Models for Tone-Mapped Image Quality Assessment/Gustavo Martins da Silva Nunes. – Rio de Janeiro: UFRJ/COPPE, 2022.

XXIII, 102 p.: il.; $29,7$cm.

Orientadores: José Gabriel Rodríguez Carneiro Gomes
Fernanda Duarte Vilela Reis de Oliveira
Mylène Christine Queiroz Farias

Tese (doutorado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2022.

Referências Bibliográficas: p. 86 – 97.

1. Tone Mapping. 2. Deep Learning. 3. Machine Learning. 4. Image Quality Assessment. I. Gomes, José Gabriel Rodríguez Carneiro *et al.* II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

Agradecimentos

Este trabalho é fruto de uma caminhada que se iniciou não no começo do doutorado, há 4 anos, mas sim há 12 anos, quando pisei pela primeira vez no auditório do bloco A da Escola Politécnica na UFRJ, realizando meu sonho de estudar nesta universidade. Portanto, nesta seção, agradeço a todos que fizeram parte dessa caminhada e foram importantes para mim.

Gostaria de agradecer à minha mãe, Ana Lúcia, ao meu pai, Nelson, e à minha irmã, Luciana, por todo o carinho e apoio que sempre me deram durante toda a minha vida. Eu tenho muita sorte de ter pais tão incríveis e que são capazes de fazer o possível e o impossível para dar as melhores condições para mim e minha irmã. Tenho sorte, também, de ter uma irmã a quem admiro tanto e que sempre me motiva a buscar o melhor.

Agradeço também ao meu orientador José Gabriel Gomes, com quem tive o prazer de trabalhar desde a graduação, passando pelo mestrado, e, agora, terminando com este doutorado, e a quem tenho como inspiração e referência na profissão. Muito obrigado por toda a ajuda e todos os conselhos ao longo de todos esses anos. Sem eles, certamente não teria chegado ao ponto que estou agora. Agradeço à minha orientadora Mylène Farias, que conheci no mestrado e que, mesmo com a correria causada por uma agenda lotada de reuniões, sempre conseguiu me ajudar de forma decisiva e precisa ao longo da pesquisa.

Um agradecimento especial à minha orientadora e querida amiga Fernanda Duarte. Lembro que a conheci ainda no mestrado, quando passei a frequentar diariamente o laboratório, já que meu computador de casa havia parado de funcionar. Hoje, agradeço muito pelo meu computador ter quebrado naquela época, pois isso me possibilitou iniciar e construir uma amizade, que levarei para o resto da vida, com uma pessoa que só traz coisas positivas. Saiba que a admiro muito profissionalmente e pessoalmente. Muito obrigado pela ajuda ao longo do trabalho, desde sugestões de experimentos e discussão de resultados até correções no texto final. E, também, por estar lá quando precisei.

Agradeço a todos os professores que contribuíram para a minha formação profissional e pessoal, em especial aos professores Luiz Wagner Biscainho, Carlos Teodósio, Eduardo da Silva, e à professora Mariane Petraglia. Agradeço, também, aos pro-

fessores que fizeram parte da banca avaliadora deste trabalho: Carlos Alexandre Barros, Eduardo da Silva e Frederic Dufaux. Obrigado por reservarem o tempo de vocês para ler o texto e tecer comentários que contribuíram para enriquecer este trabalho.

Agradeço a todos os colegas de laboratório, em especial àqueles com quem convivi de forma mais próxima: Fabián Mederos, Gabriela Dantas, Germano Fonseca, Leonardo Mazza, Odair Xavier, Olavo Sampaio, Pedro Cayres e Tiago Monnerat. Agradeço, também, aos amigos que fiz dentro da UFRJ durante essa jornada que agora se encerra: Carlos Pedro Lordelo, Felipe Gonzalez, Felipe Petraglia, Gabriel Torres, Henrique Dias, João Victor da Fonseca, Júlio Vargas, Leonardo Oliveira, Lucas Tavares, Michel Morais, Pedro Bandeira e Pedro Fonini.

Por fim, agradeço ao apoio financeiro concedido a este trabalho pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), através dos projetos 141738/2018-9 e 440074/2020-7, e pela Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro (FAPERJ), através do projeto E-26/200.699/2021 (pedido 262387). Agradeço, também, à Amazon por disponibilizar seus serviços de computação na nuvem e conceder créditos de uso, contemplados no projeto 440074/2020-7 do CNPq. Estes apoios foram fundamentais para a realização do presente trabalho.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

MODELOS DE APRENDIZADO DE MÁQUINA E MODELOS NEURAIS PROFUNDOS PARA AVALIAÇÃO DE QUALIDADE PARA IMAGENS DE TONS MAPEADOS

Gustavo Martins da Silva Nunes

Julho/2022

Orientadores: José Gabriel Rodríguez Carneiro Gomes
Fernanda Duarte Vilela Reis de Oliveira
Mylène Christine Queiroz Farias

Programa: Engenharia Elétrica

Imagens em alta faixa dinâmica precisam ser processadas por operadores de *tone mapping* (TMOs) para serem mostradas adequadamente em *displays* convencionais. Há duas pequenas bases de dados para a tarefa de avaliação da qualidade de imagens *tone-mapped* (AQITM), chamadas ESPL-LIVE e TMID. Apresentamos, neste trabalho, uma nova base de dados para AQITM, chamada PBTDB. Ela contém cerca de 175000 amostras, cada uma rotulada por quatro métricas objetivas para AQITM. Conduzimos, também, testes subjetivos para avaliar a qualidade de 3009 amostras desta base. Métricas não-referenciadas (NR) para AQITM de estado-da-arte, que são o foco deste estudo, são incapazes de avaliar de forma confiável a qualidade de imagens *tone-mapped* que não pertencem a bases de dados específicas. Investigamos duas abordagens para obter novas métricas NR de AQITM mais gerais. Na primeira, usamos amostras da base ESPL-LIVE para treinar modelos de regressão que combinam notas de diversas métricas de AQITM em uma única nota. As notas do melhor modelo apresentam baixa correlação com as notas subjetivas da base de teste TMID (PLCC de 0,65, e SRCC de 0,55). Na segunda, treinamos modelos de aprendizado profundo com amostras da base PBTDB, e usamos as bases ESPL-LIVE e TMID para testá-los. Alguns modelos alcançam desempenho moderado na base TMID, mas todos apresentam baixo desempenho na base ESPL-LIVE (melhores desempenhos: PLCC de 0,48 e SRCC de 0,43 na base ESPL-LIVE, e PLCC de 0,79 e SRCC de 0,72 na base TMID). O tipo da nota de qualidade (subjetivo ou objetivo) usada para treinar os modelos influencia fortemente nesses desempenhos.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

MACHINE LEARNING AND DEEP LEARNING MODELS FOR TONE-MAPPED IMAGE QUALITY ASSESSMENT

Gustavo Martins da Silva Nunes

July/2022

Advisors: José Gabriel Rodríguez Carneiro Gomes
Fernanda Duarte Vilela Reis de Oliveira
Mylène Christine Queiroz Farias

Department: Electrical Engineering

High dynamic range (HDR) images are increasingly more common in today consumer applications. Such images need to be processed by so-called tone-mapping operators (TMOs), in order to be properly exhibited on standard display devices. Objective tone-mapped image quality assessment (TMIQA) metrics are desirable to aid selecting the best TMO for an HDR image. Only few relatively small benchmark databases exist for TMIQA task, namely ESPL-LIVE and TMID. We present a new database for TMIQA, called PBTDB, which contains approximately 175000 samples. Sample quality is labeled by four TMIQA metrics. We also conduct subjective experiments to assess quality of 3009 PBTDB samples. Current state-of-the-art no-reference (NR) IQA metrics, which are the focus of our study, are unable to reliably assess the quality of tone-mapped images that do not come from specific databases. We design cross-dataset experiments to investigate two approaches for developing new generic NR TMIQA metrics. In the first one, we use ESPL-LIVE samples to train machine-learning based regression models that combine scores from multiple IQA metrics into a single quality score. The best performing model outputs quality scores that correlate relatively poorly with TMID sample mean opinion scores (PLCC of 0.65 and SRCC of 0.55). In the second one, we use PBTDB samples to train deep learning architectures, and ESPL-LIVE and TMID samples to test them. Models can achieve moderate performance in TMID, but all perform poorly in ESPL-LIVE (best performances: PLCC of 0.48 and SRCC of 0.43 in ESPL-LIVE, and PLCC of 0.79 and SRCC of 0.72 in TMID). Performance is heavily influenced by quality score types (subjective versus objective) used as targets to train the models.

Contents

List of Figures

List of Tables

List of Abbreviations

AMT	Amazon Mechanical Turk
AWS	Amazon Web Services
BIQI	Blind Image Quality Index
BLIINDS-II	BLind Image Integrity Notator using DCT Statistics
BLIQUE-TMI	BLInd QUality Evaluator for Tone-Mapped Images
BIO-BLIND	BIOlogically inspired BLIND Quality Assessment
BRISQUE	Blind/Referenceless Image Spatial Quality Evaluator
BTMQI	Blind Tone-Mapped Quality Index
C-DIIVINE	Complex Distortion Identification-based Image Verity and Integrity Evaluation
CNN	Convolutional Neural Network
CNS	Colorfulness, Naturalness and Structure
CORNIA	COdebook Representation for No-Reference Image Assessment
DCT	Discrete Cosine Transform
DESIQUE	DErivative Statistics-based QUality Evaluator
DIIVINE	Distortion Identification-based Image Verity and Integrity Evaluation
DTMO	Digital Tone-Mapping Operator
ENIQA	Entropy-based No-reference Image Quality Assessment
ESPL-LIVE	Embedded Signal Processing Laboratory - Laboratory for Image & Video Engineering
FC	Fully Connected
FR	Full-Reference
FRIQUEE	Feature maps-based Referenceless Image Quality Evaluation Engine
FPTMO	Focal-Plane Tone-Mapping Operator
GM-LOG	Gradient Magnitude Laplacian Of Gaussian metric
GoF	Goodness-of-Fit
GTB	Gradient Tree Boosting
GSD	Generalized Score Distribution
GSQ	Gold Standard Question

HDR	High Dynamic Range
HIGRADE	HDR Image GRADient based Evaluator
ILNIQE	Integrated Local Natural Image Quality Evaluator
INSLA	Iterated Nested Least-Squares Algorithm
IQA	Image Quality Assessment
KNN	K-Nearest Neighbor
MEF	Multi-Exposure Fusion
MOS	Mean Opinion Score
NFERM	No-reference Free Energy-Based Robust Metric
NIMA	Neural Image Assessment
NIQE	Natural Image Quality Evaluator
NIQMC	No-reference Image Quality Metric for Contrast distortion
NR	No-Reference
PAM	Personalized Aesthetic Model
PBTDB	Patch-Based Tone-mapped image DataBase
PCA	Principal Component Analysis
PLCC	Pearson Linear Correlation Coefficient
PP	Post-Processing
QoE	Quality of Experience
ReLU	Rectified Linear Unit
RMSE	Root-Mean-Squared Error
RR	Reduced-Reference
SRCC	Spearman Rank Correlation Coefficient
SFS	Sequential Forward Selection
SSEQ	Spatial-Spectral Entropy-based Quality index
SVM	Support Vector Machine
SVR	Support Vector Regression
TMIQA	Tone-Mapped Image Quality Assessment
TMO	Tone-Mapping Operator
TMQI	Tone-Mapped image Quality Index
TMID	Tone-Mapped Image Dataset

Chapter 1

Introduction

Real-world scenes may simultaneously contain areas that are very bright and areas that are very dark. These scenes contain luminance values that can span a range of up to eight orders of magnitude. Their content has a large number of details. Such scenes are commonly referred to as High Dynamic Range[1] (HDR) scenes. The human eye has complex light adaptation mechanisms that allow us to discern fine details in bright areas and dark areas, simultaneously. Exhibiting HDR scenes in conventional display media is challenging. These devices are unable to represent all the luminance values from HDR scenes. Typically, they use eight bits to quantize luminance values as pixel values. With this number of bits, pixel values can vary in approximately two orders of magnitude, at most. Therefore, they are not sufficient to represent all information that is present in the HDR scene [1]. This limitation inevitably leads to loss of details in the displayed scene, as compared to the original scene.

Tone mapping refers to a set of techniques that are designed to solve this problem. These techniques apply specific functions that map input HDR pixel values into lower dynamic ranges, aiming at mitigating the content loss in the displayed scene. Algorithms that perform this operation are called Tone-Mapping Operators (TMOs), and images resulting from this operation are called tone-mapped images. Figure 1.1 illustrates the importance of TMOs for displaying HDR images in conventional display devices.

A large number of TMOs have been proposed in the literature [2]. TMOs are developed to achieve particular goals [3], such as generating aesthetically pleasing images, enhancing visibility of details in some scene regions, or even recreating the exact same visual experience a user would have when observing the real scene. In the last case, the TMO has the goal of simulating visual effects like glare and chromatic adaptation. Depending on the TMO purpose, mapping functions may

[1] Dynamic range is generally defined as the ratio between the maximum and minimum values defining an interval.

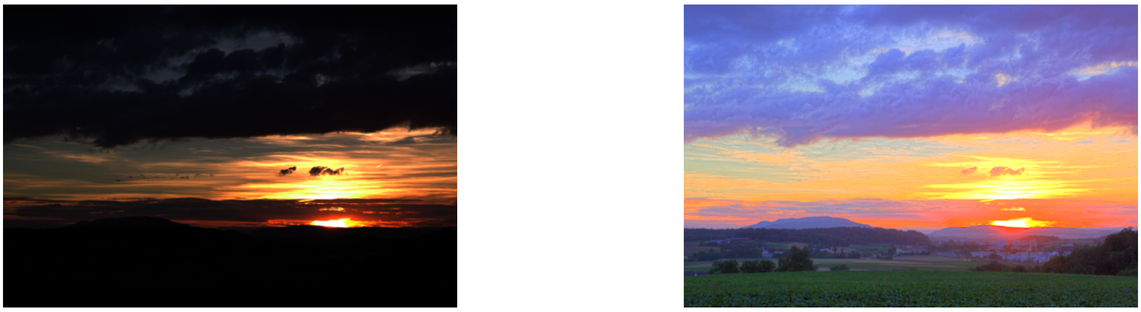
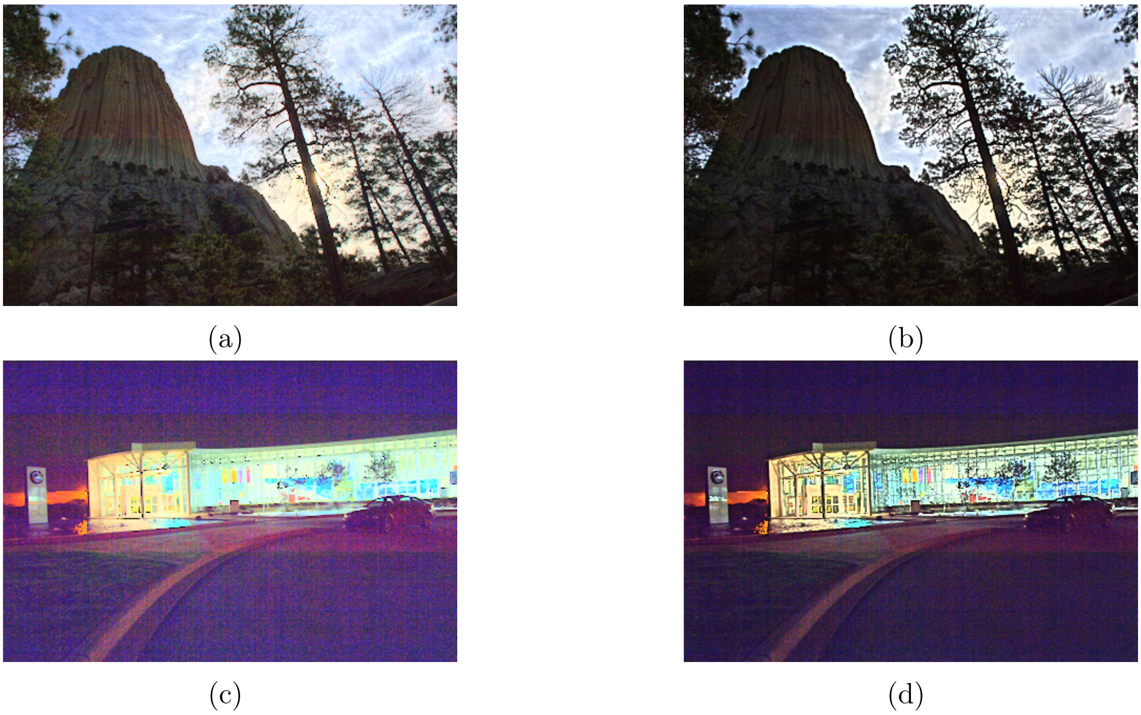


Figure 1.1: Original HDR image displayed without prior application of tone mapping (left), and resulting tone-mapped image (right). Many scene details are lost if no tone mapping is performed before displaying the HDR scene in a conventional display device.

simply clamp pixel values that lie outside a predefined range [4, 5], or apply nonlinear transformations that emulate complex aspects of human vision [6, 7]. Tone-mapped image quality assessment (TMIQA) is then an important task specially for applications that aim at providing the best user experience. TMIQA can aid such applications in selecting, for a given HDR scene, the tone-mapped image with "best" quality.

Many works compare digital TMOs using other different performance metrics [3, 8], as there are no universal criteria that establish what constitutes a "good" tone mapping, nor standard frameworks defining how TMOs should be evaluated. No TMO is acknowledged as being the "best one" for all HDR scenes. Furthermore, TMOs usually have several parameters [2] that must be tuned for each HDR image. Such parameters change the tone-mapping curve and, thus, impact directly on the quality of generated tone-mapped images. Figure 1.2 illustrates four tone-mapped images generated by the same TMO [9], with different parameter values. In this figure, the top and bottom rows, respectively, depict tone-mapped versions of the "Hill" and "BMW" HDR scenes. Figure 1.2(a) shows the tone-mapped image obtained after tuning the TMO parameters with the "best" values for the "Hill" scene. Figure 1.2(b) shows the same HDR scene tone-mapped with the same TMO, but with parameters adjusted for the "BMW" scene. In this case, haloing artifacts are observed (*i.e.* the white contour along the hill borders), and leaves are less visible in the scene bottom right part. In Figure 1.2(c), the "BMW" scene is tone-mapped with the TMO parameters calibrated to the "Hill" scene. In this case, details inside the building are lost. A tone-mapped version of the "BMW" scene, using TMO parameters tuned for this scene, is shown in Figure 1.2(d). In this case, more details are visible. TMIQA is an interesting task because it can help to select high quality tone-mapped versions of an HDR scene. TMIQA is also challenging because the definition of "best" quality – or even "good" tone mapping – is not clear.

There are relatively few TMIQA metrics available in the literature as compared



(a) (b) (c) (d)

Figure 1.2: Tone-mapped versions of the “Hill” (top row), and “BMW” (bottom row) HDR scenes. For both scenes, the same TMO [13] is used, with different parameter values. In (a), TMO parameters are actually calibrated for the “Hill” scene. In (b), the “Hill” scene is tone-mapped with TMO parameters adjusted for the “BMW” scene. In (c), the “BMW” scene is tone-mapped with TMO parameters tuned for the “Hill” scene. Finally, in (d), the tone-mapped version of “BMW” scene is obtained with TMO parameters calibrated for the “BMW” scene.

to the number of objective image quality assessment (IQA) metrics that are not dedicated to evaluate tone-mapping quality [10]. This leaves the TMIQA field open for new research ideas and improvement. Objective TMIQA metrics available in the literature extract features by performing explicit computations on pixel values in some domain (*e.g.* space or frequency domain). Traditionally, these features are combined into a single score representing the image quality using either one of the two following approaches. The first approach consists in explicitly weighing the relative contribution of each feature to the overall quality. The second approach uses regression models trained with conventional machine learning methods that map the image representation in feature space into a quality score. TMIQA metrics that adopt the second approach are considered in Chapter 2.

1.1 Research Goals and Work Contributions

Objective IQA metrics are usually classified into one of three groups, depending on how the processed image (also referred to as the test image) quality is evaluated [11]: full-reference (FR), reduced-reference (RR), or no-reference (NR). FR metrics require a reference image (*i.e.* the distortion-free, original image) with which the test image is compared in order to yield a quality score. RR metrics use a limited set of parameters extracted from the original image, rather than the entire original image. The output quality scores from RR metrics depend on the comparison between the reference parameter values and the corresponding parameter values obtained from the test image. Finally, NR metrics assess the image quality based solely on information obtained from the test image, that is to say, they do not require a reference image nor any information from it.

This work is dedicated to the study of NR IQA metrics in the context of tone-mapped images. One problem observed in currently available state-of-the-art TMIQA metrics is that they provide reliable quality scores only for a limited set of tone-mapped images, that is, their quality scores do not summarize well the overall quality impression of any tone-mapped image. The feature design process performed by state-of-the-art TMIQA metrics demands extensive domain knowledge about tone-mapping aspects. Proposing new features that capture important quality aspects in any tone-mapped image is a difficult task. Deep learning models are able to automatically learn relevant features for the TMIQA task by analyzing a wide variety of tone-mapping examples.

This work has two main goals. The first one is to propose a new NR TMIQA metric that is based on deep learning models. To the best of our knowledge, no work has been proposed yet in which the TMIQA task is fully performed by a deep learning model, in an end-to-end fashion (*i.e.* a deep learning model that receives a tone-mapped image and yields a quality score for it). The second goal is to present our tone-mapped image database. The database is divided in two parts. For the first part, we build a new extensive tone-mapped image database, containing approximately 175000 samples. All samples have quality scores given by objective state-of-the-art TMIQA metrics. For the second part, a subset of 3009 samples has quality scores obtained from human evaluators through subjective tests. A good insight of the TMIQA problem is first needed to guide the selection of the most suitable deep learning architecture for the TMIQA task. More specifically, our main goals can be subdivided into three partial goals:

1) To understand in more detail how certain image attributes impact on overall tone-mapping quality. This involves analyzing a variety of tone-mapped images, and searching for common properties that are usually associated with "good" and "bad" quality (*e.g.* high contrast, vivid colors, low brightness, and so on).

2) To examine to what extent tone-mapping quality assessment is similar to other IQA-related tasks (contrast distortion assessment or aesthetic quality assess-

ment, for instance). This involves applying general purpose IQA metrics (*i.e.* metrics that are not dedicated to assess tone-mapping quality) to tone-mapped images, and identifying in which cases, if any, the scores from such metrics reliably represent the subjective quality impression of tone-mapped images.

3) To investigate whether deep learning models are able to overcome the generalization limitation observed in state-of-the-art TMIQA metrics. This involves using a very large number of images representing, as best as possible, "fine-grained grades" of tone-mapping quality (*e.g.* "very bad", "moderately bad", "average", "moderately good", "very good", and so on) to train the deep learning models. Their performances are then evaluated in benchmark tone-mapped image databases, in which state-of-the-art TMIQA metric performance values are reported.

Our work main contributions are summarized in three points:

1) Introduction of a new extensive tone-mapped image database designed for the TMIQA task. The database contains approximately 175000 samples that represent a rich diversity of scenes. Each sample has quality scores given by four state-of-the-art TMIQA metrics, namely the Tone-Mapped image Quality Index (TMQI) [12], the Blind Tone-Mapped Quality Index [13] (BTMQI), and two versions of the HDR Image GRADient Evaluator (HIGRADE) metric [14]. We also perform subjective experiments to assess the quality of 3009 representative samples from this database. Besides objective quality scores, such samples also contain subjective quality scores obtained from human evaluators. We show that the subjective scores are consistent and reliable quality indicators for samples in this database. This subjectively evaluated sample subset is larger than the largest benchmark database designed for the TMIQA task currently available in the literature (which contains 1811 samples).

2) Presentation of a performance comparison between deep learning architectures that are trained for the TMIQA task, using different quality scores. More specifically, we present a study that investigates how model generalization capability changes according to the score used as reference to represent tone-mapped sample quality. We consider quality scores obtained from multiple TMIQA metrics, as well as from human evaluators. We show that deep learning model performance greatly varies across different TMIQA databases depending on which objective TMIQA metric is used as target quality score during training. Models trained with subjective quality scores seem to generalize better their quality predictions than those trained with objective quality scores. However, lack of more samples whose qualities are subjectively labeled

poses a challenge for improving deep learning model performances. We also compare model performances when trained with subjective quality scores obtained from two different subject groups. The first group contains fewer, but more reliable subjects, whereas the second group contains many, but more unreliable subjects. Models trained with subjective scores from the first group have better generalization capability than the ones trained with subjective scores from the second group.

3) Investigation of ways to improve TMIQA reliability using currently available NR IQA metrics. Most state-of-the-art NR IQA metrics are not specifically designed to assess tone-mapping distortions. The ones that are designed for this purpose have limited performance when evaluating the quality of tone-mapped images that do not belong to specific datasets, as we show in this work. We study different approaches of combining pieces of information from available NR IQA metrics in order to obtain new TMIQA metrics that possibly overcome such generalization limitation. We consider two main approaches. In the first one, we use regression and classification models based on different traditional machine learning algorithms to combine quality scores provided by multiple state-of-the-art NR IQA metrics into a single quality score. In the second one, we use the handcrafted features extracted by three TMIQA metrics to train these models. They learn how to map values from this feature space into a quality score.

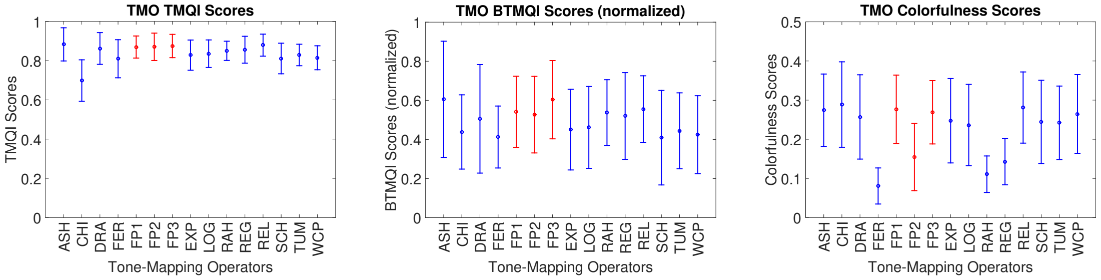
Chapter 2

TMIQA Related Work

In this chapter, we review a previous work [15] in which we compared the quality of tone-mapped images produced by different TMOs. Such work served as inspiration and basis for the present work. Then, we review the current state of TMIQA field. We list some state-of-the-art NR metrics available in the literature, and identify the ones designed for assessing tone-mapping distortions specifically. We present two benchmark tone-mapped image datasets commonly used for the TMIQA task and indicate some of their differences. Finally, we show baseline results from several state-of-the-art NR IQA metrics applied to tone-mapped images that come from the benchmark datasets.

2.1 TMO Performance Comparison

During the present research, we published an article in Elsevier Signal Processing: Image Communication journal [15], which helped us to better understand general aspects of tone mapping, such as the different ways in which the operators work, and which distortions are typically caused by TMOs. It has also served as basis for the current research, as we studied more details about the currently available state-of-the-art TMIQA metrics, and the image aspects these metrics usually consider when assessing tone-mapping quality. In that work, we performed a systematic comparison between several TMOs in terms of image quality and processing speed. In particular, we investigated potential performance differences between TMOs developed to run as software, in digital domain (called “digital TMOs”, or DTMOs), and one TMO proposed to be implemented in hardware, at the camera focal plane (called “focal-plane TMO”, or FPTMO). In the considered FPTMO, the tone-mapping operation takes place inside each pixel simultaneously, and concurrently with the image capture. The idea was to verify whether design restrictions imposed by hardware limitations cause the FPTMO to yield images with worse quality than the quality of those produced by DTMOs, which do not suffer from such restrictions. For the



(a) (b) (c)

Figure 2.1: Errorbar plots illustrating different TMO performance values considering scores from different quality metrics: (a) TMQI, (b) BTMQI (normalized), and (c) colorfulness. Red indicates software implementations that represent the FPTMOs. In all plots, higher scores indicate better quality.

comparison, the FPTMO was described by software models which take into account focal-plane restrictions for simulating the in-pixel operations. Each software model also considers different implementation decisions.

Each TMO was applied to $M = 25$ different HDR scenes, and the resulting tone-mapped images were evaluated by objective quality assessment metrics. When searching for such metrics, we mostly found metrics that are not specifically designed for tone-mapped images, as they do not focus on tone-mapping distortions, but rather on distortion types normally caused by other operations commonly applied to images in general (this is further discussed in Chapter 2). Only a few metrics dedicated to TMIQA were found. Two TMIQA metrics with publicly available implementations were chosen: BTMQI, which is an NR metric, and TMQI, which is an FR metric. These metrics assess overall tone-mapped image quality by considering aspects usually affected by TMOs, like amount of detail (represented as entropy values), structure artifacts and image naturalness. Neither TMQI nor BTMQI take into account image color appearance when assessing quality. After experimenting with several objective metrics, a third metric was chosen to analyze only color appearance by measuring image colorfulness [16]. Unlike TMQI and BTMQI, the colorfulness metric is designed for any kind of image, and not only for tone-mapped images. Figure 2.1 shows the TMO errorbar plots, considering each quality metric. Red bars identify the software implementations that represent the FPTMOs.

We applied statistical hypothesis tests to further verify whether the differences observed between TMO performance values are statistically significant. In these tests, TMOs are pairwise compared using their corresponding quality scores from one objective metric. Results from these tests indicate whether both TMOs have equal performance, or, if not, which TMO has superior performance [15]. For each TMO pair, tests are conducted three times. At each time, the corresponding qual-

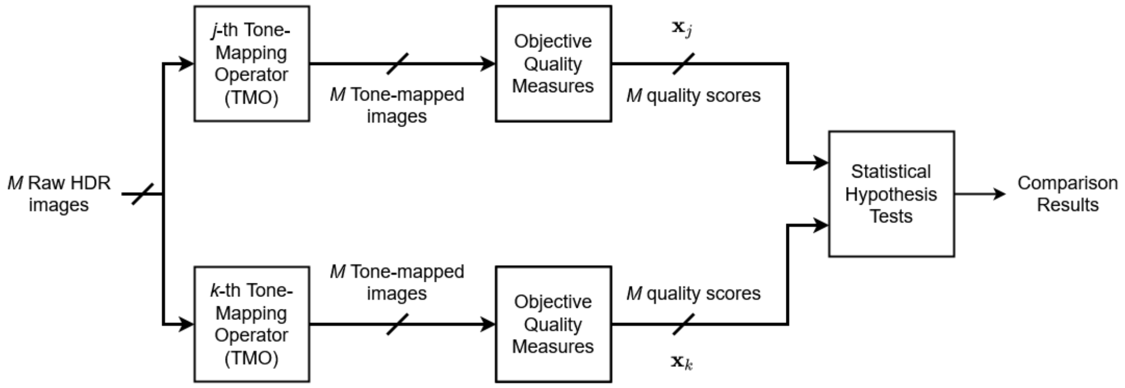

j-th Tone-Mapping Operator (TMO)
M Tone-mapped images
Objective Quality Measures
$\mathbf{x}_j$
M quality scores
M Raw HDR images
Statistical Hypothesis Tests
Comparison Results
k-th Tone-Mapping Operator (TMO)
M Tone-mapped images
Objective Quality Measures
M quality scores
$\mathbf{x}_k$

Figure 2.2: Framework for comparing TMO performance in terms of image quality. $\mathbf{x}_i$ denotes the vector containing $M = 25$ quality scores from one objective metric, considering the images generated by the i-th TMO.

ity scores from a different metric (BTMQI, TMQI, or colorfulness) are considered. The framework adopted to compare TMO performance regarding image quality is summarized in Figure 2.2. Hypothesis test results reported in [15] indicate that, considering the three metrics, the FPTMO models have performance values similar to the best-performing DTMOs, regarding image quality. These results suggest that focal-plane hardware limitations do not particularly compromise overall image quality. The FPTMOs have the advantage of running significantly faster than the best-performing DTMOs, as demonstrated by the execution time comparison conducted in [15].

2.2 Deep Learning in TMIQA

Deep learning models have not been much explored for the TMIQA task. These models have been successfully applied to other image processing tasks, including general IQA [17–19] tasks. Some initial works that use deep learning models as feature extractors have been proposed [20, 21]. In [20], a tone-mapped image is fed into a ResNet architecture [22], and corresponding feature maps from several internal layers are extracted. In [21], an U-net architecture [23] is trained to predict distortion maps from an input tone-mapped image. The reference distortion maps, which the deep learning model learns to predict, are computed by an external algorithm [24] that requires both the reference HDR image and a tone-mapped version of it. In these works, the features obtained from the deep learning models (*i.e.* the feature maps in [20], and the distortion maps in [21]) are then used to train regression models, based on conventional machine learning algorithms. These regression models learn to predict the overall quality of an input tone-mapped image, which is represented by a single quality score. To the best of our knowledge, no work has been reported in the literature yet in which the TMIQA task is carried out entirely by a deep learning model, without requiring a separate model to perform quality score prediction. In the present work, we explore the application of some deep learning architectures available in the literature to the TMIQA task, in an end-to-end fashion. In other words, the deep learning model receives a tone-mapped image and outputs a quality score for it.

2.3 State-of-the-art NR IQA Metrics

Few NR IQA metrics are dedicated to assessing tone-mapped images [10]. State-of-the-art NR TMIQA metrics adopt a common framework. They extract features from tone-mapped images and use such features to train regression models based on conventional machine learning algorithms. Usually, the chosen regression model is the Support Vector Regression (SVR) machine [25]. The SVR model maps the feature vector into a score that represents the overall image quality impression.

The main difference between NR TMIQA metrics lies in the feature set each one of them extracts. The type of features and how they are obtained have been the research focus in this field. These features are "handcrafted", that is to say, they are calculated by applying carefully designed transformations to image pixels. For instance, the HIGRADE metric computes features that are associated with artifacts caused by TMOs in the so-called "image gradient domain"[1]. The BTMQI metric collects features related to three tone-mapped image aspects: (i) amount of detail, which is represented by entropy values; (ii) naturalness, which corresponds to statistical measures from the pixel value distribution; and (iii) structure, which is based on information from image edge maps[2]. Yue's CNS (Colorfulness, Naturalness, and Structure) metric [26] considers image colorfulness features that are obtained from image pixels represented in an opponent color space[3], as well as structure and naturalness features, which are both similar to the corresponding ones from the BTMQI metric. BIO-BLIND metric [27] applies several biologically inspired transformations to the input tone-mapped image, in order to extract features related to contrast and texture. The BLInd QUality Evaluator for Tone-Mapped Images (BLIQUE-TMI) [28] extracts local features based on image sparse representations,

[1] Gradient domain refers to images obtained by calculating the gradients around each pixel of an input image.

[2] Edge maps are binary images obtained from edge detector algorithms, which mark pixels representing object boundaries in the image.

[3] In opponent color space, pixel values are represented by a tuple (rg, yb). rg represents the difference between the red and green color channels. yb corresponds to the difference between a combination of the red and green color channels and the blue color channel. This combination is computed by averaging, at each pixel, the corresponding red and green channel values. This representation is claimed in [26] to be more similar to how the human visual system captures color information.

which are inspired on the human visual system perceptual behavior, as well as global features based on statistical measurements from image luminance and chromatic components. This metric uses an "Extreme Learning Machine" [29], rather than an SVR machine, to map the feature vector into a quality score.

In this work, we consider two NR TMIQA metrics, namely the HIGRADE and BTMQI metrics. Besides them, we gathered other NR IQA metrics reported in the literature that are not originally dedicated to analyzing tone-mapped images, and applied each one of them to tone-mapped image quality evaluation. Most of these IQA metrics predict quality based on the intensity of specific distortions that might be present in the test image, like blur and white noise. These distortions are normally caused by other operations commonly applied to images, such as data compression, instead of the tone-mapping operation itself. In this work, these distortions are referred to as "generic distortions", as they are not directly related to tone mapping. Some IQA metrics, namely Neural Image Assessment (NIMA) and Personalized Aesthetic Model (PAM) metrics, consider aesthetic elements, rather than distortions, to yield quality scores. Using metrics that analyze generic-distortion features, as well as metrics that consider aesthetic-related features, for evaluating tone-mapped images allows one to verify if these types of features are somehow relevant for tone-mapping quality assessment.

Table 2.1 presents a brief summary of all IQA metrics used in the present work. In this table, the second column categorizes the IQA metrics based on how their quality assessment learning process[4] is conducted: either "Opinion-Aware" or "Opinion-Unaware" [30]. "Opinion-Aware" refers to metrics that require images labeled with actual subjective quality scores, in order to learn to predict image quality. "Opinion-Unaware" denotes metrics that learn to assess image quality without using scores given by human observers. Instead, other information is used to represent image quality, such as distortion indicators for instance. In the third column of Table 2.1, we indicate what the features extracted by each IQA metric represent: (i) generic distortions (*i.e.* distortions not necessarily related to tone mapping); (ii) tone-mapping distortions; and (iii) aesthetic elements. Finally, the last column provides an overview of the IQA metric. The dagger symbol within Table 2.1 indicates the IQA metrics that are based on deep learning models, namely RankIQA, NIMA and PAM. Deep learning models automatically learn, from a very large set of images, relevant features for the task at hand. In other approaches, features are handcrafted from fewer image examples.

[4] In this case, the "learning process" term has a loose meaning. The term refers to different strategies which make use of distinct premises and pieces of information, in order to learn how to perform the quality assessment task. It does not necessarily refer to training-based approaches, such as machine learning algorithms or deep learning models.

Table 2.1: IQA metrics considered in the present work for the TMIQA task. The dagger (†) symbol marks IQA metrics based on deep learning models.

IQA Metric	**Learning Process**	**Feature Type**	**Description**
BIQI [31]	Opinion-Aware	Generic Distortions	Uses statistics from wavelet-domain coefficients to estimate quality in a two-step framework.
BLIINDS-2 [32]	Opinion-Aware	Generic Distortions	Features are parameters from the statistical distribution of the image discrete cosine transform (DCT) coefficients.
BPADA [33]	Opinion-Aware	Generic Distortions	Trains an AdaBoosting back-propagated neural network using features computed from image gradient domain.
BRISQUE [34]	Opinion-Aware	Generic Distortions	Uses luminance-related features in spatial domain that correspond to parameters from natural scene statistic model.
BTMQI [13]	Opinion-Aware	Tone-mapping Distortions	Trains an SVR model using entropy-related, naturalness and structure features from tone-mapped images
CDIIVINE [35]	Opinion-Aware	Generic Distortions	Complex-domain extension of DIIVINE algorithm to account for phase information of wavelet coefficients.
CORNIA [36]	Opinion-Aware	Generic Distortions	Features correspond to similarities between local image descriptors and visual codebooks that were learned previously, in unsupervised fashion.
DESIQUE [37]	Opinion-Aware	Generic Distortions	Computes features based on image log-derivative statistics in both space domain and frequency domain.
DIIVINE [38]	Opinion-Aware	Generic Distortions	Similar to BIQI, but uses a larger feature set calculated from wavelet coefficients that are obtained from a steerable pyramid decomposition.
ENIQA [39]	Opinion-Aware	Generic Distortions	Uses entropy-based features that take into account image color information.
FRIQUEE [40]	Opinion-Aware	Generic Distortions	Computes statistical values from feature maps that are obtained from different color spaces and transform domains.
GM-LOG [41]	Opinion-Aware	Generic Distortions	Features are joint statistics of local contrast measures obtained from image gradient magnitudes and laplacian of gaussian response.
HIGRADE [14]	Opinion-Aware	Tone-mapping Distortions	Calculates gradient-based features that capture processing artifacts in tone-mapped images.
ILNIQE [42]	Opinion-Unaware	Generic Distortions	Similar to NIQE, but uses a larger feature set, and averages local quality scores from patches to obtain the final image quality score.
NFERM [43]	Opinion-Aware	Generic Distortions	Uses models inspired in the human visual system to calculate features.
NIMA† [18]	Opinion-Aware	Aesthetic Elements	Uses a deep learning model to predict, for a given image, a score distribution, and uses its mean value to represent image quality.
NIQE [44]	Opinion-Unaware	Generic Distortions	Yields quality scores by comparing statistics from several test image patches with corresponding statistics that represent distortion-free natural images.
NIQMC [45]	Opinion-Unaware	Generic Distortions	Combines both local and global entropy-based features to yield quality scores.
NJQA [46]	Opinion-Unaware	Generic Distortions	Estimates quality of JPEG compressed images by using quality relevance maps and counting zero-valued DCT coefficients.
PAM† [19]	Opinion-Aware	Aesthetic Elements	Uses a residual-based learning strategy to train a model that predicts image quality scores according to personal aesthetics preferences.
RANKIQA† [17, 47]	Opinion-Aware	Generic Distortions	Predicts image quality by using a model that is first trained to learn to rank images from pairs based on distortion intensities present on them.
SISBLIM [48]	Opinion-Unaware	Generic Distortions	Considers intensities of single distortion types as well as mutual effects between distortions to output image quality.
SSEQ [49]	Opinion-Aware	Generic Distortions	Uses spatial and spectral (in DCT domain) entropy-based features to compute image quality.

2.4 TMIQA Datasets

As mentioned in the first paragraph of Section 2.3, the extracted features are mapped into quality scores by SVR machines. In order to learn how to perform such mapping, the SVR machines must be trained using datasets that contain images labeled with quality scores obtained from subjective tests. In these tests, several subjects are asked to rate image quality within a previously defined score range. Each image is then labeled with its average quality score value computed from the individual scores given by the subjects. These labels are called "Mean Opinion Score" (MOS)[5]. Only a couple of such datasets for tone-mapped images exist. In this work, we consider the two tone-mapped image datasets that are used as benchmark in many related works. Details regarding each dataset are provided next.

- ESPL-LIVE Dataset

  The ESPL-LIVE dataset [50] contains 1811 tone-mapped images depicting a variety of indoor and outdoor scenes under different lighting conditions (*i.e.* daytime and nighttime). This dataset is further subdivided into three groups, which classify how tone mapping is performed. The three groups are: (i) TMO, which refers to operators that realize tone mapping by receiving as input only one HDR scene captured with a fixed exposure time (747 samples); (ii) MEF ("Multiple-Exposure Fusion"), which denotes operators that combine several captures of the same HDR scene, each one with a different exposure time, in order to generate the final tone-mapped image (710 samples); and (iii) PP ("Post-Processing"), which corresponds to operators that introduce artificial elements mostly for artistic purposes, such as over-saturated colors and contrast effects (354 samples). Subjective tests were performed via online crowdsourcing, using more than 5000 human observers to rate the quality of images from the dataset. On average, each image was evaluated by 110 observers. Subjective MOS values correspond to the average score value calculated from scores given by the observers. MOS values roughly vary from 15 (worst quality) to 70 (best quality). To the best of our knowledge, this is the largest tone-mapped dataset labeled with subjective MOS reported in the literature.

- TMID Dataset

[5] In this work, we use the term "subjective MOS" to denote quality scores obtained from subjective tests, and "predicted MOS" to refer to quality scores obtained from objective IQA metrics.

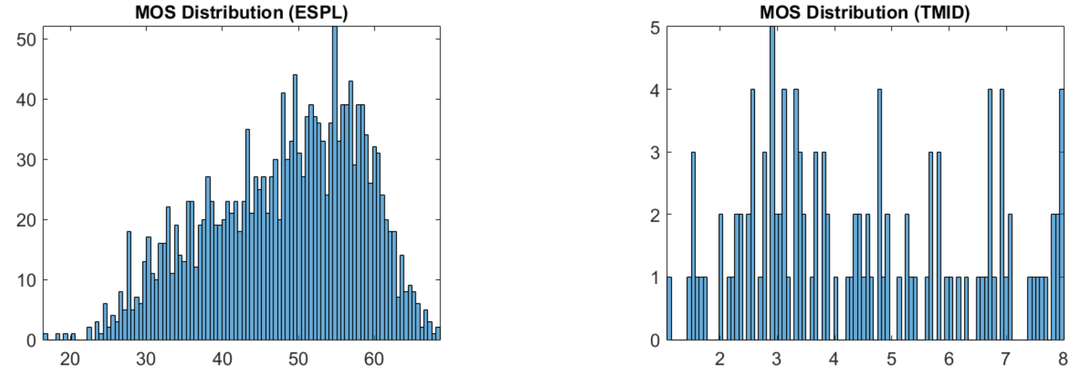
> The TMID dataset was proposed to test the TMQI metric, which is a state-of-the-art FR TMIQA metric. In this dataset, 15 different HDR scenes are tone-mapped by eight different TMOs, thus yielding 120 tone-mapped images. Then, for each HDR scene, 20 subjects are asked to rank the eight tone-mapped images, with rank "1" denoting the best quality, and rank "8" denoting the worst quality. The subjective MOS for each tone-mapped scene corresponds to the average ranking label from the subjects.

Figure 2.3 shows the MOS distributions from each dataset. In TMID dataset, MOS values are more evenly distributed, whereas, in ESPL-LIVE dataset, MOS values are concentrated around MOS value of 55. Figure 2.4 shows images containing examples of different distortions normally caused by TMOs. Tone-mapping operation has an impact on the overall image contrast, and it may introduce some degradation, such as haloing artifacts, over-exposure of bright areas, or under-exposure of dark areas. Tone mapping may also affect the image color appearance by exceedingly increasing or decreasing saturation of colors, or even generating unrealistic colors. Such distortions are highly noted in "bad" tone-mapping examples. In general, the idea of "good" tone mapping involves images with natural-looking colors and with clear visibility of details in bright regions and in dark regions, simultaneously. Figure 2.5 illustrates examples of "good" tone mapping. Observing samples from both datasets and their respective subjective MOS values, a general idea of "good" and "bad" tone-mapping quality can be established.

We note that, in TMID dataset, "bad" tone mapping (*i.e.* samples with scores closer to 8) is mostly represented by images with excessively large areas having either saturated pixels or very dark pixels. In ESPL-LIVE dataset, "bad" tone-mapping quality (*i.e.* samples with scores closer to 15) also includes this type of distortion. However, in this dataset, other tone-mapping distortions, such as pronounced haloing artifacts and very saturated colors, are dominant among the examples of "bad tone-mapped image quality". Figure 2.6 illustrates "bad quality' examples most commonly observed in each dataset.

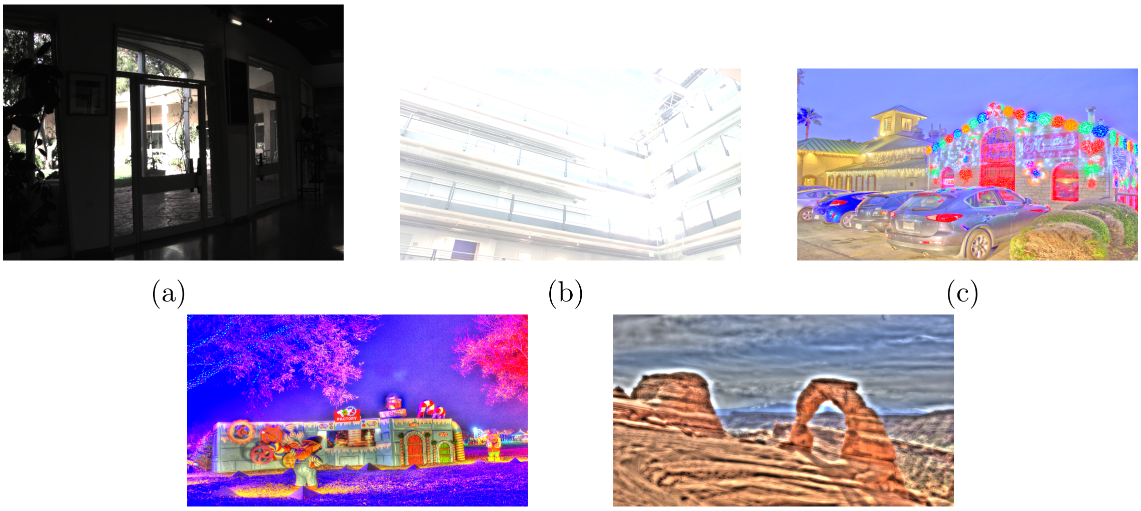
2.5 IQA Metric Performance Values for TMIQA

We analyze the suitability of each IQA metric for the TMIQA task by comparing their output quality scores with subjective MOS values of samples in ESPL-LIVE and TMID datasets. Three performance metrics are used: Pearson Linear Correlation Coefficient (PLCC), Spearman Rank Correlation Coefficient (SRCC), and Root-Mean-Squared Error (RMSE). Each metric evaluates performance under a different aspect [51]. PLCC measures the IQA metric ability to correctly predict



(a) (b)

Figure 2.3: MOS distributions in (a) ESPL-LIVE, and (b) TMID datasets. Higher image quality is associated with higher MOS values in ESPL-LIVE dataset, and lower MOS values in TMID dataset.



(a) (b) (c)

(d) (e)

Figure 2.4: Examples of tone-mapping distortions: (a) under-exposed areas, (b) over-exposed areas, (c) highly saturated colors, (d) unnatural-looking colors, and (e) haloing artifacts.

subjective MOS values. SRCC indicates prediction monotonicity, that is to say, the degree of agreement between relative magnitudes (or rankings) of predicted scores and of subjective MOS values. Finally, RMSE measures the IQA metric prediction consistency (or robustness) over a set of images (*e.g.* similarly distorted images should yield similar scores). The best possible performance values correspond to PLCC and SRCC equal to 1, and RMSE equal to 0.

We follow the recommendations from [51, 52], for reporting objective IQA metric performance. This is the standard protocol adopted by similar works in IQA field [13, 14, 27, 30]. For each IQA metric, the output quality scores are used, along with the subjective MOS values, to fit a non-linear function, which converts the output

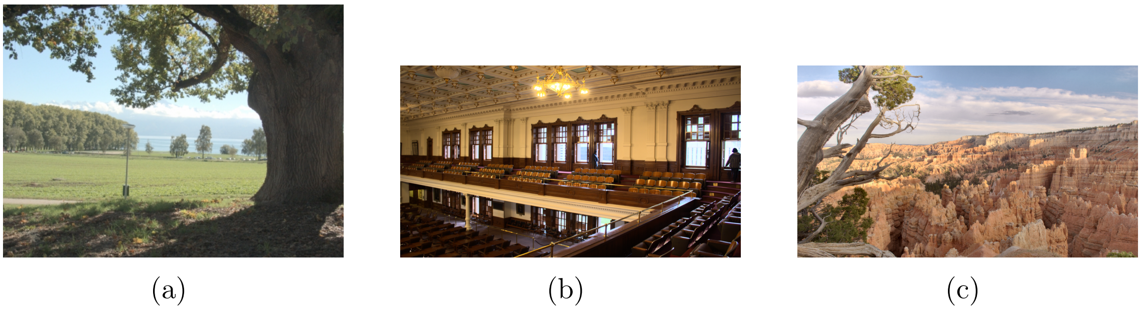


Figure 2.5: Examples of good tone mapping. MOS values and respective datasets of each image are: (a) 2.3 (TMID), (b) 65.6 (ESPL-LIVE), and (c) 65.2 (ESPL-LIVE). In TMID, MOS values range from 1 (best quality) to 8 (worst quality). In ESPL-LIVE, MOS values range from 15 (worst quality) to 70 (best quality). Good quality is often associated with aspects such as natural-looking colors, and rich number of details in bright regions (e.g. clouds in the sky in (a) and (c), and chandelier ornaments in (b)), and in dark regions (e.g. leaves on the floor in (a), first floor seats in (b), and canyon cracks in (c)) at the same time.

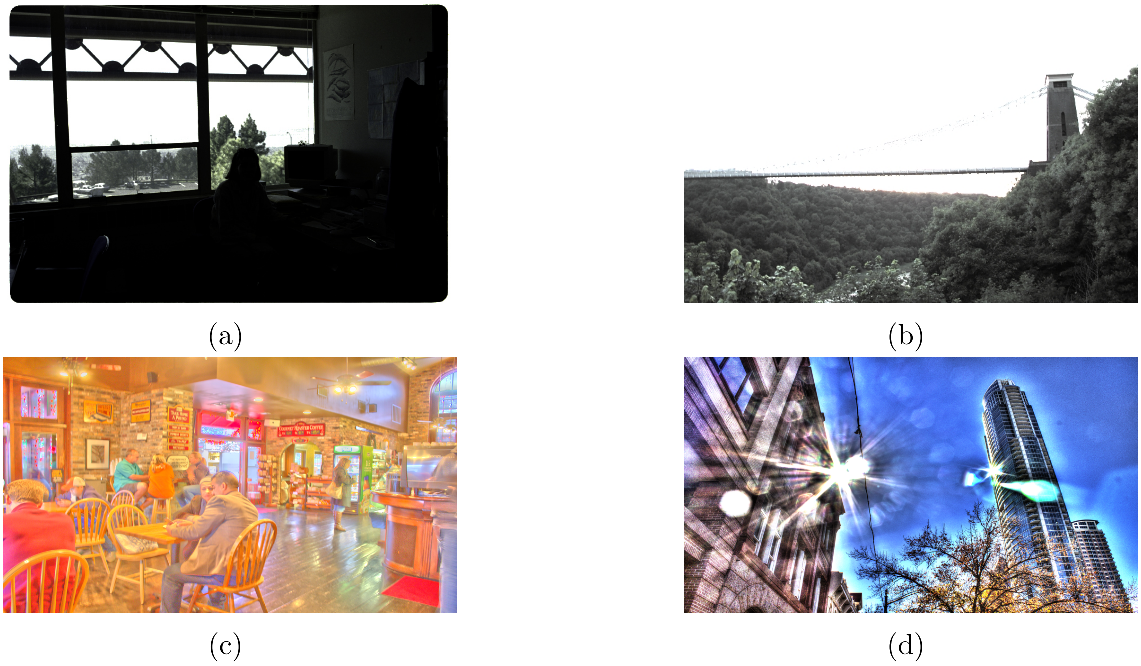


Figure 2.6: Examples of “bad” tone-mapped image quality most commonly found in TMID dataset (top row), and ESPL-LIVE dataset (bottom row). In the TMID dataset, MOS values range from 1 (best quality) to 8 (worst quality). In the ESPL-LIVE dataset, MOS values range from 15 (worst quality) to 70 (best quality). MOS values for each image are: (a) 5.7; (b) 7.6; (c) 25.9; and (d) 31.8.

quality scores into predicted MOS values for a given dataset. The fit function aims at removing compression-like non-linearities, caused by the subjective rating process, that may arise at both extremes of the score range [53]. These predicted MOS values and the corresponding subjective MOS values are then used to calculate the performance metric values. We choose the four-parameter logistic function as the non-linear fitting function [52]:

$$y = \beta_1 + \frac{\beta_2}{1 + e^{-\beta_3(x-\beta_4)}}, \tag{2.1}$$

where x is the output quality score from an IQA metric, y is the resulting predicted MOS value, and $\{\beta_1,\ \beta_2,\ \beta_3,\ \beta_4\}$ are the parameters tuned via regression, so as to provide the best fit in a least-squares sense. Unless otherwise stated, in all experiments reported in this work, performance metric values are obtained following this protocol.

The HIGRADE and BTMQI metric implementations that are publicly available have their training [13, 14] based on the entire ESPL-LIVE and TMID datasets, respectively. To avoid overestimating their performance values in these datasets, we retrain the HIGRADE metric in the ESPL-LIVE dataset, and the BTMQI metric in the TMID dataset. In each case, the corresponding dataset is randomly split into a training subset (comprising 80% of the total number of samples), and a test subset (containing the remaining 20% of samples). These metrics are trained and tested 1000 times, using different training and test sets each time, in order to remove performance biases potentially caused by one specific split. The other IQA metrics are not retrained. For each one of the 1000 trials, the corresponding IQA metric performance values (PLCC, SRCC, and RMSE) are calculated in the test subset. Following similar works in the literature that adopt such testing procedure, we report, for each IQA metric, the median performance values obtained from the 1000 trials. Arguably, the median value is a more reliable estimate for performance than the mean value, as it is less sensitive to extreme performance values that may be achieved with some training/test subset pairs.

Table 2.2 shows the IQA metric performance values. The scatter plots between the subjective and predicted MOS of each IQA metric can be found in Appendix A. HIGRADE metric is subdivided into two versions: HIGRADE-1 and HIGRADE-2. For each version, the SVR model is trained with a different feature set. More specifically, what distinguishes the feature sets is how features in the image gradient domain are calculated. The same reasoning applies to the SISBLIM metric, which is subdivided into four versions. Each version uses a different technique to estimate features that are associated with image noise level.

In ESPL-LIVE dataset, apart from HIGRADE metrics, all IQA metrics have poor performance values, including the BTMQI metric, which is dedicated to TMIQA. Besides the HIGRADE metrics, the BTMQI metric is also outperformed by NIQMC, which is not a metric developed for tone-mapping distortions specifically.

Table 2.2: IQA metric performance values in the ESPL-LIVE and TMID datasets. In each column, the reported result is the median value from the 1000 performance metric values that are calculated using the test sets of the corresponding trials. Boldface text indicates the best result for each performance metric in the corresponding dataset.

IQA Metric	**ESPL-LIVE Dataset**			**TMID Dataset**		
	PLCC	**SRCC**	**RMSE**	**PLCC**	**SRCC**	**RMSE**
BIQI	0.173	0.166	9.873	0.357	0.301	1.809
BLIINDS-2	0.065	0.052	9.997	0.521	0.447	1.662
BPADA	0.217	0.171	9.772	0.652	0.587	1.462
BRISQUE	0.101	0.103	9.968	0.560	0.497	1.603
BTMQI	0.399	0.403	9.183	**0.829**	**0.725**	**1.054**
CDIIVINE	0.148	0.165	9.915	0.473	0.245	1.691
CORNIA	0.269	0.253	9.787	0.200	0.184	1.894
DESIQUE	0.147	0.134	9.920	0.559	0.438	1.601
DIIVINE	0.102	0.103	9.979	0.379	0.303	1.784
ENIQA	0.296	0.301	9.578	0.188	0.146	1.899
FRIQUEE	0.348	0.336	9.393	0.363	0.288	1.800
GM-LOG	0.014	0.002	10.017	0.141	0.142	1.915
HIGRADE-1	**0.800**	**0.766**	**6.139**	0.439	0.409	1.740
HIGRADE-2	0.788	0.746	6.325	0.539	0.351	1.623
ILNIQE	0.216	0.226	9.822	0.322	0.249	1.833
NFERM	0.178	0.154	9.858	0.326	0.244	1.822
NIMA	0.206	0.160	9.797	0.323	0.300	1.825
NIQE	0.094	0.091	9.976	0.571	0.488	1.593
NIQMC	0.466	0.452	8.869	0.610	0.533	1.531
NJQA	0.124	-0.088	9.948	0.512	0.464	1.653
PAM	0.241	0.162	9.729	0.320	0.228	1.831
RANKIQA	0.252	0.235	9.712	0.108	0.137	1.922
SISBLIM-SFB	0.169	0.164	9.883	0.321	0.245	1.833
SISBLIM-SM	0.240	0.203	9.739	0.205	0.171	1.890
SISBLIM-WFB	0.155	0.151	9.906	0.332	0.265	1.828
SISBLIM-WM	0.213	0.186	9.796	0.162	0.188	1.912
SSEQ	0.137	0.135	9.982	0.622	0.527	1.521

In TMID dataset, performance values are more discrepant[6]. In this case, BTMQI is the best-performing metric, whereas the HIGRADE metrics present rather poor performance values, and they are also outperformed by other IQA metrics. This suggests that the quality predictions from TMIQA metrics do not generalize well for any kind of tone-mapped image. They are highly dependent on the dataset from which the test sample come.

[6] Such high performance discrepancies probably occur because performance statistics are calculated over a smaller score set, as the TMID dataset has fewer samples than the ESPL-LIVE dataset.

2.6 Chapter Summary

In this chapter, we briefly reviewed the work in [15] that motivated and served as basis for the present work. In [15], TMOs are compared in terms of output image quality using two state-of-the-art TMIQA metrics: TMQI and BTMQI. Both of these metrics, along with the HIGRADE metric, are the three TMIQA metrics considered in the present work. We listed other state-of-the-art NR IQA metrics and highlighted the features that are used by each one of them. We presented the two publicly available datasets used as benchmark in the TMIQA task, namely ESPL-LIVE and TMID. We pointed out some differences between these datasets, such as their MOS distributions and the tone-mapping distortions that are mostly found in each dataset.

Finally, we applied the IQA metrics indicated in Table 2.2 to evaluate the quality of tone-mapped samples from each dataset. Based on the performance values from each metric, we concluded that none of the considered metrics can reliably predict tone-mapped image quality in both datasets. The best-performing metric in TMID dataset (BTMQI) performs poorly in ESPL-LIVE, whereas the best one in ESPL-LIVE (HIGRADE-1) performs poorly in TMID.

In the next chapter, we investigate possibilities for obtaining better correlation coefficients between predicted quality scores and MOS values in both datasets. More specifically, we present experiments in which we train regression and classification models, based on conventional machine learning algorithms, that combine quality predictions from multiple IQA metrics into a single quality score. We also consider using all features extracted by both HIGRADE versions and BTMQI to train such models.

Chapter 3

TMIQA Experiments with Regression and Classification Models Based on Machine Learning

We design four experiments to investigate whether the IQA metrics presented in Chapter 2 can be combined in order to yield quality scores that have high correlations with subjective MOS of tone-mapped images from different datasets. The premise is that subjective MOS values should have higher correlations with quality scores obtained from a "committee" of IQA metrics than with those obtained from any single IQA metric considered in this work. In the first experiment, we train different regression models, using the quality scores predicted by IQA metrics as features for such models. In the second experiment, we directly use the handcrafted features extracted by TMIQA metrics (namely HIGRADE-1, HIGRADE-2, and BT-MQI metrics) to train regression models. In the third experiment, we group each tone-mapped image into one of three categories ("good", "average", or "bad") that represent the image quality, and train classification models to predict the category to which a tone-mapped image belongs. In these three experiments, we use the ESPL-LIVE dataset to train and validate the regression/classification models, and the TMID dataset to test the trained models. In the fourth experiment, we combine samples from ESPL-LIVE and TMID datasets, in order to build mixed training and testing datasets. Then, we repeat the methodology defined in the first and second experiments, but using the mixed datasets to obtain the regression models and evaluate their performance.

For the first, second and fourth experiments, we train four different regression models: K-Nearest Neighbors (KNN) [54], Support Vector Regression (SVR) machine, Random Forest [54], and Gradient Tree Boosting (GTB) [54]. For the third experiment, we train three different classification models: KNN, Support Vector

Table 3.1: Summary of experiments presented in this chapter. We point out the main aspects of each experiments: the training and test databases for the machine learning models; whether the TMIQA task is treated as a regression or classification problem; and the model training features considered.

Experiment Identifier	**Experiment Part**	**Training Database**	**Test Database**	**Task Type**	**Model Features**	**Section**
1	-	ESPL-LIVE	TMID	Regression	Selected IQA metric scores	3.1
2	-	ESPL-LIVE	TMID	Regression	TMIQA metric “raw” features	3.2
3	1	ESPL-LIVE	TMID	Classification	Selected IQA metric scores	3.3
	2	ESPL-LIVE	TMID	Classification	TMIQA metric “raw” features	3.3
4	1	ESPL-LIVE + TMID	ESPL-LIVE + TMID	Regression	Selected IQA metric scores	3.4
	2	ESPL-LIVE + TMID	ESPL-LIVE + TMID	Regression	TMIQA metric “raw” features	3.4

Machine (SVM) [54], and Random Forest. Both SVR and SVM models use radial basis functions as their kernel functions. The experiments are further explained in Sections 3.1 to 3.4. Table 3.1 summarizes the experiments presented in this chapter.

3.1 Experiment 1: IQA Metric Scores as Features for Regression Models

The first experiment is inspired by the works of [55] and [56], which use objective quality scores as features to train regression models for the HDR IQA task. Figure 3.1 depicts how quality assessment is performed in this experiment. The regression models considered in the present work use the ESPL-LIVE and TMID datasets for training and testing, respectively. As explained in Chapter 2, previous SVR machines [13, 14] from TMIQA metrics (*i.e.* HIGRADE-1, HIGRADE-2, and BTMQI metrics) were trained in these datasets. In order to avoid overestimating the training and test performance values of the regression models that use these TMIQA metric scores as features, we retrain the SVR machines from such metrics. All TMIQA metrics are now retrained in the same dataset, namely ESPL-LIVE, using 80% of its samples (1449 samples). The retrained TMIQA metrics and the other IQA metrics are applied to the remaining 20% samples (363 samples), thus yielding quality scores. These quality scores are then used as features to train the regression models. The entire TMID dataset (120 samples) is used to test the trained regression models. Figure 3.2 summarizes the dataset partition for training and testing the regression models, and for training the TMIQA metrics.

The regression models are trained following a procedure similar to the one defined

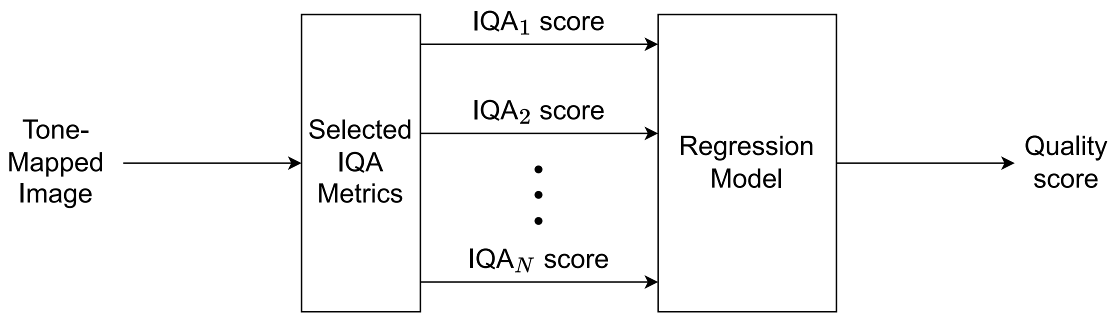


Figure 3.1: Schematic of how quality assessment is performed in the first experiment. Quality scores from selected IQA metrics are used as input for a regression model, which maps these scores into a single quality score. The IQA metrics that are considered depend on the regression model used (KNN, SVR, Random Forest or GTB). The IQA metrics are selected through a procedure called Sequential Forward Selection, which is explained in Section 3.1.

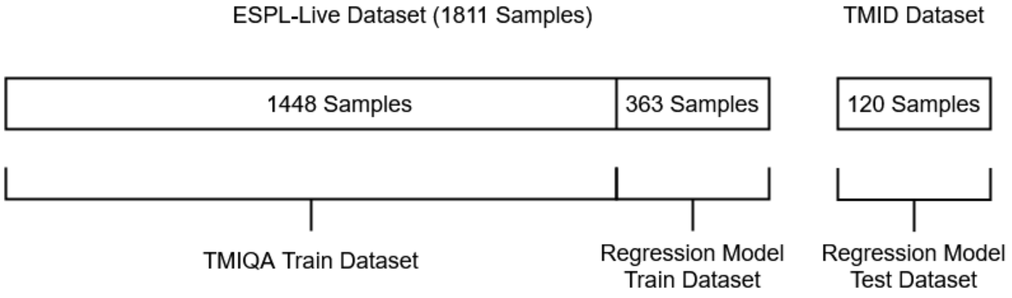

ESPL-Live Dataset (1811 Samples)
TMID Dataset
1448 Samples
363 Samples
120 Samples
TMIQA Train Dataset
Regression Model Train Dataset
Regression Model Test Dataset

Figure 3.2: Dataset partition for retraining the SVR machine from the TMIQA metrics, and for training and testing the considered regression models.

in [55]. This procedure uses a technique called Sequential Forward Selection (SFS) [57], which selects the subset of IQA metrics that yields the best regression model performance. The procedure is outlined next. The regression model training is divided into consecutive rounds. In the first round, several models are trained, each one using the quality score from a different IQA metric as a feature. The IQA metric that yields the best model performance in terms of SRCC[1] is saved for the next rounds, and the corresponding metric is removed from the metric pool. In each subsequent round, models are trained using as features the quality scores from different IQA metric combinations. Each combination comprises the IQA metrics selected in previous rounds, plus one new IQA metric from the remaining metrics in the pool. At the end of every round, the SRCC values are calculated for every trained model, and the best SRCC performance is evaluated. If it improves the best SRCC performance computed in the previous round by at least a minimum threshold, then the corresponding IQA metric combination is carried over into the next round, and the metric pool is updated. This training process is repeated until

[1] We choose SRCC, because this performance metric is less sensitive to the non-linear regression function applied to the model original scores.

the minimum SRCC improvement over the previous round is not achieved, or all IQA metrics are incorporated into the regression model feature set.

For every model that is trained in a round using a specific IQA metric combination, the corresponding hyperparameter set is tuned through a grid search strategy. For each point in the grid (consisting of one set of hyperparameter values), the model performance is evaluated using the k-fold cross-validation [58] (with $k = 10$), which is described next. The training dataset is divided into ten subsets. Each subset contains roughly the same number of samples. Then, a model is trained using nine subsets, and tested using the remaining subset. This process is repeated ten times, using different subsets to train and test a model each time, such that all samples are used at least once for training and testing. In the end of this process, ten models are trained. For each one of the ten models, performance metric values (PLCC, SRCC, and RMSE) are calculated using the corresponding test subset. The model performance with the given set of hyperparameter values is then represented by the median value of each performance metric. The hyperparameter values that yield the highest median SRCC performance are saved, and used to train the final regression model in the entire ESPL-LIVE dataset. Table 3.2 exemplifies the value range of some hyperparameters considered in the grid search, for each model. Model implementations are the ones from the *scikit-learn* [59] Python package (version 0.24.2). The package documentation defines, for each model, other hyperparameters (not listed in Table 3.2) that are associated with the package implementation decisions. Hyperparameters not listed in Table 3.2 are set to their default values defined by the package. Such hyperparameters and their default values can be found in Appendix B.

Table 3.3 reports the training results from each regression model using the best hyperparameter set, along with the associated IQA metric combination whose quality scores integrate the model feature set. Each performance metric column shows the median value from the corresponding set of values obtained from each test fold in cross-validation. Table 3.4 shows the corresponding regression model performance values when applied to the TMID dataset (*i.e.* the test dataset). In both tables, the best value for each performance metric is highlighted in boldface text. Although training results indicate a minor SRCC performance improvement of the regression models over the best-performing metric in the ESPL-LIVE dataset (HIGRADE-1, as shown in Table 2.2, with SRCC of 0.766), such improvements are not observed when these models are applied to the images from the TMID dataset. In fact, the regression model performance values are worse than some individual IQA metrics in the TMID dataset.

Figure 3.3 illustrates the scatter plots between subjective MOS and predicted MOS from each regression model evaluated in the TMID dataset. The highly dis-

Table 3.2: Hyperparameters and corresponding value ranges adopted in the grid search, for each regression model. The value ranges are defined empirically. N_{metrics} refers to the total number IQA metrics in the metric pool (in our case, 27, according to Table 2.2).

	Hyperparameter	**Value Range**
KNN	Number of Neighbors	1 to 30
SVR	Regularization Strength	$2^i, i = -4, -3, \ldots, 15$
Random Forest	Number of Trees	$2^i, i = 0, 1, \ldots, 9$
	Max Features	N_{metrics} to 1, in steps of -5
GTB	Number of Boosting Stages	$2^i, i = 1, 2, \ldots, 10$
	Boosting Stage Learning Rate	$10^i, i = -1, -2, \ldots, -5$
	Max Tree Depth Level	1 to 8
	Max Features	N_{metrics} to 1, in steps of -5

Table 3.3: Regression model training performance values in the ESPL-LIVE dataset, in the first experiment. The "Selected IQA Metrics" column presents the IQA metrics that are selected via SFS method and whose quality scores are used as features for training each considered model. Each performance metric column shows the median of the values obtained from the ten cross-validation test subsets. Boldface text indicates the best results for each performance metric.

Regression Model	**Selected IQA Metrics**	**PLCC**	**SRCC**	**RMSE**
SVR	'HIGRADE-1', 'BTMQI', 'PAM', 'NIMA,' 'NIQE', 'NJQA', 'GM-LOG', 'CDIIVINE'	0.790	**0.831**	6.586
KNN	'HIGRADE-1', 'BTMQI', 'PAM', 'NIQMC', 'CORNIA'	**0.798**	0.819	6.414
Random Forest	'HIGRADE-1', 'BTMQI', 'PAM', 'SISBLIM-WM', 'RANKIQA', 'CDIIVINE', 'NFERM', 'BPADA', 'NIQMC', 'SSEQ'	0.791	0.820	**6.314**
GTB	'HIGRADE-1', 'BTMQI', 'PAM', 'GM-LOG', 'ENIQA', 'NIMA', 'SISBLIM-SFB', 'BLIINDS-2'	0.788	0.829	6.592

persed nature of the scatter plots may cause the non-linear fit function from Equation (2.1) to suffer abrupt transitions and to have discontinuities. This is the case for the SVR model in Figure 3.3(b). In extreme cases, the fit function may map all predicted MOS values into a single value that corresponds to the center of mass of the data points. Examples of such cases are the BLIINDS-2 scores in ESPL-LIVE dataset,

Table 3.4: Regression model performance values in the TMID dataset, in the first experiment.

Regression Model	**PLCC**	**SRCC**	**RMSE**
SVR	**0.456**	**0.432**	**1.712**
KNN	0.416	0.296	1.750
Random Forest	0.442	0.367	1.726
GTB	0.437	0.352	1.730

Table 3.5: Regression model performance values in the TMID dataset, in the first experiment, without applying the non-linear fit function (Equation (2.1)) to the predicted MOS values from each model. Optimal performance values correspond to PLCC and SRCC values equal to -1. Boldface text indicates the best results for each performance metric.

Regression Model	**PLCC**	**SRCC**
KNN	-0.321	-0.296
SVR	-0.321	-0.350
Random Forest	$\mathbf{-0.397}$	$\mathbf{-0.367}$
GTB	-0.396	-0.352

and NJQA scores in TMID dataset, as shown in their scatter plots in Appendix A. Despite the better SVR performance reported in Table 3.4 in comparison to the performance of other models (especially in terms of SRCC), the scatter plots actually show that the SVR model has similar performance to other models (or, arguably, even worse performance than the others). Using non-smooth fit functions to map the predicted MOS values into the MOS value range of the test dataset affects all performance metric values, including the SRCC metric. As stated in the beginning of Section 3.1, SRCC should be insensitive to non-linear score mappings performed by the fit function, as long as the fit function is smooth. Table 3.5 shows the performance values considering the predicted MOS values from each model, without applying the non-linear fit function to map these values. In this table, all models have similar performance values, and these results are in more agreement with the scatter plots observed in Figure 3.3.

The higher SVR performance values reported in Table 3.4 most likely occur by chance because of the scores that are mapped by the non-linear fit function from Equation (2.1). The fit function parameters are obtained from the least-squares optimization. Samples with predicted MOS values less than approximately 47 are mapped into the same value. Most samples from this set also have high subjective MOS values (*i.e.* they represent “bad quality” tone mapping). These subjective MOS values are closer to the value into which the fit function maps the model predictions. A similar effect is observed for samples with predicted MOS values greater than 47. This justifies the higher performance values calculated for the SVR model. Although the application of the non-linear fit function to the predicted MOS values may influence the calculated performance values, such procedure is commonly adopted by similar works in the field to report the IQA metric performance. This fit aims at correcting possible non-linear aspects, caused by the subjective testing procedure, of the subjective MOS values from a dataset, as pointed out in Section 2.5.

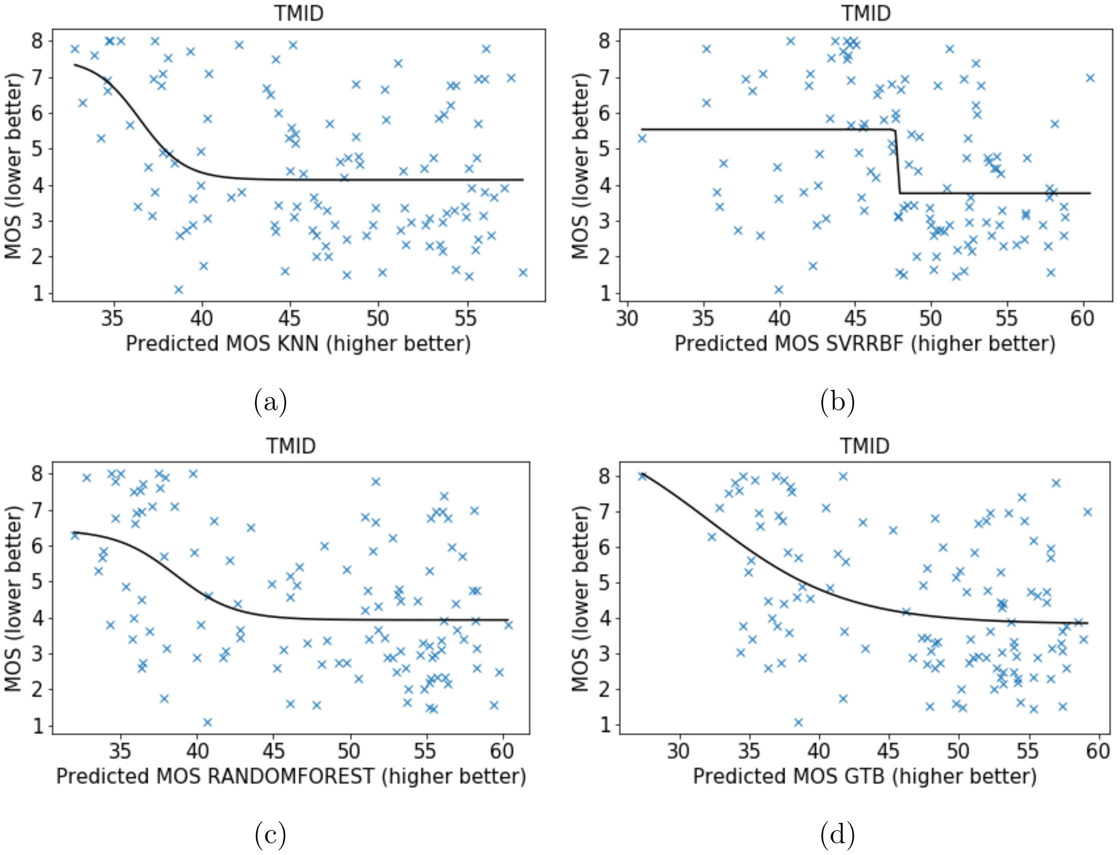
It is likely that the regression model poor performance values are caused by disagreements between predictions from the considered IQA metrics regarding the quality of tone-mapped images in this dataset. Combining multiple IQA metrics that yield highly discrepant scores for the same image may result into a noisy effect, which degrades the regression model performance. To illustrate this, Figure 3.4 shows images from TMID dataset with different scores given by HIGRADE-1, BTMQI, and PAM metrics (which are the IQA metrics selected by all regression models, according to Table 3.3). Each metric score (obtained after the non-linear fitting, as discussed in Section 2.5) ranges from 1 (best quality) to 8 (worst quality). This prediction-disagreement assumption is strengthened by the highly dispersed scatter plots of most individual IQA metrics, which are shown in Appendix A.

Another possibility is that TMID images and ESPL-LIVE images have different properties that are particularly important for identifying specific tone-mapping distortions. As pointed out in Section 2.4, from a subjective perspective, "bad" tone-mapping quality'[2] is mostly represented by different tone-mapping distortions in each dataset. In TMID dataset, over-exposed and under-exposed images dominate the "bad" tone-mapped image group. In ESPL-LIVE dataset, the most common tone-mapping distortions observed in "bad" tone-mapped images are pronounced haloing artifacts and very saturated colors. Apart from that, no other evidence that might distinguish samples of one dataset from another was found. From a statistical perspective, subjective MOS is more evenly distributed in the TMID dataset than in the ESPL-LIVE dataset, as shown in Figure 2.3.

3.2 Experiment 2: TMIQA Metric Features for Regression Models

In the second experiment, we feed the handcrafted features extracted by IQA metrics directly into the regression models, instead of using IQA metric quality scores

[2] In Section 3.3, it is stated that samples are classified as "bad quality" if their subjective MOS values are less than 43 in ESPL-LIVE dataset, and greater than 5.3 in TMID dataset.



(a)

(b)

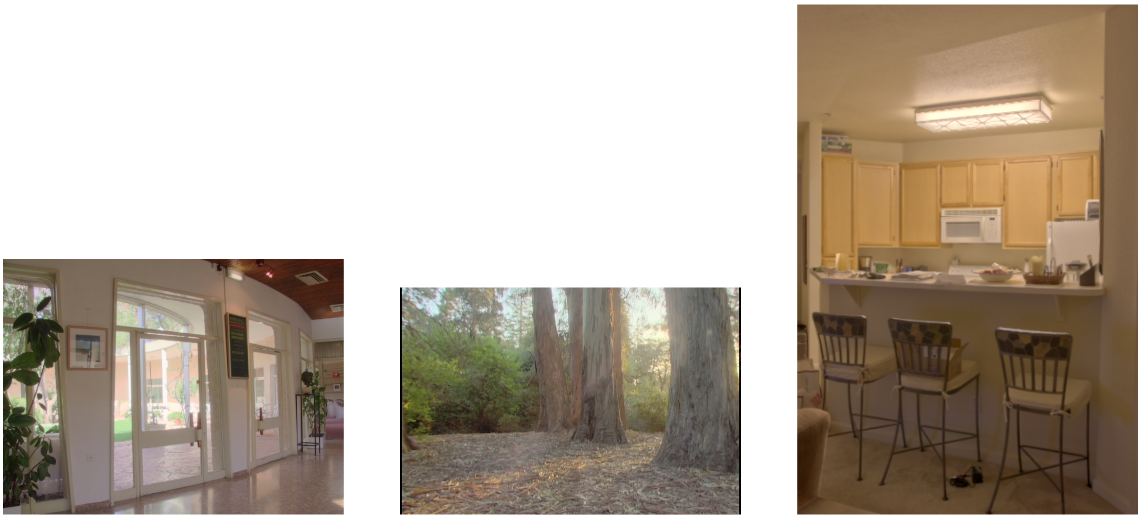
(c)

(d)

Figure 3.3: Scatter plots of the subjective MOS versus predicted MOS from each regression model, considering the samples from the TMID dataset, in the first experiment: (a) KNN, (b) SVR, (c) Random Forest, and (d) GTB. The black line shows the non-linear function (Equation (2.1)) fit to the data.

as features for such models. This is shown in Figure 3.5. Henceforth, we refer to the extracted handcrafted features as "raw" features. For this experiment, we use the entire ESPL-LIVE dataset to train the regression models, and the TMID dataset to test them. Because the total number of raw features obtained from the IQA metrics (38178 features) largely exceeds the number of samples available to train the regression models (1811 samples), we consider only the raw features from the TMIQA metrics, that is, HIGRADE-1, HIGRADE-2, and BTMQI metrics. Also, TMIQA raw features are designed specifically for evaluating distortions related to tone mapping, unlike raw features from other IQA metrics. The three TMIQA metrics extract 345 raw features (216 from HIGRADE-1, 120 from HIGRADE-2, and 9 from BTMQI), and all of them are used to train the regression models[3]. Hyperparameter tuning is performed for each regression model using the same strategy outlined in the first experiment (grid search and 10-fold cross-validation). Table 3.6 shows the regression model training results. Values reported in this table corre-

[3] We experimented with feature space dimensionality reduction using PCA with different numbers of principal components, but results were worse than using all features.



(a) (b) (c)

Figure 3.4: Images from TMID dataset with different scores obtained from individual IQA metrics (after the non-linear fitting described in Section 2.5). For all scores, the range is from 1 (best quality) to 8 (worst quality): (a) MOS: 1.5, HIGRADE-1: 4.1, BTMQI: 2.3, PAM: 5.0; (b) MOS: 2.6, HIGRADE-1: 6.2, BTMQI: 2.3, PAM: 3.6; (c) MOS: 1.1, HIGRADE-1: 4.1, BTMQI: 2.0, PAM: 4.9.

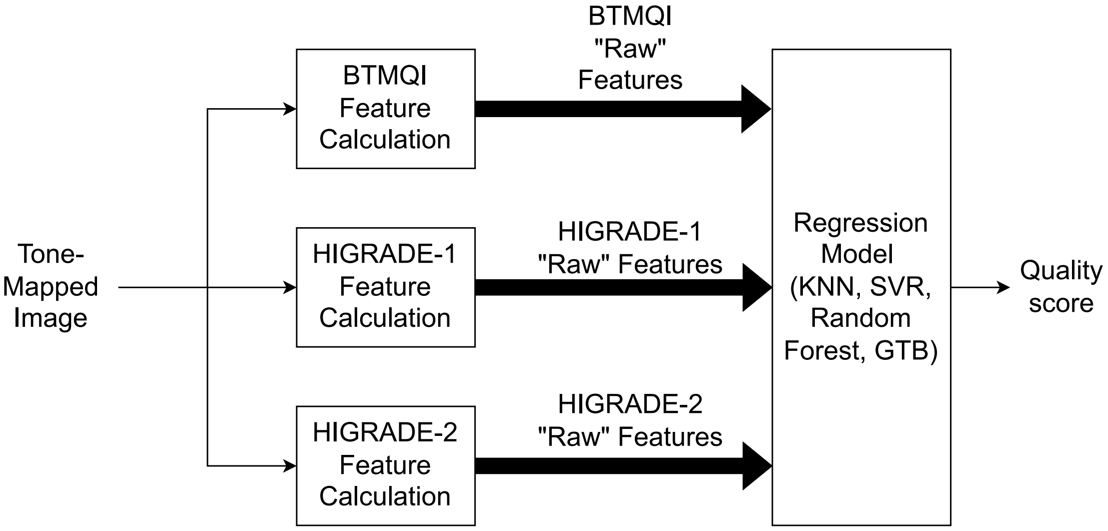

Tone-Mapped Image
BTMQI Feature Calculation
HIGRADE-1 Feature Calculation
HIGRADE-2 Feature Calculation
BTMQI "Raw" Features
HIGRADE-1 "Raw" Features
HIGRADE-2 "Raw" Features
Regression Model (KNN, SVR, Random Forest, GTB)
Quality score

Figure 3.5: Schematic of how quality assessment is performed in the second experiment. All features extracted by each considered TMIQA metric (denoted as "raw" features) are directly fed into a regression model. This model maps these feature values into a single score, representing the image overall quality.

spond to the median performance values obtained from the 10 performance metric values calculated across the cross-validation folds. The best model (GTB) does not outperform the best individual IQA metric performance in this dataset (which is the HIGRADE-1 metric, as reported in Table 2.2).

Table 3.7 shows the performance values of each regression model in the TMID dataset. The regression model performance values are also not better than the best

Table 3.6: Regression model training performance values in the ESPL-LIVE dataset, in the second experiment. Each performance metric column shows the median of the values obtained from the ten cross-validation test subsets. Boldface text indicates the best results for each performance metric.

Regression Model	**PLCC**	**SRCC**	**RMSE**
SVR	0.732	0.735	6.715
KNN	0.640	0.641	7.801
Random Forest	0.736	0.734	6.770
GTB	**0.748**	**0.753**	**6.634**

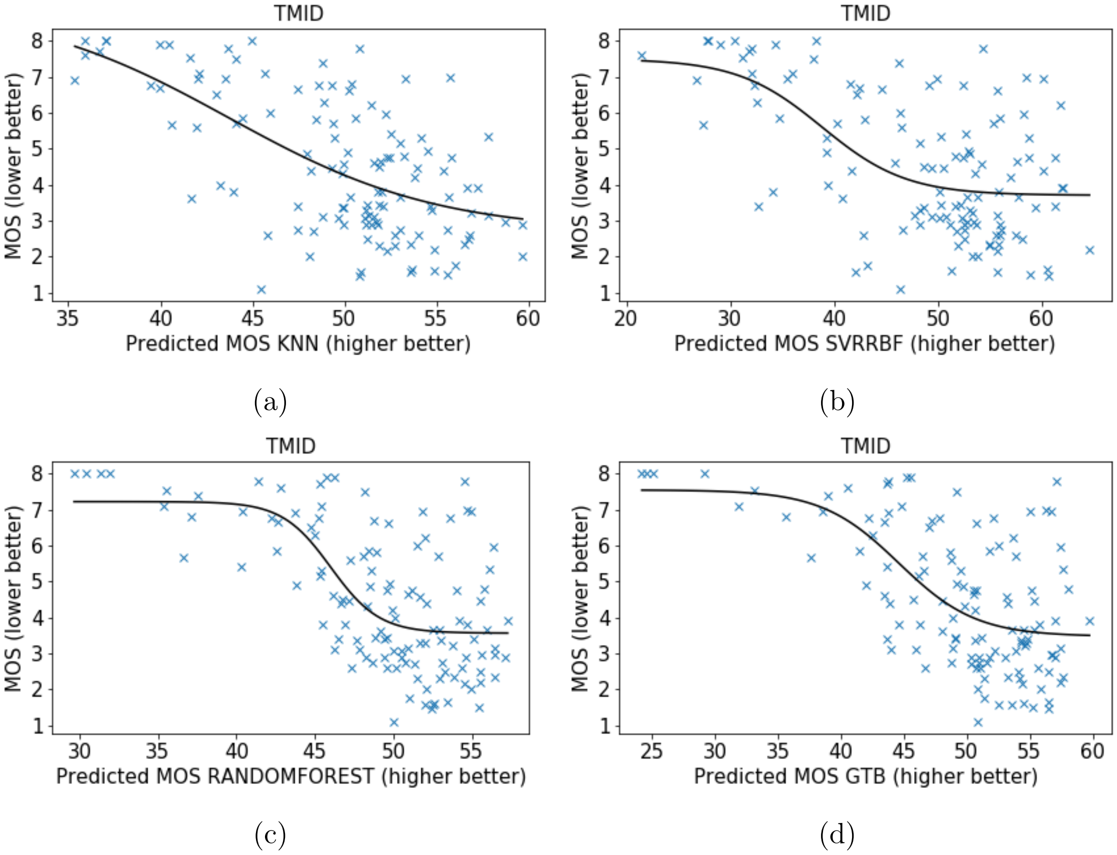
individual IQA metric performance in the TMID dataset (BTMQI), although the regression models use raw features from such metric. This suggests that the TMIQA raw features are tailored for specific datasets. For example, the BTMQI raw features are adequate for the TMID dataset, which explains the regression model performance gain in this dataset when compared to the respective performance values reported in Section 3.1 for the first experiment. The HIGRADE-1 and HIGRADE-2 raw features may not be as appropriate for this dataset as the BTMQI raw features, such that, in this case, the raw features from both HIGRADE metrics act more as noisy inputs for the regression models. This causes the trained regression models, which use raw features from all TMIQA metrics, to perform worse than the BTMQI metric. The same reasoning applies to the regression model training results presented in Table 3.6. In the ESPL-LIVE dataset, HIGRADE raw features seem to be more suitable than BTMQI raw features.

Figure 3.6 shows the corresponding scatter plots considering the TMID dataset. The regression models achieve better performance values as compared to their counterparts in the first experiment, reported in Section 3.1. For the Random Forest and GTB models, the corresponding scatter plots are now a little less dispersed and show a general decreasing trend, that is, higher predicted MOS (better quality) are more associated with lower subjective MOS (better quality). Ideally, points in the scatter plots would be arranged along on a line with slope analogous to -1, thus representing perfect correlations between subjective and predicted MOS.

The regression models trained in this experiment distinguish between general “good quality” and “bad quality” examples, and the models are particularly able to recognize extreme “bad quality” examples. To illustrate this, Figure 3.7 shows the three best and worst images predicted by the Random Forest model in the TMID dataset.

Table 3.7: Regression model performance values in the TMID dataset, in the second experiment.

Regression Model	**PLCC**	**SRCC**	**RMSE**
SVR	0.597	0.465	1.543
KNN	0.631	0.548	1.493
Random Forest	**0.650**	**0.550**	**1.462**
GTB	0.611	0.512	1.522



(a)

(b)

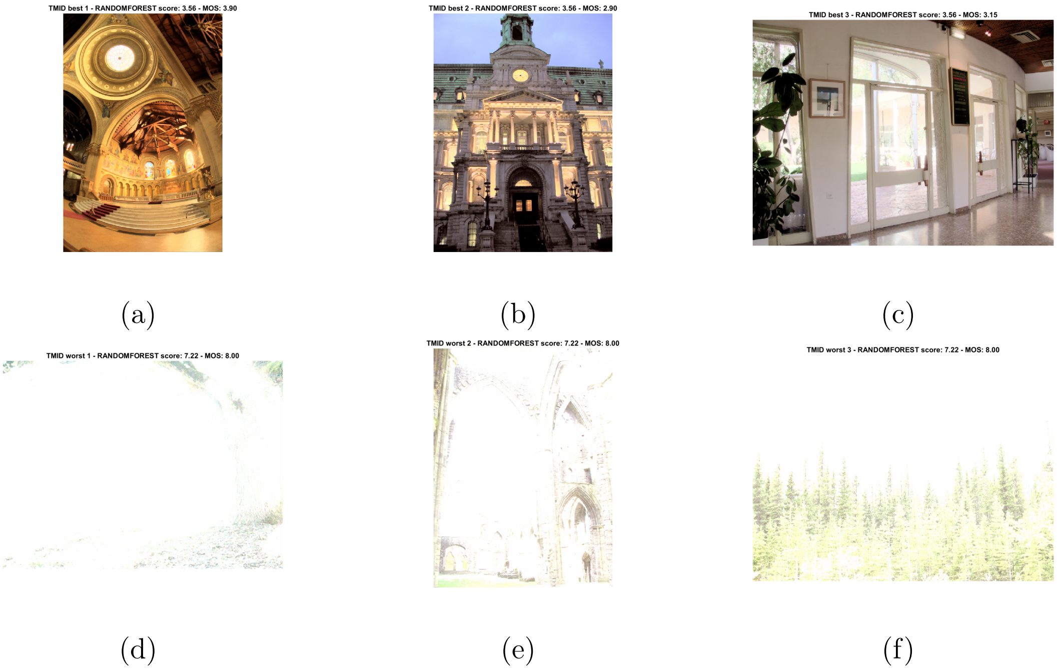
(c)

(d)

Figure 3.6: Scatter plots of the subjective MOS versus predicted MOS from each regression model considering the samples from the TMID dataset, in the second experiment. (a) KNN, (b) SVR, (c) Random Forest, and (d) GTB. The black line shows the non-linear function fit to data.

3.3 Experiment 3: IQA Metric Scores as Features for Classification Models

The third experiment consists in transforming the regression problem into a classification problem. The motivation for this experiment is based on the observation that regression models trained in the second experiment are apparently capable of distinguishing between overall good and bad tone-mapping examples. In this third experiment, we repeat exactly the two previous experiments, but replacing MOS


TMID best 1 - RANDOMFOREST score: 3.56 - MOS: 3.90
TMID best 2 - RANDOMFOREST score: 3.56 - MOS: 2.90
TMID best 3 - RANDOMFOREST score: 3.56 - MOS: 3.15
TMID worst 1 - RANDOMFOREST score: 7.22 - MOS: 8.00
TMID worst 2 - RANDOMFOREST score: 7.22 - MOS: 8.00
TMID worst 3 - RANDOMFOREST score: 7.22 - MOS: 8.00

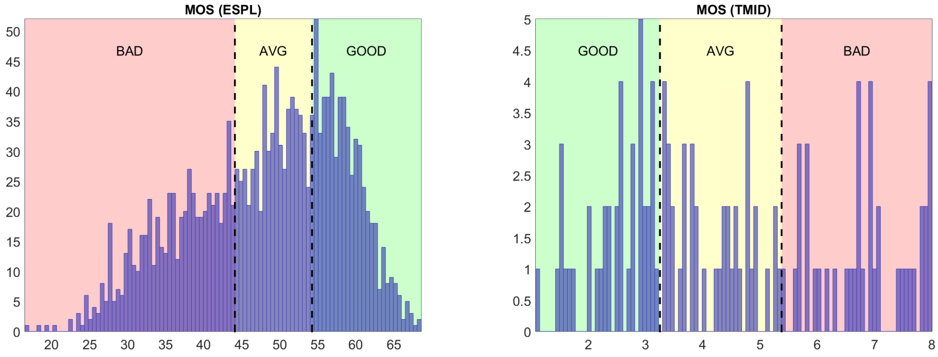
(a) (b) (c)

(d) (e) (f)

Figure 3.7: Three images with the best quality (top row) and three images with the worst quality (bottom row) predicted by the Random Forest model in the TMID dataset. Scores range from 1 (best quality) to 8 (worst quality). For each image, the predicted MOS value is reported, along with the subjective MOS value, in parenthesis: (a) 3.56 (3.90); (b) 3.56 (2.90); (c) 3.56 (3.15); and (d), (e) and (f) 7.22 (8.00).

values by labels. We use three labels to categorize the corresponding image quality as “bad” (label “0”), “average” (label “1”), or “good” (label “2”). Classification models are then trained to predict such labels. MOS categorization is performed as follows. Considering all MOS values from a dataset, we calculate the MOS values that correspond to the 33th and 66th percentiles (p_{33} and p_{66}, respectively), and use such values as thresholds to divide the MOS distribution into three regions. This ensures that each region contains roughly the same number of samples. Labels are assigned to each sample according to the region into which the associated MOS value falls. Figure 3.8 depicts the MOS distributions of each dataset divided into three regions.

For this experiment, we use accuracy as the performance metric to evaluate the classification models. Before training the models, we analyze how well the individual IQA metrics perform in the classification task. For each IQA metric, we transform the corresponding scores into labels using the same procedure described in the previous paragraph for the MOS values, and then calculate the respective metric accuracy. As pointed out in Chapter 2, to report the accuracy of TMIQA metrics in their original datasets (*i.e.* HIGRADE metrics in ESPL-LIVE dataset, and BTMQI metric in TMID dataset), part of the respective dataset is used to retrain the metric, and the other part is used to test it. Scores from the test part are then labeled, and accuracy is calculated. The corresponding TMIQA metric



(a)

(b)

Figure 3.8: MOS distributions divided into three quality regions: (a) ESPL-LIVE ($p_{33} = 43.9$ and $p_{66} = 54.0$), and (b) TMID ($p_{33} = 3.2$ and $p_{66} = 5.3$) datasets.

is retrained 1000 times, using a different training/test split each time. Table 3.8 reports the median accuracy of each IQA metric obtained from the 1000 test sets. These results serve as baseline comparison for the classification model results.

The third experiment results are presented in two parts. In the first part, classification models are trained using IQA metric scores as features, and model design follows the same procedure outlined for the regression models reported in Section 3.1 (that is, retraining TMIQA metrics in a part of ESPL-LIVE dataset, selecting IQA metrics with SFS method, and tuning hyperparameters using grid search and 10-fold cross-validation). For cross-validation, the different labels are represented by the same number of samples in each subset, thus preserving the label distribution observed in the entire dataset. In the second part, classification models are trained using the TMIQA raw features.

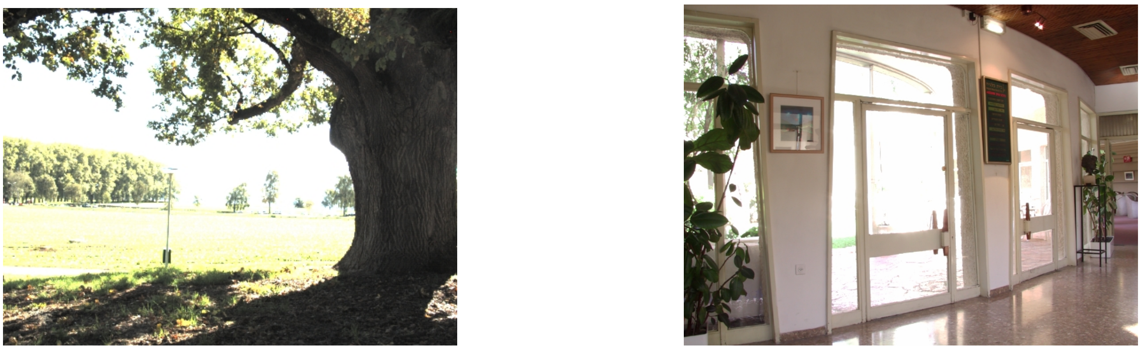
Table 3.9 shows the results from the classification models in the TMID dataset, along with the IQA metrics that are selected as features during training for each model. The corresponding confusion matrices are reported in Figure 3.10. Regarding overall accuracy, training classification models that use scores from multiple IQA metrics as features does not provide performance improvement over the BTMQI metric performance, which is the best one among individual IQA metrics in TMID dataset (with median accuracy of 0.667, as reported in Table 3.8). All classification models have similar accuracy values in TMID dataset. The Random Forest model is the one with the highest median accuracy value of 0.450. For discriminating between overall “good” and “bad” tone-mapped image quality, these classification models have moderate performance. In general, half of the samples from the “good quality” and “bad quality” groups are correctly classified. For the “bad quality” group in particular, classification models tend to overlook the negative impact that over-exposed areas have on image quality, as long details are visible in other parts of the scene. Figure 3.9 illustrates these misclassifications.

Table 3.8: IQA metric performance values in the ESPL-LIVE and TMID datasets, in the classification task. Each column shows the median accuracy (ACC) calculated from the 1000 accuracy values in the test sets. Boldface text indicates the best accuracy in the corresponding dataset.

IQA Metric	**ACC (ESPL-LIVE)**	**ACC (TMID)**
BIQI	0.374	0.261
BLIINDS-2	0.332	0.478
BPADA	0.393	0.522
BRISQUE	0.363	0.565
BTMQI	0.468	**0.667**
CDIIVINE	0.380	0.391
CORNIA	0.416	0.348
DESIQUE	0.366	0.522
DIIVINE	0.357	0.391
ENIQA	0.427	0.348
FRIQUEE	0.233	0.261
GM-LOG	0.319	0.304
HIGRADE-1	**0.588**	0.435
HIGRADE-2	0.580	0.435
ILNIQE	0.393	0.391
NFERM	0.374	0.435
NIMA	0.360	0.478
NIQE	0.371	0.565
NIQMC	0.479	0.522
NJQA	0.327	0.565
PAM	0.381	0.348
RANKIQA	0.413	0.304
SISBLIM-SM	0.380	0.304
SISBLIM-SFB	0.258	0.435
SISBLIM-WM	0.382	0.261
SISBLIM-WFB	0.269	0.391
SSEQ	0.368	0.522

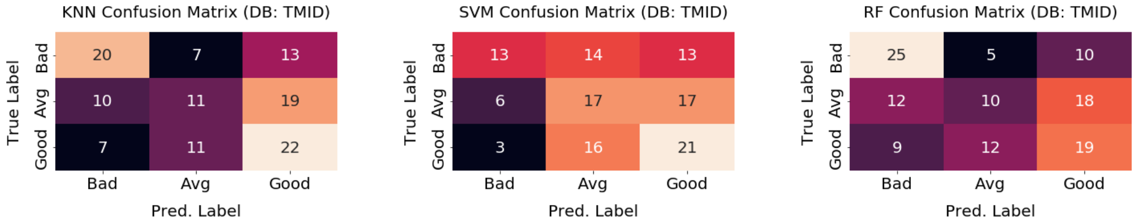
Table 3.9: Classification model performance values in the TMID dataset, along with the selected IQA metrics for each model. Boldface text indicates the best result.

Classification Model	**Selected IQA Metrics**	**ACC**
KNN	‘HIGRADE-2’, ‘NIQMC’, ‘BTMQI’, ‘ENIQA’, ‘DESIQUE’	0.442
SVM	‘HIGRADE-2’, ‘BTMQI’, ‘ENIQA’, ‘SISBLIM-WFB’	0.425
Random Forest	‘HIGRADE-1’, ‘BTMQI’, ‘ENIQA’	**0.450**



(a) (b)

Figure 3.9: “Bad quality” tone-mapping examples incorrectly classified as “good quality” by the models. The models seem to neglect the negative impact of over-exposed areas in image quality (*e.g.* sky and grassland in (a), and outdoors in (b)), as long as details are visible in other regions (*e.g.* tree trunk and leaves in (a), and the plants inside the room and the ceiling in (b)).



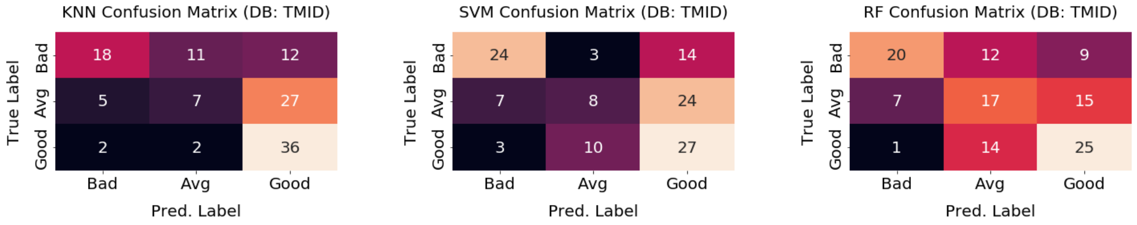
(a) (b) (c)

Figure 3.10: Confusion matrices from each classification model (using IQA metric scores as features) in the TMID dataset: (a) KNN, (b) SVM, and (c) Random Forest.

Table 3.10 shows the performance values of classification models trained with the TMIQA raw features, and Figure 3.11 shows the corresponding confusion matrices. As observed with the regression models, classification models trained directly with TMIQA raw features achieve overall performance values that are slightly better than their counterparts that use IQA scores as features. The best performance value achieved in this experiment (Random Forest model, with median accuracy of 0.517) is still not better than the median accuracy value of the BTMQI metric in the TMID dataset. Also, performance values are similar among the classification models. Particularly, these models tend to predict more samples as “good” tone mapping. This causes their performance to improve in the “good quality” group, at the cost of mistakenly classifying more samples from the “average quality” group. Considering the three quality groups, classification models trained with either IQA scores or TMIQA raw features tend to have the worst performance values when categorizing samples from the “average quality” group.

Table 3.10: Performance values, in the TMID dataset, of classification models that use TMIQA raw features.

Classification Model	ACC
KNN	0.508
SVM	0.492
Random Forest	**0.517**



(a) (b) (c)

Figure 3.11: Confusion matrices from each classification model (using TMIQA raw features) in the TMID dataset: (a) KNN, (b) SVM, and (c) Random Forest.

3.4 Experiment 4: Regression Models for TMIQA Trained in Mixed Datasets

In the previous experiments, regression and classification models are trained in ESPL-LIVE dataset only, and then tested in TMID dataset. Poor test performance values suggest that ESPL-LIVE and TMID samples may have different features that are important for quality assessment. In this fourth experiment, we create training and test datasets that contain, each one, samples from both ESPL-LIVE and TMID datasets. This is in contrast with previous experiments, in which samples from a single dataset are used for training, and samples from another different dataset are used for testing. We follow the same procedure from experiments one and two, but now perform regression model training and testing in the corresponding mixed datasets. First, IQA metric scores are considered as regression model features. Afterwards, TMIQA metric features as used as regression model features. Next, we explain how ESPL-LIVE and TMID datasets are mixed.

As shown in Figure 3.2, the original regression model dataset partitions considered 362 ESPL-LIVE samples for training, and 120 TMID samples for testing. We preserve the same number of samples in training and testing datasets, except that now we swap 60 TMID samples for 60 ESPL-LIVE samples between datasets. Figure 3.12 illustrates the new regression model training and testing sets. We take care to ensure that the entire quality range defined in TMID dataset is represented in TMID samples used for training, as well as in TMID samples used for testing. This avoids selecting only samples from one quality range ("good", for instance) for

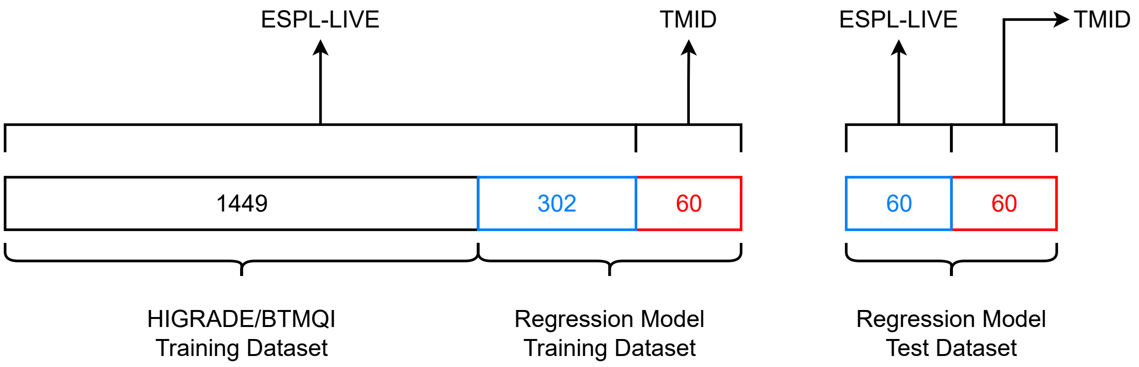
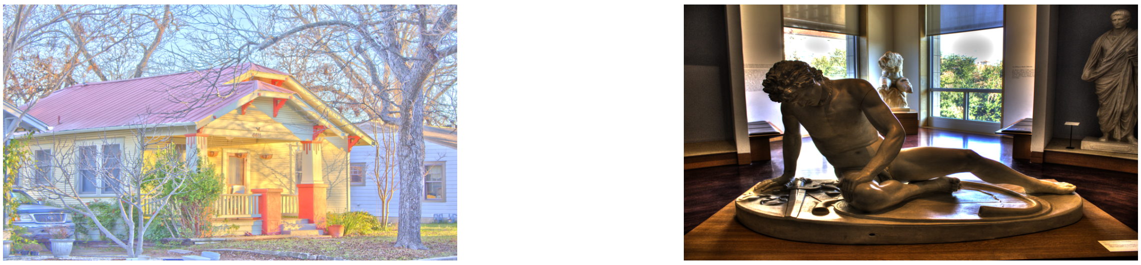

ESPL-LIVE
TMID
ESPL-LIVE
TMID
1449
302
60
60
60
HIGRADE/BTMQI Training Dataset
Regression Model Training Dataset
Regression Model Test Dataset

Figure 3.12: Dataset partition for training and testing the regression models, now considering that TMID and ESPL-LIVE samples are mixed.

training and only samples from another quality range (“bad”, for instance) for testing. Same care is taken when considering ESPL-LIVE samples that are used for training and testing.

To aid the selection of swapping samples, we follow a procedure similar to the one adopted in the third experiment presented in the previous section. We categorize sample quality into three groups: “Good”, “Average”, and “Bad”. Unlike the previous experiment, we perform such classification according to our own subjective quality impressions. The idea here is that sample subjective selection yields a better representation of different quality groups, as opposed to using “hard” numeric threshold values that organize samples into such quality groups (such as the ones defined in Figure 3.8). Our quality criteria are mainly based in two aspects. The first one is scene detail visibility, that is, whether visual content is perceived in dark and bright regions simultaneously. The second one is whether scene colors look natural. The categorization is performed to the 362 ESPL-LIVE samples and the 120 TMID samples.

Next, we illustrate some few cases where the classification based on our subjective criteria differs from sample quality categorized strictly according to MOS values. Figure 3.13 shows examples in ESPL-LIVE dataset where our subjective quality impressions do not agree with the MOS values. In 3.13(a), the sample has low MOS value (“Bad” quality), but we classified it as “Good” quality. We believe that the scene has a low MOS value because colors are a little over-saturated, which make them seem artificial. We do not consider such colors unnatural and classified this scene as “Good” quality because its content is entirely visible with no noticeable distortions. In 3.13(b), the sample has high MOS value (“Good” quality), but we classified it as “Bad” quality. We justify our choice because the scene contains perceivable halo artifacts (for instance, outside the windows and along the edges of walls behind the central sculpture) and over-exposed regions in which details are not seen (such as outside the windows and part of the floor behind the central



(a) (b)

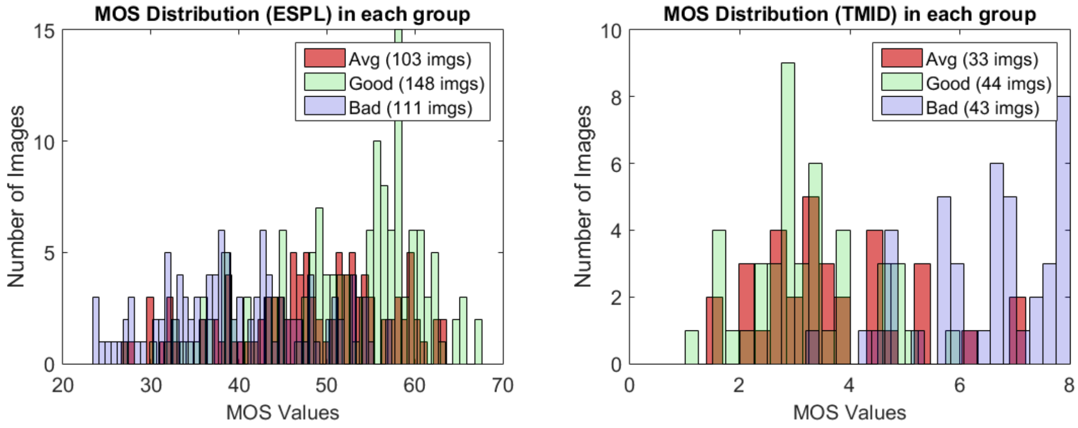
Figure 3.13: Examples of ESPL-LIVE samples in which our subjective criteria do not agree with MOS values. In (a), MOS value is 30.2 ("Bad" quality), but we classified it as "Good" quality. In (b), MOS value is 57.3 ("Good" quality), but we classified it as "Bad" quality. Explanations of quality assessment disagreements for such cases are provided in text.

sculpture). We hypothesize that the image has a high MOS value because the aforementioned distortions are not present in the scene main content, which are the sculptures. Therefore, for many viewers, such distortions may go unnoticed or not really bother.

Figure 3.14 shows the MOS distributions of samples from each quality group, after classifying them according to our quality impressions. We see that in ESPL-LIVE dataset, generally, we assign to "Good" quality group samples with higher MOS values (which, indeed, denote better quality), and to "Bad" quality group samples with lower MOS values (which denote worse quality). Samples assigned to "Average" quality group contain average MOS values, with slightly more samples having higher scores than lower scores. Similar conclusions can be drawn for samples from TMID dataset, in which higher scores represent worse quality, and lower scores represent better quality. Overall, our subjective separation is consistent with sample MOS values from each dataset.

When selecting the 60 ESPL-LIVE and 60 TMID samples that are swapped between each other, we aim at preserving, in each subset, the same sample quality proportion (according to our subjective categorization) observed in the original regression model training and testing datasets, respectively. Table 3.11 summarizes the sample quality proportions and the associated number of samples in the original datasets and in the respective new subsets. After creating the mixed training and test datasets, we map the TMID MOS value range (1 - best quality to 8 - worst quality) into the ESPL-LIVE MOS value range (70 - best quality to 15 - worst quality). This ensures that all samples in the mixed datasets have MOS values in the same value range. We use a linear mapping[4] for this purpose, expressed as follows:

[4] We also tried a more sophisticated algorithm that maps MOS values from different subjective experiments into a common score scale, called Iterated Nested Least-Squares Algorithm (INSLA) [60]. Such mapping did not lead to better regression model training nor test performance.



(a) (b)

Figure 3.14: Sample MOS distribution from (a) ESPL-LIVE and (b) TMID datasets, after grouping samples according to subjective impression. Green bars denote samples subjectively classified as "Good", red bars denote samples subjectively classified as "Average", and blue bars denote samples subjectively classified as "Bad". In ESPL-LIVE, higher scores correspond to better quality, whereas, in TMID, lower scores correspond to worse quality. Subjective division is consistent with sample MOS values in both datasets.

Table 3.11: Sample quality proportions observed in original regression model training and test datasets, and in sample subsets that comprise the new regression model mixed training and test datasets.

Dataset/Subset	**Number of Samples**	**"Good" Quality**	**"Avg" Quality**	**"Bad" Quality**
ESPL-LIVE (Original Training Dataset)	362	148 (41%)	103 (28%)	111 (31%)
ESPL-LIVE Training Subset	302	123 (41%)	87 (29%)	92 (30%)
ESPL-LIVE Test Subset	60	25 (42%)	16 (27%)	19 (31%)
TMID (Original Test Dataset)	120	44 (36%)	33 (28%)	43 (36%)
TMID Training Subset	60	21 (35%)	18 (30%)	21 (35%)
TMID Test Subset	60	23 (38%)	15 (25%)	22 (37%)

$$\widehat{\text{TMID}}_{MOS} = \frac{1}{7} \times (-55 \times \text{TMID}_{MOS} + 545). \tag{3.1}$$

In Equation 3.1, $\widehat{\text{TMID}}_{MOS}$ denotes the TMID sample MOS value in the common score range (*i.e.* the one from ESPL-LIVE), and TMID_{MOS} corresponds to the original TMID sample MOS value. We then train and test four types of regression models in the corresponding mixed datasets, namely KNN, SVR, Random Forest and

Table 3.12: Regression model performance values in the mixed training dataset, considering selected IQA metric scores as training features. Each performance metric column shows the median of the values obtained from the ten cross-validation test subsets. Boldface text indicates the best results for each performance metric.

Regression Model	**Selected IQA Metrics**	**PLCC**	**SRCC**	**RMSE**
SVR	‘HIGRADE-1’, ‘BTMQI’ ‘NJQA’, ‘RANKIQA’	0.735	0.758	7.755
KNN	‘HIGRADE-1’, ‘BTMQI’, ‘NJQA’, ‘BRISQUE’	0.745	0.765	7.740
Random Forest	‘HIGRADE-1’, ‘BTMQI’, ‘NJQA’, ‘NFERM’, ‘NIQE’, ‘BIQI’, ‘CORNIA’	**0.767**	**0.776**	**7.379**
GTB	‘HIGRADE-1’, ‘BTMQI’, ‘NJQA’, ‘BPADA’, ‘SISBLIM-WFB	0.734	0.761	10.725

GTB. First, we use quality scores from some IQA metrics (selected via SFS method) as model features, similarly as the first experiment presented in this chapter.

Table 3.12 shows the regression model training performance values, along with the corresponding IQA metric scores that are selected as features in each model. These performance values correspond to median values computed from test sets of a 10-fold cross-validation. Comparing Table 3.12 with Table 3.3, we note that regression models trained in the mixed dataset have different (and smaller) feature sets than their counterparts trained in ESPL-LIVE only. This indicates an influence of TMID samples in training, and suggests that TMID and ESPL-LIVE samples have different attributes that are relevant for quality assessment. These results also show that only few IQA metrics are suited for evaluating quality of samples from both datasets, as observed by the smaller feature sets. All regression models present similar performance values.

Table 3.13 shows the regression model performance values in the mixed test dataset, and Figure 3.15 shows the associated scatter plots. As comparison baselines, we train the TMIQA metrics (HIGRADE-1, HIGRADE-2 and BTMQI) in the same mixed training dataset used for the regression models. We also present in Table 3.13 their performance in the mixed test dataset. We point out that these retrained versions are different from the ones trained in the 1449 ESPL-LIVE samples (Figure 3.12). The ones trained in ESPL-LIVE only are applied to mixed training dataset samples, and their corresponding quality scores serve as potential features (in case they are selected via SFS method) for the regression models, as shown in Table 3.12.

From Table 3.13, we observe that none of the regression models outperform the retrained HIGRADE-1 metric. Apart from the Random Forest model, all other regression models have worse performance values than any other retrained individual TMIQA metric considered. This indicates that using IQA scores as features is

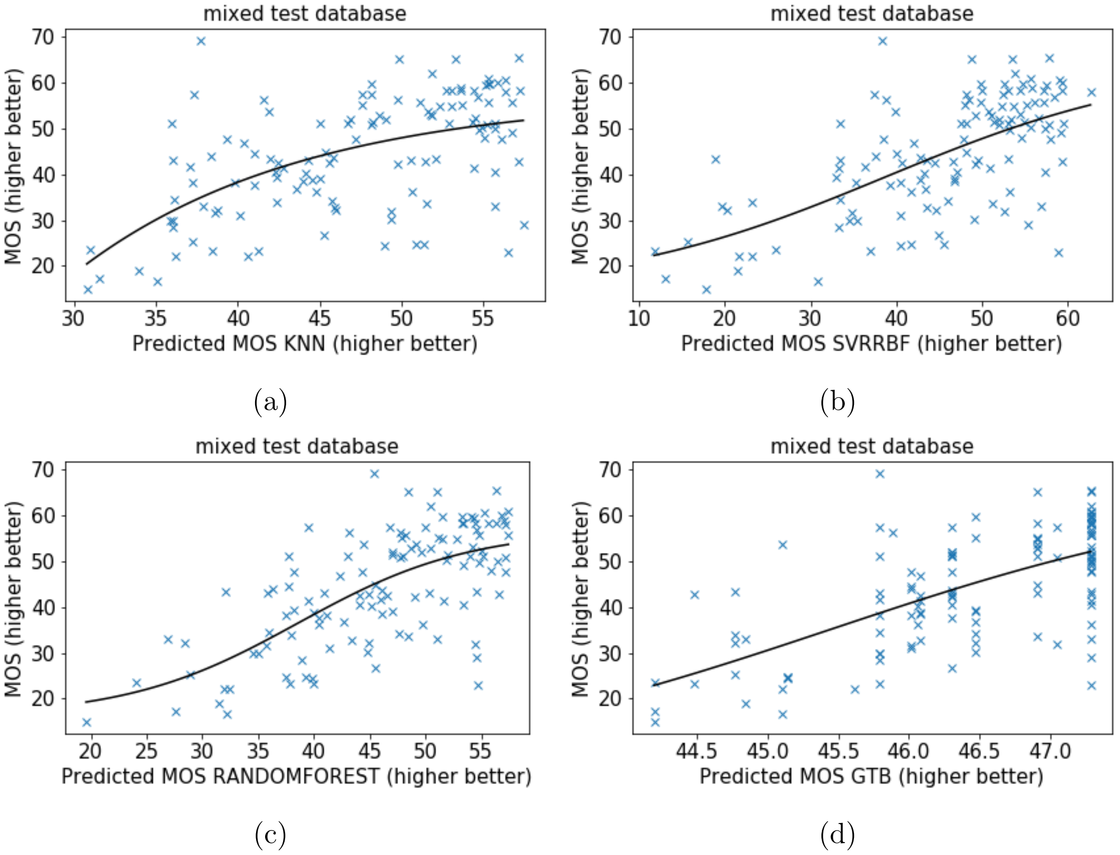
Table 3.13: Regression model performance values in the mixed test dataset, considering selected IQA metric scores as features. TMIQA metrics presented in this table (HIGRADE-1, HIGRADE-2 and BTMQI) are trained in the same mixed training dataset used for the regression models.

Regression Model	**PLCC**	**SRCC**	**RMSE**
HIGRADE-1	**0.714**	**0.680**	**8.847**
HIGRADE-2	0.671	0.607	9.370
BTMQI	0.678	0.628	9.289
GTB	0.641	0.595	9.701
KNN	0.573	0.532	10.358
Random Forest	0.684	0.641	9.217
SVR	0.640	0.600	9.712

less reliable to assess any generic tone-mapped image quality than using features extracted by TMIQA metrics from image attributes. Limited generalization capability from IQA scores likely reflects the observation that most IQA metrics, either used individually or collectively as a "commitee", are not reliable quality predictors for tone-mapped images.

The second experiment involving the mixed datasets consists in training the same regression models as the previous experiment, except that features extracted from each TMIQA metric ("raw" features) are now directly used to train the regression models. Table 3.14 shows the training median performance values of each regression model, considering performance values computed from the ten cross-validation test subsets. We observe that the SVR presents slightly better performance values than the Random Forest model, and the KNN model has the worst performance Even though the GTB model also presents better correlation values than the Random Forest model, from Figure 3.15(d) we note its predictions are actually not reliable in this experiment. Its quality scores span a very small range and they barely vary, as the same score values are given to lots of different samples. This indicates some problem occurred during this model training. Further investigation is required to pinpoint the causes of this problem, so that it can be corrected.

Table 3.15 shows the performance values of the same regression models in the mixed test dataset. Figure 3.16 shows the corresponding scatter plots considering sample scores from the mixed test dataset. Comparing Table 3.15 with Table 3.13, we observe that, apart from the Random Forest model, all regression models have improved their performance values over the ones from their counterparts in the previous experiment. Comparing Figure 3.16 with Figure 3.15, we note that corresponding model scatter plots are slightly less dispersed when "raw" features are used to train the models. This reinforces the idea that such "raw" features are more generic and reliable features for TMIQA than IQA scores.



(a)

(b)

(c)

(d)

Figure 3.15: Scatter plots of the subjective MOS versus predicted MOS from each regression model considering the samples from the mixed test dataset and selected IQA metric scores as the regression model features. (a) KNN, (b) SVR, (c) Random Forest, and (d) GTB. The black line shows the non-linear function fit to data.

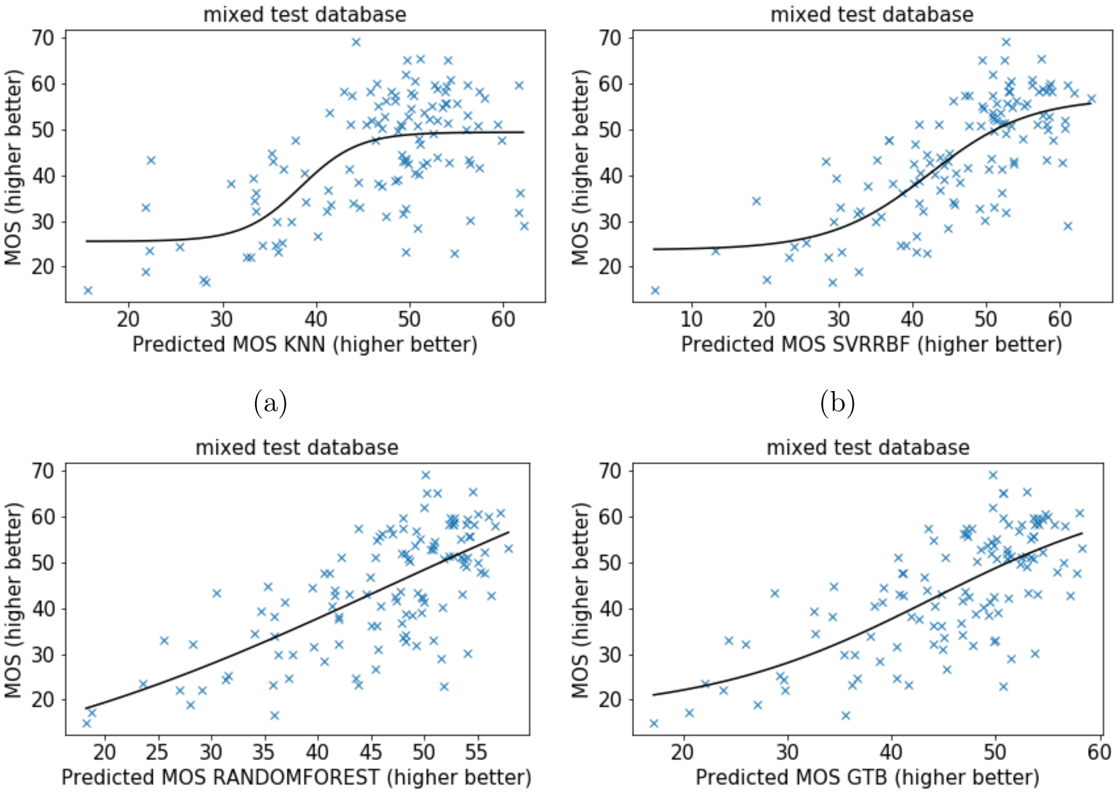
Table 3.14: Regression model performance values in the mixed training dataset, considering "raw" TMIQA features as training features. Each performance metric column shows the median of the values obtained from the ten cross-validation test subsets. Boldface text indicates the best results for each performance metric.

Regression Model	**PLCC**	**SRCC**	**RMSE**
SVR	0.673	0.671	8.465
KNN	0.518	0.556	9.838
Random Forest	0.630	0.641	8.394
GTB	**0.696**	**0.682**	**8.259**

In particular, the SVR model presents the highest performance improvement, and is now the best-performing metric. The HIGRADE-1, HIGRADE-2, and BTMQI metrics also use an SVR model that is trained with their own respective "raw" features. The SVR model we train in this experiment combines the "raw" features from the three aforementioned TMIQA metrics. Such combination is probably the cause of the slight performance improvement from this SVR over the SVR model from individual TMIQA metrics, as shown in Table 3.15.

Table 3.15: Regression model performance values in the mixed test dataset, considering "raw" TMIQA features as model features. TMIQA metrics presented in this table (HIGRADE-1, HIGRADE-2 and BTMQI) are trained in the same mixed training dataset used for the regression models.

Regression Model	**PLCC**	**SRCC**	**RMSE**
HIGRADE-1	0.714	0.680	8.847
HIGRADE-2	0.671	0.607	9.370
BTMQI	0.678	0.628	9.289
GTB	0.695	0.658	9.084
KNN	0.622	0.462	9.891
Random Forest	0.683	0.636	9.235
SVR	**0.741**	**0.705**	**8.484**



(a)

(b)

(c)

(d)

Figure 3.16: Scatter plots of the subjective MOS versus predicted MOS from each regression model considering the samples from the mixed test dataset and "raw" TMIQA features as the regression model features. (a) KNN, (b) SVR, (c) Random Forest, and (d) GTB. The black line shows the non-linear function fit to data.

3.5 Chapter Summary

In this chapter, we presented four experiments in which we used regression and classification models based on machine learning algorithms to assess quality of tone-mapped images. In the first one, regression models were trained in ESPL-LIVE dataset, using quality scores from different IQA metric combinations as features, and tested in TMID dataset. In the second experiment, regression models were trained in ESPL-LIVE using the features extracted by the TMIQA metrics (“raw” features), namely HIGRADE-1, HIGRADE-2 and BTMQI. They were then tested in TMID. In the third experiment, we changed the regression problem to a classification problem, and trained classification models using the same methodology defined in the first and second experiments (*i.e.* combination of IQA metric quality scores as training features, and “raw” TMIQA features as training features). In the fourth experiment, we repeated the first and second experiments, but trained and tested regression models using a mix of samples from ESPL-LIVE and TMID datasets.

Based on this chapter results, two main conclusions can be drawn. First, models trained with “raw” TMIQA features perform better than models trained with quality scores from multiple IQA metrics as features. Combining quality scores from different IQA metrics yields poor performance values probably because most of the considered metrics output contradicting quality predictions for the same tone-mapped image. The second conclusion is that mixing samples from ESPL-LIVE and TMID datasets for training and testing the models leads to better quality predictions than using only one dataset (ESPL-LIVE) for training, and only another dataset (TMID) for testing. This indicates that TMID and ESPL-LIVE samples contain different features that are relevant for TMIQA. This also suggests that none of these datasets should be used alone to train the regression models, as some tone-mapping distortions may not be present in any of their samples at all. Even combining both datasets, it is also possible that other tone-mapping distortions are missing, or present in only a few samples from these datasets. This most likely affects model capability of providing reliable quality predictions for samples that do not come from these datasets.

To overcome this limitation, in the next chapter, we introduce a new database that we assembled for the TMIQA application. This database contains approximately 175000 samples with their qualities labeled, and can be used to train models based on either machine learning or deep learning approaches. The samples comprise HDR scenes, depicting different scenarios, that are mapped by different TMOs. We aim at capturing as many tone-mapping distortions as possible, so that models trained in this dataset provide more reliable quality predictions to generic tone-mapped images (*i.e.* images that do not belong to specific datasets).

Chapter 4

Patch-Based Tone-Mapped Image Database

The results in Chapter 2 indicate that individual IQA metric performance values vary according to the test dataset. The results in Chapter 3 suggest that the trained regression model performance depends on the dataset from which the test samples originate. This limitation is probably because the raw features used by TMIQA metrics are tailored for sample images from one specific dataset. These features are handcrafted, and they may not capture all aspects that are important for assessing the quality of tone-mapped images in general. Deep learning models should be able to overcome such limitation, as they are able to automatically learn, from a large pool of samples, features that are relevant for the task they are trained to solve (in this case, TMIQA).

Although deep learning models have not yet been used for assessing the quality of tone-mapped images directly, they have been proposed in the literature for other IQA tasks. For example, three metrics used in this work are based on deep learning models that were trained originally for different IQA purposes: RankIQA (designed for quality assessment of images impaired by particular distortions, such as blur and white noise), NIMA, and PAM (both of which are designed for aesthetic quality assessment). These metric quality scores, which are used to achieve the results reported in Chapters 2 and 3, are obtained by applying the metrics to tone-mapped images directly. The deep learning models for RankIQA, NIMA, and PAM are not previously trained for the TMIQA task.

The lack of sufficiently large tone-mapped image datasets in which samples are labeled with subjective quality scores poses a challenge for training deep learning models for such task. In this chapter, we introduce a new tone-mapped image dataset that we assembled for this purpose. This dataset contains approximately 175000 samples, and is henceforth referred to as PBTDB (“Patch-Based Tone-mapped image DataBase”). Each sample is labeled by quality scores obtained from four TMIQA

Table 4.1: Summary of subjective experiments performed in a subset of 3009 PBTDB samples. Data in columns "Subject Group", "Number of Sessions" and "Samples per Sessions" are presented in Section 4.2.1. The average number of labels per sample, shown in the fourth column, for the private experiment is defined in Section 4.2.1. For the crowd experiment, this number is defined in Section 4.2.2, after removing some subjects by the "Gold Standard Questions" approach.

Subject Group	**Number of Sessions**	**Samples per Session**	**Avg. Labels per Sample**	**Gold Standard Questions?**
Private	12	250 or 251	16	No
Crowd	10	300 or 301	51	Yes

metrics: BTMQI, HIGRADE-1, HIGRADE-2, and TMQI. TMQI is an FR metric, unlike the other three metrics, which are NR metrics.

We also present in this chapter two subjective experiments that are performed in a subset of 3009 representative PBTDB samples, in order to assess image quality. Samples from such subset thus contain human subject scores besides TMIQA metric objective scores. Both experiments are performed remotely on an online platform (Section 4.2.1), and each one considers different groups of subjects. Table 4.1 summarizes the experiment setups. We detail the procedure used to verify whether subjective experiments performed in remote fashion are consistent and thus ensure that resulting labels are reliable. We conclude this chapter showing that one of the experiments is fully consistent, and observing similarities between subjective scores and objective TMIQA scores.

4.1 PBTDB Assembly

First, we collect HDR images from multiple sources publicly available in the internet. The HDR images correspond to different scene types, such as landscapes, urban scenarios, daytime and others. The HDR datasets are listed in Table 4.2. Overall, 789 HDR images are collected. Then, the combined HDR dataset is augmented by extracting a certain number of patches from each HDR image. Figure 4.1 illustrates the patch extraction procedure for a single HDR image, and how these patches are processed. More patches can be extracted from the HDR images if smaller patch sizes are used. However, with smaller patches it is also more difficult to ensure that they contain HDR luminance values. We tried three resolutions for extracting patches: 128×128, 256×256, and 512×512. We choose patch sizes of 256 × 256 because we obtain a fairly high number of HDR patches per HDR image. Also, this patch size is close to VGG-16 model[61] input image size, which is 224 × 224. The VGG-16 model is a well known deep learning architecture used for many image processing tasks, and is one of the deep learning models considered in this work, as

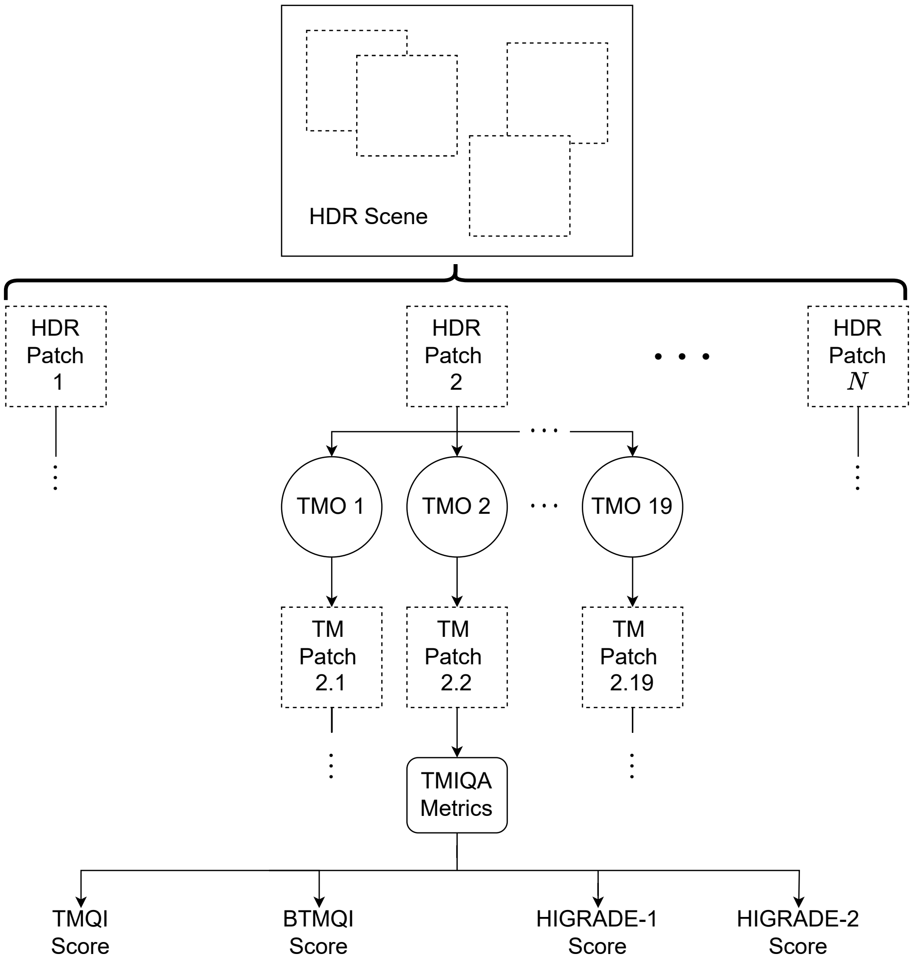


Figure 4.1: Patch extraction procedure and processing operations that are executed in order to assemble PBTDB. For clarity purpose, we depict the patch extraction procedure for a single HDR image, and the patch processing procedure for a single extracted HDR patch. N denotes the maximum number of extracted patches from a single HDR scene, and it varies according to the HDR scene (it can be, at most, 15).

discussed in Chapter 5.

Each patch is extracted from a random position in the image. In order to ensure that the extracted patch is an HDR patch, we use the following empirical criterion for determining the patch dynamic range DR[1]:

$$\text{DR} = \log_{10} \frac{\max(\mathcal{I})}{\text{prctile}(\mathcal{I}^*,\, 1)}. \tag{4.1}$$

In Equation (4.1), $\mathcal{I}$ denotes the set of pixel values from the input patch, max(.) is the function that returns the maximum value from the input set (in this case, the brightest pixel from the patch), and prctile(., j) is the function that returns the value that corresponds to the j-th percentile from the set of input values (in this case, the

[1] We use samples from Ward dataset to validate the empirical criterion, since this dataset reports each sample dynamic range.

Table 4.2: HDR image datasets from which HDR samples are taken and later tone-mapped, in order to generate the PBTDB samples.

Dataset	**Samples**
HDR Eye [63]	47
HDR Stanford [64]	88
Ward [65]	18
Funt HDR [66]	107
HDRI Haven [67]	341
HDR Photographic Survey [68]	105
sIBL [69]	52
EMPA HDR Dataset [70]	31
Total	**789**

1st percentile value from the non-zero patch pixel value set $\mathcal{I}^*$). The 1st percentile value is used, rather than the minimum non-zero pixel value, because it is a more reliable estimate of the minimum visible scene content, as observed empirically. Very low pixel values usually represent the camera sensor noise floor and, hence, do not constitute actual scene content [62]. The extracted patch is considered HDR if its dynamic range DR is higher than $\log_{10} 256$. This threshold value corresponds to the dynamic range covered by 8 bits. If the extracted patch is not an HDR patch, it is discarded, and a new patch is randomly selected and analyzed. The entire process is repeated until 15 HDR patches are chosen, or 100 consecutive extracted patches are not HDR patches. Patches that are too close to each other are not selected, in order to avoid extraction of very similar patches from the same region in the image. In the end of this process, 10171 patches are extracted, and they constitute the augmented HDR dataset.

Next, we apply 19 different TMOs from Banterle's HDR MATLAB Toolbox [71] to each sample from the augmented HDR dataset, thus yielding 193249 tone-mapped samples. In this work, we refer as "scene" to the content depicted in an original HDR patch prior to application of any tone-mapping algorithm, and as "sample" to a tone-mapped version of any scene. Each sample is labeled with the quality score given by four TMIQA metrics: HIGRADE-1, HIGRADE-2, BTMQI, and TMQI. After examining the quality score distribution from each metric, an anomaly was detected in the corresponding "low quality" score range parts. This anomaly, illustrated in Figure 4.2 for the HIGRADE-1 score distribution, is caused by outlier samples. These samples are not representative of "bad tone-mapping quality". They were generated for two reasons related to the methodology adopted for assembling the dataset. First, the patch extraction procedure may sometimes select patches that pass the HDR test defined in Equation (4.1), but present little visual content (e.g. very dark image areas with nearly no contrast variation, and

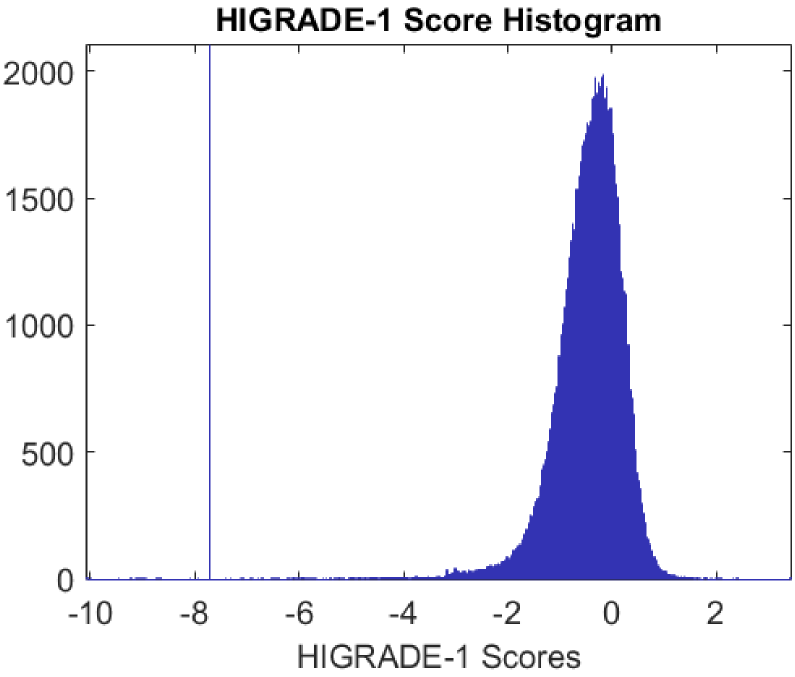


Figure 4.2: HIGRADE-1 score distribution in PBTDB, with outlier samples mostly represented by the tall bin (2102 samples) in the lower score range. This effect is also observed in the other metric score distributions.

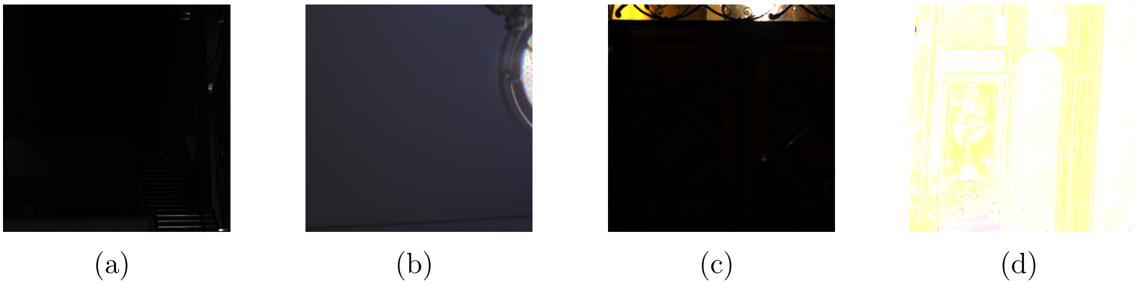


Figure 4.3: Examples of outlier samples removed from PBTDB. Very dark regions with nearly no contrast, and with small bright regions are observed in (a), (b), and (c). An example of over-saturated image is shown in (d).

with small bright regions that do not contain any details). Second, parameters of each TMO were set to their respective default values defined by the HDR Toolbox. This causes TMOs to generate some images with very little visual content (*e.g.* over-saturated or very dark images) because parameters are not calibrated for each image individually. Such samples are removed from the tone-mapped dataset. Figure 4.3 shows some removed samples. After removing the outlier samples, the final PBTDB dataset is obtained, with 175919 tone-mapped images. Figure 4.4 illustrates some samples from the dataset.

Figure 4.5 shows the score distributions from each metric in PBTDB. The BT-MQI score histogram is more dispersed than the histograms of other metrics, and it has a higher concentration around higher scores, which represent “bad” tone-mapping

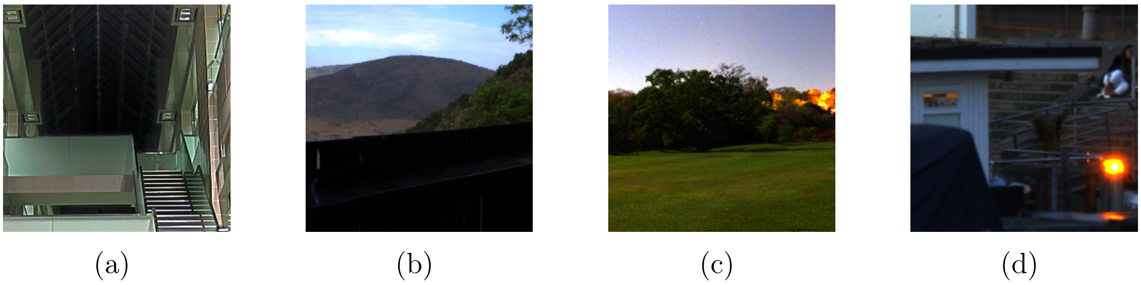

(a) (b) (c) (d)

Figure 4.4: Examples of samples within PBTDB. Different types of scenes are contained within PBTDB: (a) Indoors; (b) and (c) Landscapes; and (d) Urban Scenario.

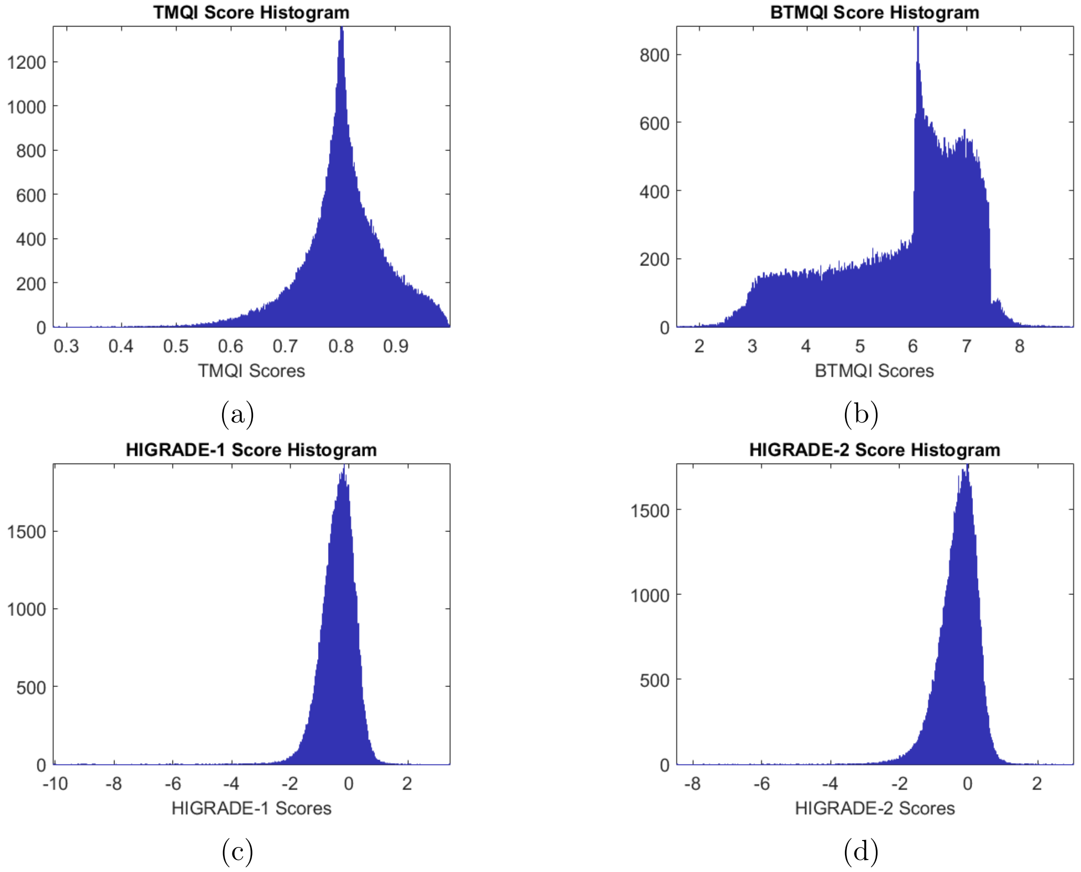
quality. The TMQI and both HIGRADE metrics have more similar distributions, which are more concentrated around an average value (0.8 for TMQI, and 0 for both HIGRADE metrics). Also, it is noted empirically, both in this work and in [15], that TMQI scores normally do not span the entire [0, 1] score range. They are rather distributed in the [0.6, 1] range. In this case, values around the TMQI histogram peak value of 0.8 observed in Figure 4.5 normally represent "average" tone-mapped image quality. For the HIGRADE metrics, the "average" quality is represented by values around 0.

Figure 4.6 shows the scatter plots between quality scores from each metric pair. Overall, the scatter plots show low correlation between the metric scores, except for the HIGRADE-1 and HIGRADE-2 metric pair, and BTMQI and TMQI metric pair. Particularly low correlations are observed between BTMQI and HIGRADE-1 quality scores, and between BTMQI and HIGRADE-2 quality scores. This strengthens the point that such metrics do not provide consistent quality predictions for images that do not originate from TMID dataset (for BTMQI metric), and ESPL-LIVE dataset (for both HIGRADE metrics). TMQI scores have higher correlation with BTMQI scores, than with HIGRADE-1 and HIGRADE-2 scores. This is probably because TMQI and BTMQI metrics analyze features that are somewhat related, although they are not the same. As reported in [12], TMQI scores achieve high correlations with subjective MOS from the TMID dataset[2], and this is also the case for BTMQI scores (see Table 2.2).

4.2 PBTDB Subjective Tests

The low correlation between objective TMIQA scores (Section 4.1) makes it difficult to define which one is a more reliable quality indicator. Furthermore, the

[2] In [12], the TMID dataset is used to test the TMQI metric performance only. TMQI has sensitivity parameters that define the relative importance of extracted features, before combining these features into a final quality score. According to [12], such parameters were optimized previously, using another tone-mapped dataset that is not considered in this work.



(a)

(b)

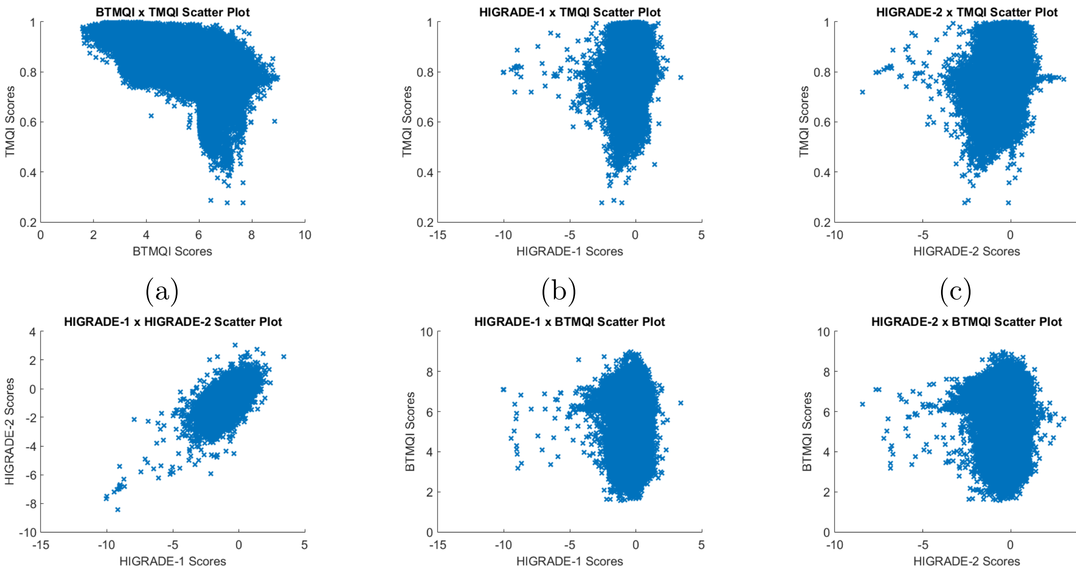
(c)

(d)

Figure 4.5: Quality score distributions from each TMIQA metric in PBTDB: (a) TMQI, (b) BTMQI, (c) HIGRADE-1, and (d) HIGRADE-2. For TMQI, HIGRADE-1, and HIGRADE-2 metrics, higher scores correspond to better quality, whereas for BTMQI metric, lower scores represent better quality.

available databases with subjective MOS, *i.e.* ESPL-LIVE and TMID, contain a small number of samples each. To assess overall tone-mapping quality and create a larger database, we perform subjective tests in a smaller subset of 3009 samples from PBTDB. Overall, 164 different scenes are represented by such subset samples. Up to 19 samples originate from one scene, each sample corresponding to a distinct tone-mapping operator. Images for this subset are chosen as representative samples from PBTDB. The term “representative” means the smaller subset mostly preserves each TMIQA score distribution from the entire PBTDB. Figure 4.7 shows each score distribution in the 3009 sample subset.

During sample selection, care is taken to ensure scene content variety in the subset. Scene content is categorized into one of three types: “Indoors” (enclosed areas, such as interior of buildings), “Landscape” (outdoor natural scenery only, such as mountains and forests) and “Urban” (outdoor scenes containing any buildings or objects made by humans, such as city streets and plazas). We aim at keeping an overall balance between the three scene types inside the subset. Figure 4.8 shows each scene type proportion in the subset. Besides type categorization, we check



(a) (b) (c)

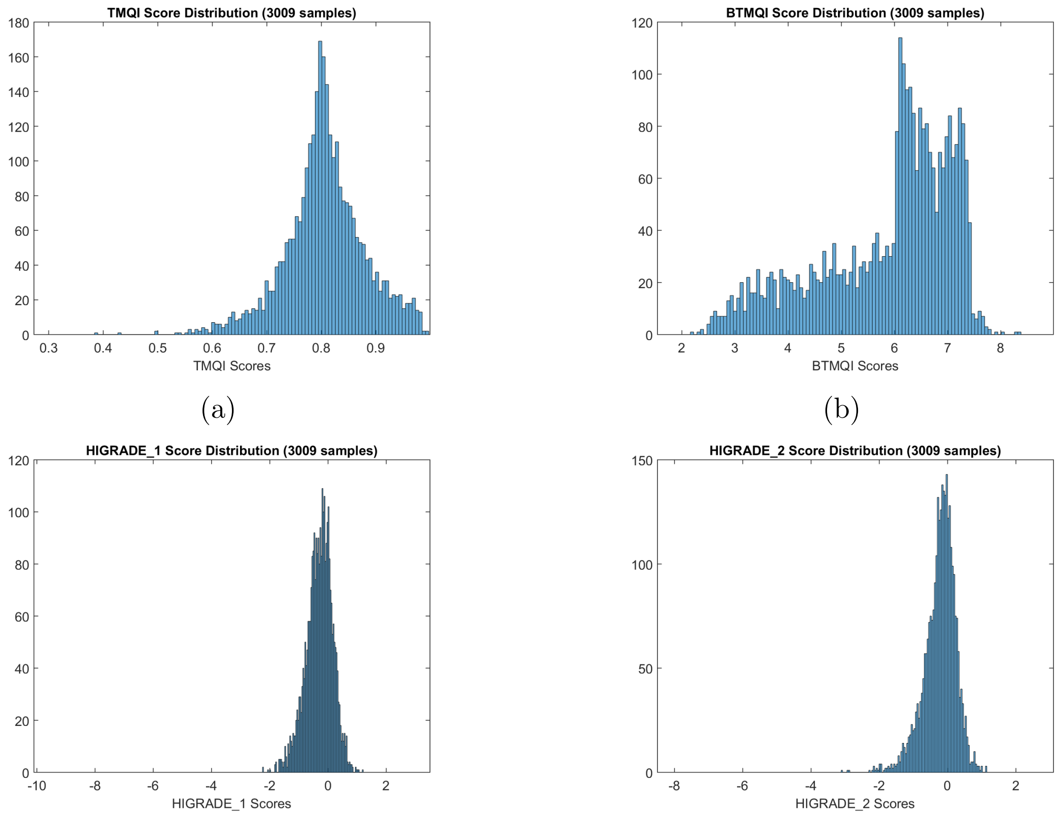
(d) (e) (f)

Figure 4.6: Scatter plots between quality scores from each metric pair in PBTDB. For HIGRADE-1, HIGRADE-2 and TMQI metrics, higher scores correspond to better quality, whereas for BTMQI metric, lower scores represent better quality. The PLCC values for each metric pair are: (a) -0.61, (b) 0.25, (c) 0.35, (d) 0.73, (e) -0.19, (f) -0.24

content diversity using three image attributes: colorfulness [16], spatial information [72], and image key [73]. Colorfulness indicates color diversity and intensity within a scene. Higher values indicate more vivid and different colors. We use the same colorfulness metric we adopted in our previous work [15]. Spatial information measures image complexity by computing the average energy from the image edge map. The image edge map is obtained from the application of the Sobel edge operator. Higher spatial information values indicate more edges and, thus, more structures and details in the scene. Image key corresponds to the average scene brightness, whose values range from 0 to 1. Higher values indicate brighter scenes. Figure 4.9 shows the scatter plots between each metric pair. We observe that the scatter plots are highly spread, which highlights the sample content diversity contained in the subset.

4.2.1 Subjective Experiment Setup

Subjective tests are conducted to label image qualities from the 3009 PBTDB sample subset. Because of COVID-19 pandemic restrictions, tests are performed remotely only. To manage the remote subjective tests, we use a service from Amazon Web Services (AWS), called “SageMaker Ground Truth”. This service offers functionalities that facilitate the preparation and execution of labeling tasks. The service is integrated with Amazon Mechanical Turk (AMT) platform, which allows perform-



(a)

(b)

(c)

(d)

Figure 4.7: Quality score distributions from each TMIQA metric in the 3009 sample subset from PBTDB: (a) TMQI, (b) BTMQI, (c) HIGRADE-1, and (d) HIGRADE-2. Each distribution is similar to its corresponding one shown in Figure 4.5.

ing such tasks using crowdsourcing. Crowdsourcing refers to tasks that demand some sort of input from a large number of people [74], as opposed to smaller scale tests usually performed in controlled environments, such as laboratories. In AMT and similar purpose platforms, a dedicated and large workforce is allocated to participate in labeling tasks, in exchange for financial compensation. Crowdsourcing labeling has been used in the literature for quality of experience (QoE) evaluation and IQA-related tasks [75], including tone mapping [50].

In this work, samples are labeled by two different groups independently. In the first one, subjects are known by the authors, and are able to contact the authors about the tests. We denote this group as “private” group. The second group comprises subjects that are registered in the AMT platform, also called “AMT workers”. No interaction exists between AMT workers and the authors. We denote this group as “crowd” group. To reduce test duration and mitigate subject fatigue effects, we divide the 3009 samples into smaller sets [76]. Each set is called a test session. Samples are randomly assigned to each test session, and we ensure that scene content variety (defined in terms of scene type proportions and TMIQA score distributions) within each session is roughly the same observed when considering all 3009 samples

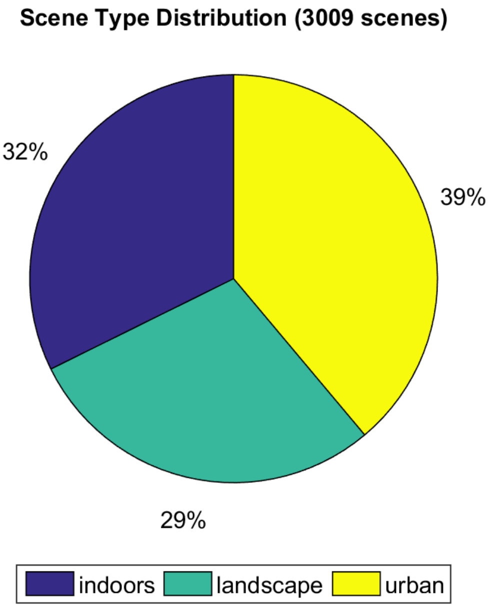


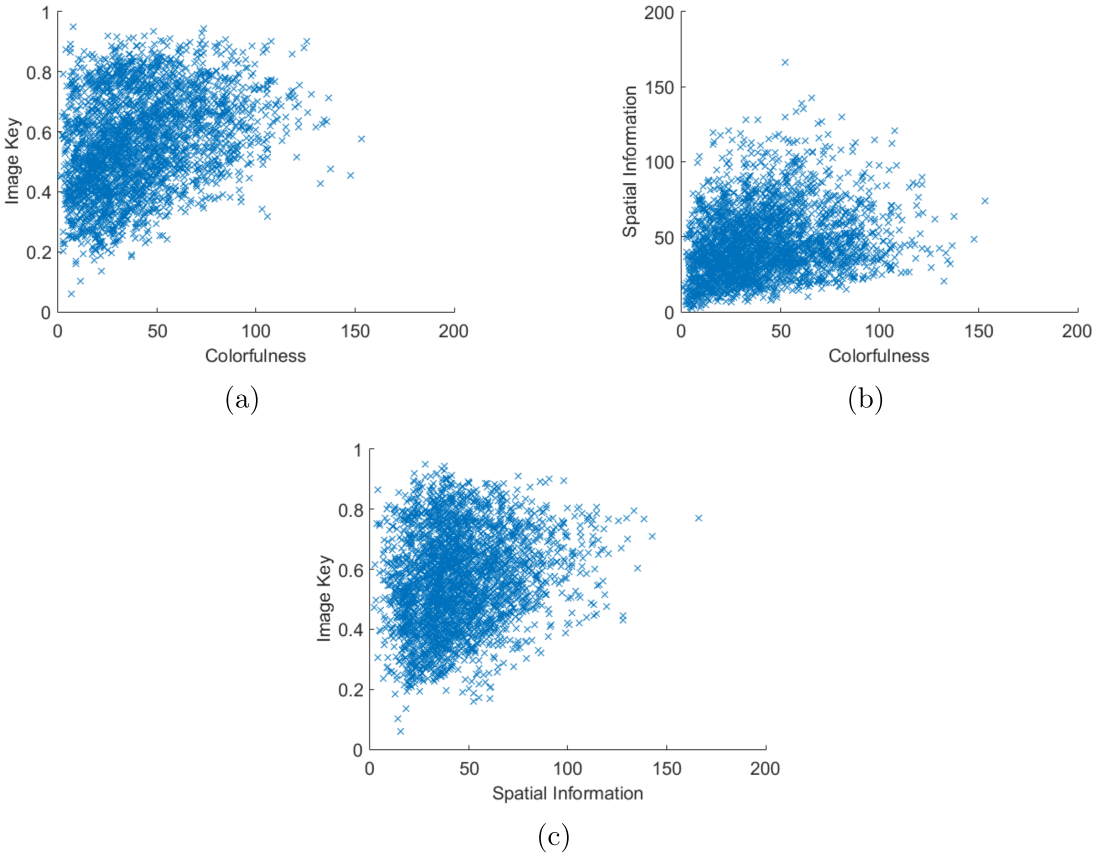
Figure 4.8: Pie chart showing each scene type proportion inside the 3009 sample subset. Overall balance between the three scene types is achieved.

(Figures 4.7 and 4.8).

We follow the Absolute Category Rating (ACR) method [77] to obtain the quality labels. The test interface contains a single sample that is displayed in its original resolution (256 × 256) in the center of the screen. Below the sample, a rating scale containing five adjectives is shown: “Excellent”, “Good”, “Fair”, “Poor”, and “Bad”. Subjects must choose the one adjective that better describes their sample quality impression. The sample remains on screen until an option is selected and submitted, so subjects may take as much time as they need to perform their assessment. Once the “Submit” button is clicked, scores for such sample can not be changed. After submission, a new sample is loaded on screen, and the test is repeated until all samples have been evaluated. Sample presentation order is randomized for each test session and subject.

Subjects can access test instructions at any time during the tests. We instruct subjects to perform their assessment based on image naturalness (*i.e.* if any degradation or artificial element is observed in the scene) and detail visibility, rather than scene content itself. We ask subjects not to zoom in or zoom out, as it can affect sample quality impression. Subjects may use different display devices for performing the test sessions, but we suggest using display devices on which test images can be evaluated in their native resolution (256 × 256) without zoom tools[3].

[3] How much screen area is occupied by the test image gives an idea whether the image can be properly evaluated in its native resolution. For instance: in full HD display devices (1920 x 1080), the test image occupies about 3% of total screen area. In comparison, the same test image appears



(a)

(b)

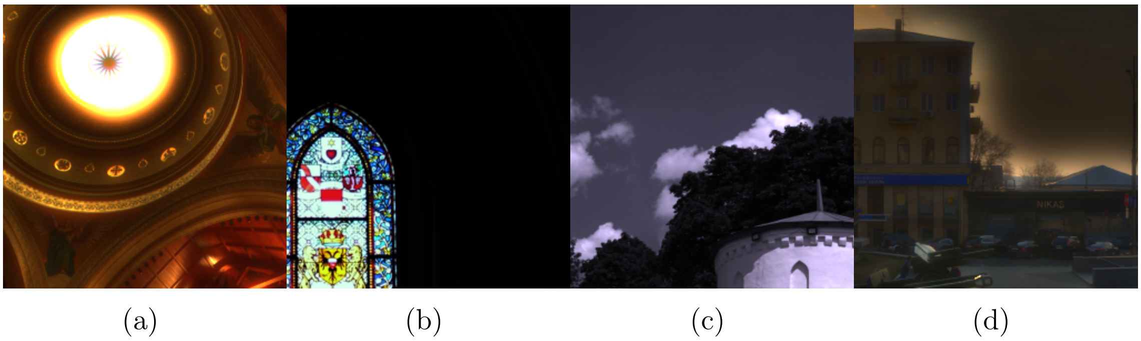
(c)

Figure 4.9: Scatter plots showing the colorfulness, spatial information and image key metric values for each sample in the 3009 PBTDB subset: (a) Colorfulness vs. Image Key; (b) Colorfulness vs. Spatial Information; and (c) Spatial Information vs. Image Key.

Because of AWS Ground Truth limitations, we do not create a separate session for training the subjects before the actual assessment task. Instead, subject training is performed at the beginning of the test session, and quality labels from “training” samples are discarded. Such training aims at familiarizing subjects with the test interface and with sample qualities they may find during the test. Each launched test session incurs in financial costs that are proportional to the number of evaluated samples. We want to use as few training samples as possible, in order to make the best use of financial resources. We consider that five samples are enough for training the subjects to reliably evaluate the quality of remaining test samples. In the instruction screen, we inform subjects that their first five sample quality assessments are not taken into account. We also present, in the instruction screen, some exemplary samples containing distortions that should be considered when evaluating quality, and distortion-free samples. Distortions include over-exposure, under-exposure, color degradation, and halo artifacts (Figure 4.10).

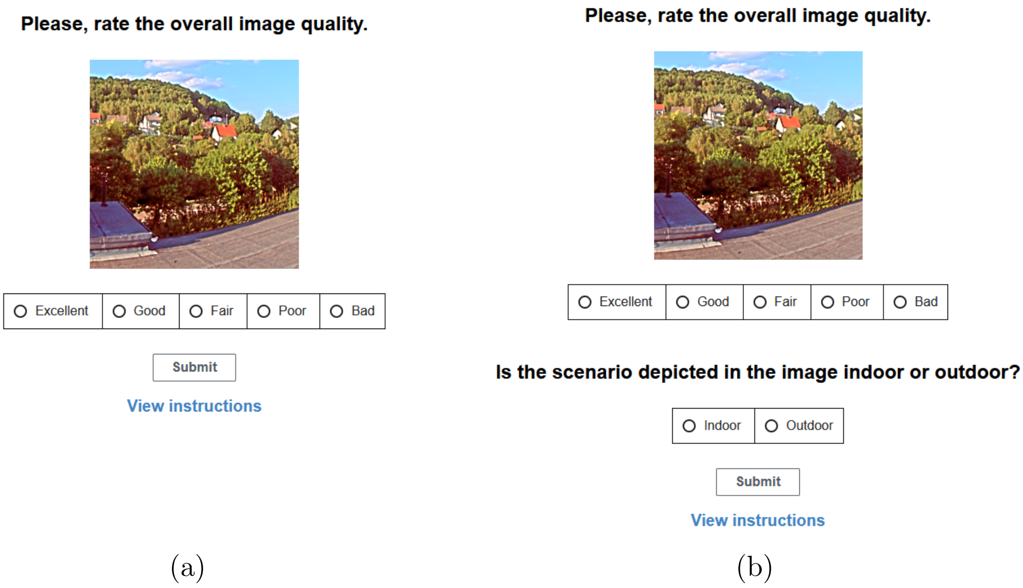
Subjects are allowed to take breaks during test sessions, as recommended in [77]

smaller on 4k display devices (3840 x 2160), as it occupies about 0.7% of total screen area.



(a) (b) (c) (d)

Figure 4.10: Exemplary samples shown in the test instruction screen of each distortion type: (a) Over-exposure; (b) Under-exposure; (c) Color degradation; and (d) Halo artifacts.



Please, rate the overall image quality.

Please, rate the overall image quality.

Is the scenario depicted in the image indoor or outdoor?

(a) (b)

Figure 4.11: Test interfaces used for the (a) private group, and (b) crowd group.

for large experiments. They can interrupt a test session at any time, saving their progress, and resume it later. We recommend subjects that break periods should not last more than 20 minutes. Examples of test interfaces used for the private and crowd groups are shown in Figure 4.11. Next, we detail some differences in test design for private and crowd groups.

Private Group Test Design

Subjects were invited to participate in up to three test sessions, which must be performed on different days. Each subject received written instructions on how to access the test platform and on how the test interface worked, prior to their first session. Subjects could access these instructions at any time, for subsequent sessions, if desired. We created 12 test sessions, each one containing 250 or 251 samples, and allocated subjects in order to roughly obtain the same number of labels per samples

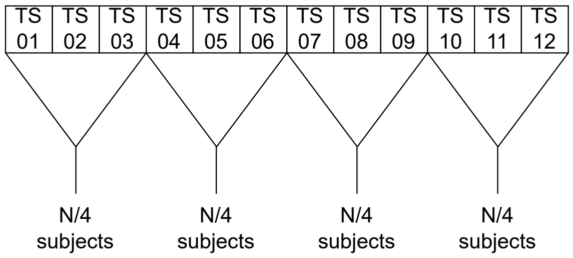
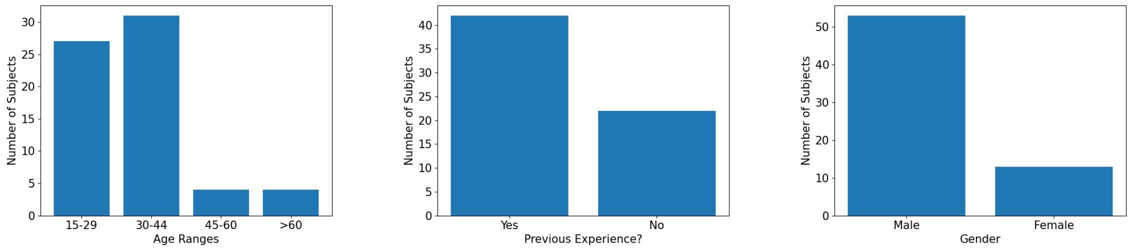


Figure 4.12: Private group subject allocation for test sessions, assuming all subjects perform the three sessions. “TS” stands for “Test Session”, and “N” is the total number of subjects in the private group (66). “Extra” subjects (*i.e.* if the remainder of N/4 is not zero) are assigned to different private groups. On average, 16 subjects participated per test session.



(a) (b) (c)

Figure 4.13: Private group subject demographics: (a) Age; (b) Previous experience with image processing tasks; and (c) Gender

in each test session, as depicted in Figure 4.12. Overall, 66 subjects participated in at least one test session (62 in three sessions, one in two sessions, and three in one session). On average, samples in each test session were labeled by 16 subjects. Figure 4.13 summarizes demographic data regarding the private group subjects. From the 66 subjects, 53 are men and 13 are women. Subject ages range from 19 to 72 years, with the majority (58) under 44 years old. Over half of the subjects (42) are familiar with image processing tasks and concepts, such as color saturation, brightness, among others. All subjects have normal or corrected-to-normal visual acuity. Two subjects have colorblindness, and the remaining subjects have normal color perception. We do not remove colorblind subjects from the experiments. We argue that, in our work, color distortions introduced by tone mapping are related to changes in color saturation rather than in hue. Then, colorblindness should not have a relevant impact on quality assessment of samples that may contain such color distortions.

Crowd Group Test Design

We created 10 test sessions, denoted as HITs (“Human-Intelligence Tasks”) in AWS Ground Truth, each containing 300 or 301 samples. We requested 75 labels per sample. Once an HIT is accepted, AMT workers have the option to abandon it at any time, even without completing it (that is, labeling all the samples in the test session). In this case, the worker releases the unfinished task in AMT platform, so that other workers can accept it and assess the remaining unlabeled samples. The HIT remains available for AMT workers until the desired number of labels per sample is achieved. On average, 266 AMT workers participated per HIT. Each worker is paid $0.012 per sample that they label. AWS Ground Truth does not provide demographic information regarding test session participants.

Because of the nature of crowdsourcing tasks (uncontrolled environments, subjects unknown to the authors, among others), additional steps must be taken when designing the tests for the "crowd" group, in order to ensure that subject labels are trustworthy [78]. These steps often include strategies that seek to increase subject attention to the task at hand, such as giving extra financial compensation if some subject performance value is achieved [78], or adding some entertainment value to the task [79]. Some strategies also aim at discouraging "cheating" behaviors from subjects, like monitoring subject activities during tests and making them aware of it [78, 80]. A "cheating" behavior usually occurs when subjects try to maximize financial gains by examining large numbers of samples in shorter times. In such cases, subjects give any label to the sample, without actually assessing its quality. In [81], authors categorize different "cheating" behavior patterns, and propose strategies to deal with them.

A simple strategy to rule out workers that carelessly assess sample quality consists in adding to the test questions whose answers are known to test designers and are also easy to determine by subjects that are truly engaged in the task. These questions are called Gold Standard Questions (GSQ), and are recommended in [76, 81]. We follow this strategy when designing the test interface for the crowd group, as shown in Figure 4.11(b). Besides image quality, we ask subjects to inform whether the current sample corresponds to an outdoor or an indoor scene. Most samples in the test clearly contain elements, such as open skies or walls and ceilings, that make subjects easily infer if the scene depicts an open-air (outdoor) or closed (indoor) environment.

4.2.2 Subjective Tests Consistency Analysis

After collecting the assessment labels, we discard the first five labels from each subject, as we treat these as "training" samples for the subjects. For the "crowd" group, two extra actions are taken before converting sample labels to MOS values. First, we remove all labels from workers that evaluated less than ten samples in total (after discarding their initial five samples). We argue that, in such cases, subjects may not have been fully familiarized yet with the task and the sample quality range to be expected, thus providing less reliable assessments. Also, a minimum number of evaluated samples is required so that the statistics representing worker GSQ performance are meaningful.

Next, for each worker, we count in how many samples the corresponding worker GSQ answer matches the expected GSQ answer. We calculate the percentage of correct GSQ answers, and denote it as GSQ correctness rate. Workers with GSQ correctness rates below a minimum threshold value are removed from the tests. We empirically set this threshold value to be 72%. To obtain this threshold value, we analyzed the outdoor sample percentage in each one of the ten test sessions for the crowd group experiment. We choose outdoor sample percentage, as there are more outdoor samples than indoor samples. The maximum outdoor sample percentage value, considering the ten sessions, is 69.4%. The chosen threshold value of 72% is slightly above this maximum value because it filters out workers that marked the same GSQ answer for all samples, or randomly marked any GSQ answer, without actually examining the samples. Such workers most likely did not perform the test carefully but wanted to optimize financial rewards by assessing more samples in shorter times [81]. Figure 4.14 illustrates the GSQ correctness rate of workers from one test session. After removing workers by the aforementioned actions, each sample is labeled by 51 subjects from the crowd group, on average.

Many unknown external factors may influence subject assessments, especially when tests are performed in uncontrolled environments. We follow the methodology proposed in [82] to check whether the private and crowd group subjective tests are consistent, and their corresponding labels are reliable. The idea is briefly explained next. The consistency analysis involves categorizing the label distribution from each sample as typical or atypical. Different patterns of label distributions are commonly observed in subjective tests [82]. Such distributions comprise the "typical" category. A statistical model that discriminates between both categories is required (that is, a model that fits typical distributions well, and fits any other distribution poorly). The statistical distribution used as model for this purpose is the Generalized Score Distribution (GSD) [83]. This model is described by two parameters: ψ, which denotes the sample "true"[4] label, and θ, which denotes the answer spread for the current sample.

The consistency analysis procedure is summarized in Figure 4.15. For each sample, we fit a GSD model to the observed distribution (*i.e.* estimate the ψ and θ parameters that better approximate the GSD model to the observed distribution). Next, we verify how well the fit GSD model actually represents the observed distribution. This is performed using a goodness-of-fit (GoF) test. The chosen GoF

[4] "True" as in the expected label according to the fit distribution.

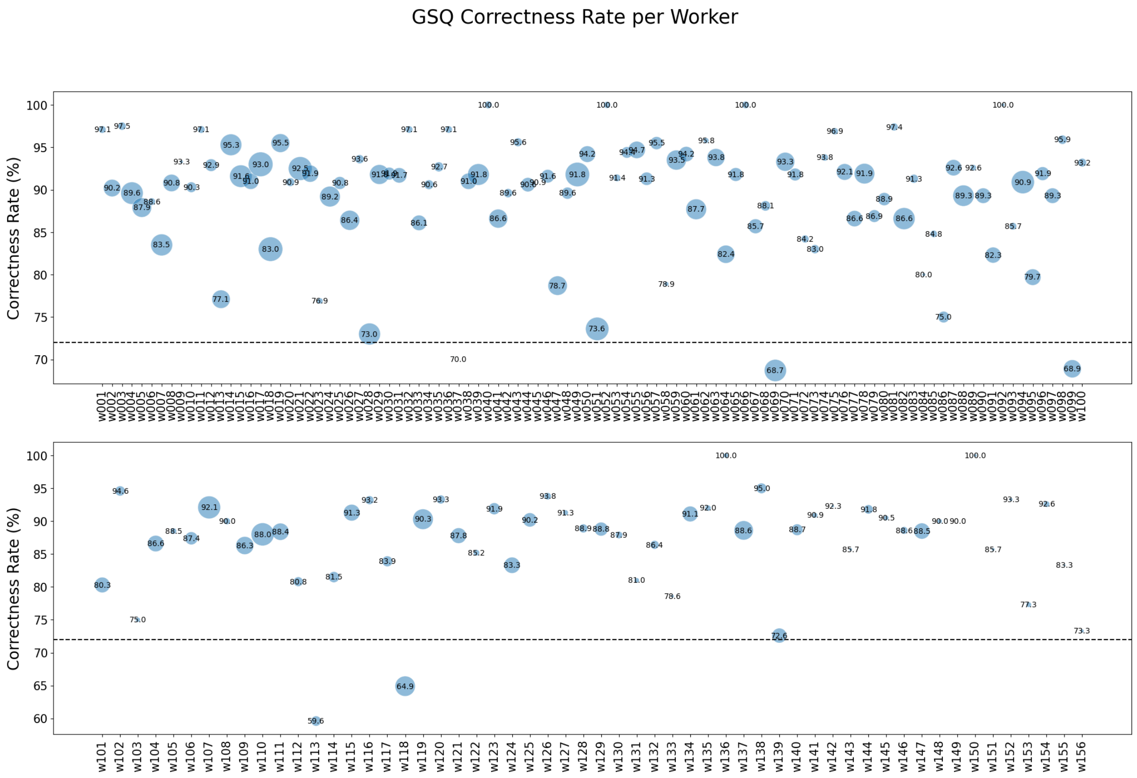


Figure 4.14: Bubble plot showing GSQ correctness rate for workers that participated in test session #5. In total, 156 participated in this session. Top plot shows performance from the first 100 workers, and bottom plot shows performance from 101th to 156th worker. Larger bubbles indicate that the corresponding worker evaluated more samples. Numbers inside the bubbles show the GSQ correctness rate from each worker. The dashed black line marks the minimum threshold of 72%.

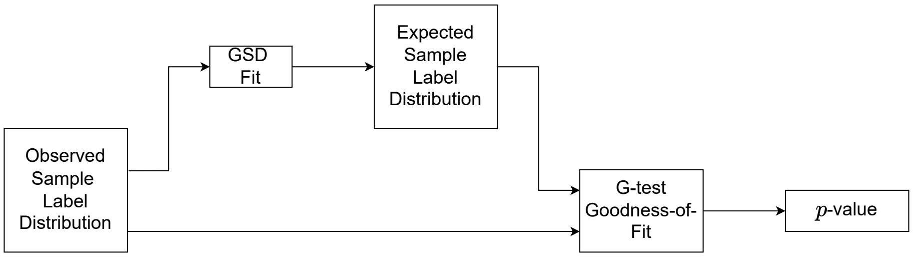

Observed Sample Label Distribution
GSD Fit
Expected Sample Label Distribution
G-test Goodness-of-Fit
p-value

Figure 4.15: Consistency test methodology proposed in [81]. Empirical label distributions are fit using a GSD model, thereby creating expected label distributions. Then, the G-test Goodness-of-Fit is performed and yields a p-value that indicates how well the GSD model fits the empirical label distribution. p-values vary from 0 to 1. Low p-values indicate poor fits.

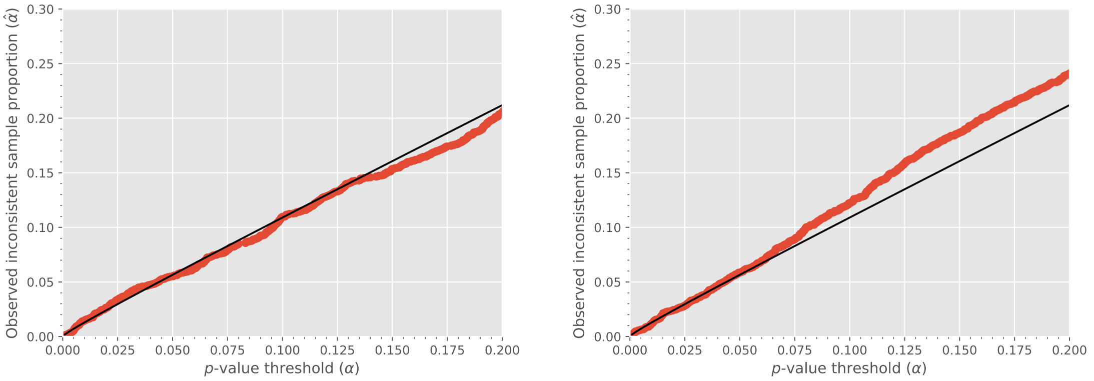
test is called G-test [84]. The test yields a number, called p-value, which ranges between 0 and 1. Higher p-values indicate that the model provides a good fit for the subjective data.

A single sample is treated as consistent if its label distribution can be well represented by the GSD model. Conclusions regarding consistency can also be extended to the subjective experiment as a whole by taking into account all the samples from the labeled dataset in this analysis. More specifically, the subjective test is considered consistent if the observed proportion of inconsistent samples, considering all samples in the dataset, is below a theoretical threshold. This threshold defines the maximum proportion of inconsistent samples that is expected for a given p-value used as reference. We denote the current p-value used as reference as α, and the inconsistent sample proportion, for the given α, as $\hat{\alpha}$. If we consider all possible p-values and calculate the corresponding theoretical thresholds, we obtain a threshold curve. Interested reader can find more details on how the theoretical threshold curve is determined in [82].

Subjective experiment consistency information can be graphically summarized by a plot called P-P plot [82]. In this plot, we show points whose coordinates are defined by α (*i.e.* the p-value being considered as reference) and the corresponding $\hat{\alpha}$ associated with it. Along with these plots is the theoretical threshold curve. Points that are above the threshold curve represent inconsistencies in the experiment. These cases correspond to an observed inconsistent sample proportion that exceeds the maximum expected inconsistent proportion for the current α. The difference between the observed and maximum expected inconsistent proportions (assuming the observed is higher than expected one) indicates how critical the inconsistency is. In order to draw conclusions regarding experiment consistency, we need to check how often inconsistencies occur, and how strong they are.

Figure 4.16 shows the P-P plots for the private and crowd subjective tests. The theoretical threshold curve is represented by the solid black line. Although α ranges between 0 and 1, the maximum considered α value is limited to 0.2 when analyzing the P-P plot, as suggested in [82]. This range comprises samples whose label distributions are poorly fit by the GSD model. Because such distributions are classified as atypical, they correspond to experiment inconsistencies that need to be carefully addressed and treated. Samples with greater p-values do not critically impact on experiment consistency. For these samples, the GSD model provides a good fit for their empirical label distributions, and the distributions are classified as typical. If the observed inconsistent proportion exceeds the maximum expected one, for a given range that includes greater p-values, then it is mainly caused by the low p-value samples contained within this range, rather than the higher p-value ones. Therefore, the α value range from 0 to 0.2 is more critical to consider than the entire range from 0 to 1 in order to draw conclusions regarding experiment consistency.

From Figure 4.16, we observe that the crowd group experiment is more inconsistent than the private group one, since more points fall above the black line. However, because $\hat{\alpha}$ values also oscillate around the theoretical thresholds, neither experiment can be considered entirely consistent or inconsistent. In such cases, samples with low p-values must be analyzed individually. This is to gain insight into the causes



(a)

(b)

Figure 4.16: p-value P-P plots of the subjective tests, before taking any actions: (a) private group; and (b) crowd group. Black line corresponds to the theoretical threshold. Points above the black line indicate inconsistencies. Neither experiment is entirely consistent or inconsistent, as points oscillate around the black line.

of inconsistency, and then potentially take actions that may resolve them. For each experiment, we examine all samples whose p-values are lesser than or equal to the α value of the first right-most point above the black line in the corresponding P-P plot. In total, 438 and 724 samples are examined for the private and crowd group experiments, respectively.

We define three possible actions to be taken for each examined sample: (i) remove the sample from the experiment; (ii) remove some labels from the sample distribution; and (iii) do nothing. First action is chosen when subject assessments are highly contradicting (for instance, a similar number of "Bad" and "Excellent" assessments for the current sample). Possible explanations for this situation include personal preference for some scene contents, test instruction misunderstanding, or even "difficult" scenes to evaluate (that is, scenes that do not present much content to assess). Figure 4.17 illustrates some removed samples, along with their label distributions. Second action is normally taken when samples present few labels that deviate from the most voted groups. Usually such labels are unacceptable, as they either ignore the strong presence of distortions that should be considered in the assessment or negatively qualifies samples that have no visible distortions. We hypothesize that such cases may also happen because subjects use different display devices and configurations, which affect how the sample is displayed on screen. Also, viewing conditions (*e.g.* distance from screen, room illumination conditions) may vary across subjects, which impact on quality evaluation. We do not specify recommended test environment setups because tests are performed remotely and, thus, we are unable to ensure that such recommendations are met by subjects. These labels are considered outliers. Examples of samples that have at least a score

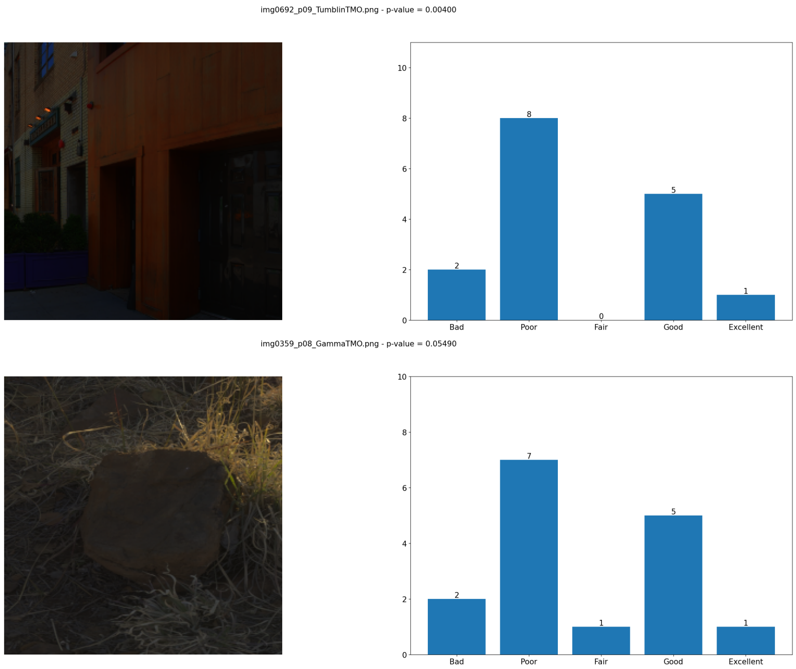


Figure 4.17: Examples of samples removed during private group experiment consistency analysis. Both samples present two groups of subjects that vote for opposing quality labels.

removed are shown in Figure 4.18. Finally, in many cases, we consider all opinions to be valid for the current sample, and reasons for the inconsistency are unclear. Then, we decide for the third action, which is to keep the sample and preserve its subjective label distribution. Figure 4.19 shows some examples of unchanged samples.

Table 4.3 summarizes the actions that are adopted to treat the examined low p-value samples, and how many samples are affected by them in each subjective experiment. This table also shows the final number of samples obtained after subtracting the number of inconsistent samples presented in "*Removed Entirely*" column from the original 3009 sample subset in each experiment. Although the original 3009 sample subset is the same for both groups, inconsistent samples are not necessarily the same for each group. Even for samples that are inconsistent in both subjective experiments, we point out that not necessarily the same action is taken for these samples in each experiment. The action depends on the sample label distribution, which may differ between each group. We note that proportionally more samples are discarded when considering the crowd group experiment. This is expected, as this experiment is more inconsistent than the private group one.

After taking such actions, we repeat the aforementioned consistency analysis, and observe the corresponding P-P plots from each subjective experiment. These plots are shown in Figure 4.20. In the private group experiment, all points now fall below the theoretical threshold line. Thus, this experiment is now regarded as consistent. In the crowd group experiment, the points slightly exceed the theoretical

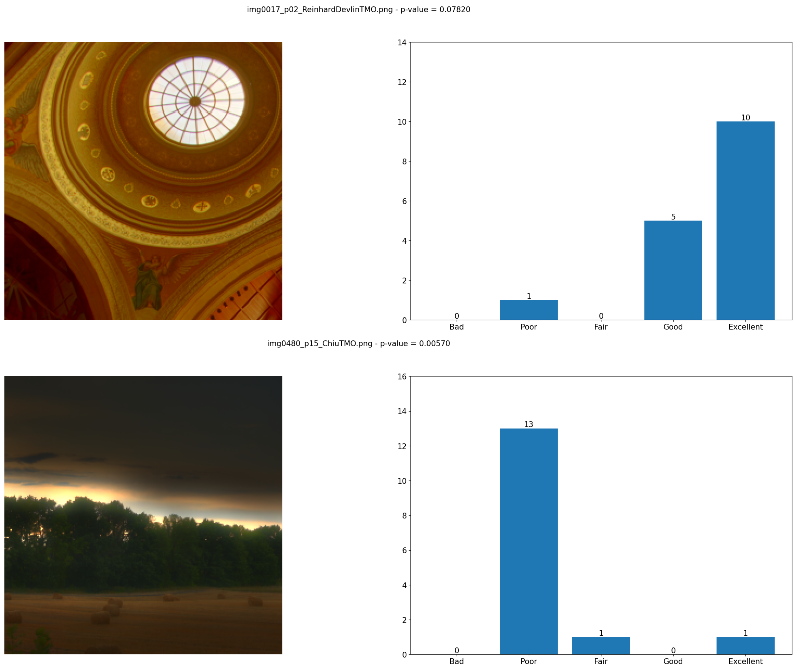


Figure 4.18: Examples of samples that had at least one label removed from the original distribution during consistency analysis, as they are considered outliers. In top row, the single “Poor” label is removed, because no distortions are visible in the image. In the bottom row, the single “Excellent” label is removed, because a strong distortion (halo artifact) is perceived in the image.

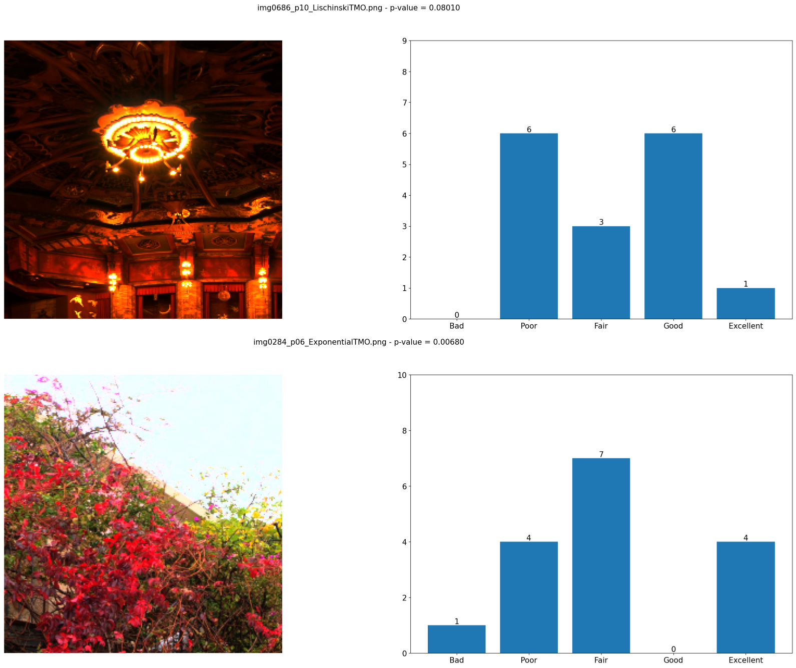
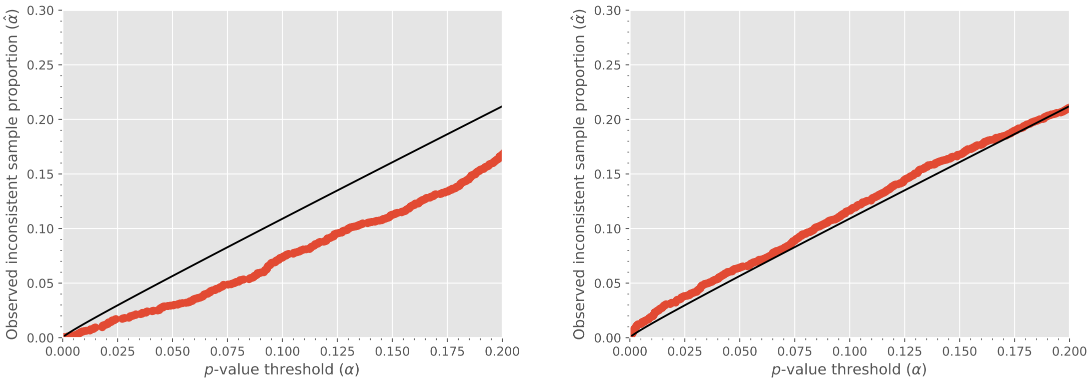


Figure 4.19: Examples of samples that were preserved during consistency analysis. There are no clear indications of what caused the inconsistency. All labels are considered valid opinions for these samples.

threshold line, but are generally closer to this line than they were before taking actions (Figure 4.16(b)). Although some inconsistencies are still present in the

Table 4.3: Table summarizing the actions taken to treat samples with low p-values and the number of samples affected by them in each subjective experiment. Percentages refer to the sample proportion relative to the total number of inconsistent samples in each case (shown in "Total Examined Samples" column). Last column shows the final number of samples after subtracting the respective numbers defined in the "*Removed Entirely*" column from the original 3009 sample subset.

Experiment	**Action**			**Total Examined Samples**	**Final Subset Size**
	Remove Entirely	*Remove At Least One Label*	*Leave Unchanged*		
Private Group	9 (2.1%)	124 (28.3%)	305 (69.3%)	438	3000 samples
Crowd Group	65 (9%)	126 (17.4%)	533 (73.6%)	724	2944 samples

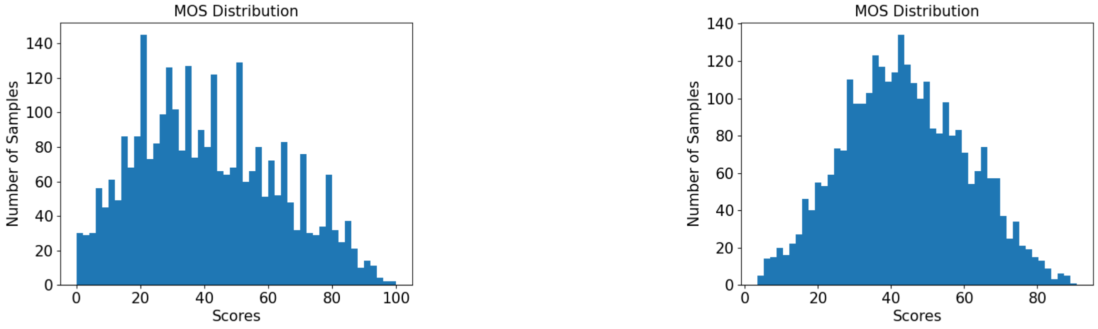


(a) (b)

Figure 4.20: p-value P-P plots of the subjective tests, after taking the specified actions: (a) private group; and (b) crowd group. Black line corresponds to the theoretical threshold. Now, the private group experiment is consistent, as all points fall below the black line. The crowd group experiment is still not entirely consistent, but inconsistencies are now less critical, as points are closer to the black line.

crowd group experiment, they are less critical than previously. We also point out that the GSQ approach alone may be effective when dealing with subjects that quickly and randomly mark any answer and quality label for samples. However, this is not the case when dealing with other types of "cheating" behaviors [81]. As such, other unreliable workers may not have been removed, and their labels contribute to experiment inconsistency.

To summarize, we have almost the same PBTDB sample subset (apart from some removed samples in each experiment) subjectively labeled by two groups: the private group, with 3000 samples, and the crowd group, with 2944 samples. Finally, for each sample, we use its label distribution to obtain the corresponding MOS value, which is a single numeric value that represents the overall sample quality score. We assign the following scores to each label: 100 to "Excellent", 75 to "Good", 50 to



(a) (b)

Figure 4.21: MOS distributions considering the scores obtained from different subjective labeling: (a) private group labeling; and (b) crowd group labeling.

"Fair", 25 to "Poor", and 0 to "Bad". Then, the sample MOS value is calculated by summing all its scores and dividing the result by the number of labels for that sample. Such score mapping rule ensures that MOS values are within the [0, 100] range, and higher scores correspond to better sample qualities, which is the same convention used in [50].

Figure 4.21 shows the MOS distributions obtained from the private and crowd group scores. In this work, we denote the sample subset that uses private group MOS values as "private-PBTDB", and the sample subset that uses crowd group MOS values as "crowd-PBTDB". We note that the private-PBTDB MOS distribution is slightly skewed towards the lower score range (and, hence, lower quality samples). Comparing with Figure 4.7, we observe that this pattern is similar to the BTMQI score distribution, in which higher scores correspond to lower quality samples. The crowd-PBTDB MOS distribution is almost centered in the middle of the score range. Such pattern is also similar to objective TMIQA score distributions, namely TMQI and HIGRADE (Figure 4.7). This suggests that the MOS values obtained from each subjective experiment that we performed are somewhat related to objective TMIQA metrics from the literature.

4.3 Chapter Summary

In this chapter, we presented the new tone-mapped image database for TMIQA, which is called PBTDB. It contains about 175000 samples, whose quality is labeled by TMQI, BTMQI, HIGRADE-1 and HIGRADE-2 metrics. We also performed two remote subjective tests on a subset of 3009 representative samples from PBTDB. These experiments are denoted as private group (small-scale experiment) and crowd group (large-scale experiment) tests. Subjects in private group are known to the authors and were personally invited to the tests, whereas subjects in crowd group are unknown to the authors and hired from AWS Ground Truth online service.

We detailed how the subjective tests were performed. For instance, how the test interface in each experiment looks like, what aspects should be considered in evaluating sample quality, how many test sessions were performed, among others. We removed unreliable subjects based on how many samples they evaluated, and in how many samples they answered the GSQ correctly. We used the methodology proposed in [82] to identify and treat samples with atypical label distributions (which are also referred to as inconsistent samples) in each experiment.

After treating inconsistent samples in each experiment, we obtained the final subjectively labeled subsets: the private group subset (private-PBTDB), containing 3000 samples, and the crowd group subset (crowd-PBTDB), containing 2944 samples. Consistency analysis on the final subsets showed that all samples in private-PBTDB are consistent, whereas crowd-PBTDB still presented some inconsistent samples. We conjecture that inconsistent samples from crowd-PBTDB occurred because subjects did not really assess sample quality as we requested. This suggests that additional mechanisms that capture attention of the subjects, and engage them in the task they are performing, need to be considered. The only such mechanism adopted in this work is the GSQ approach. Such approach alone is not sufficient to ensure that subjects from the crowd group are performing the test carefully and are following test instructions correctly.

In the next chapter, we present the experiments in which we train deep learning models using PBTDB, and test them in different datasets, namely ESPL-LIVE and TMID. The idea of the deep learning experiments is to verify how generic are the features learned from PBTDB samples for the TMIQA task. We consider different objective quality scores (from entire PBTDB), as well as subjective quality scores (from private-PBTDB and crowd-PBTDB), for training the deep learning models and compare their performance values.

Chapter 5

TMIQA Experiments using Regression Models Based on Deep Learning

The present chapter describes three experiments. In the first experiment, we fine-tune the RankIQA model for the TMIQA task. In the second experiment, we train deep learning models that are based on three well-known architectures, namely VGG-16, ResNet50 [22], and Inception [85]. Such architectures were trained originally for the image classification task in the ImageNet dataset [86], and are now used for a variety of image processing tasks. In both experiments, deep learning models are fine-tuned/trained using the PBTDB dataset, and tested in the TMID and ESPL-LIVE datasets. In the third experiment, we use the PBTDB sample subset described in Chapter 4 to train a deep learning model based on the VGG-16 architecture. Quality labels from this subset correspond to MOS values obtained from private group and crowd group subjective experiments. We compare how different subjective labeling affect model performance. These experiments are summarized in Table 5.1, and their results are presented in the Sections 5.1, 5.2 and 5.3.

Table 5.1: Summary of experiments presented in this chapter. In columns "Training Databases" and "Test Databases", numbers in parenthesis denote the sample quantity in the respective databases. We point out the main aspects of each experiments: the training and test databases for the deep learning models; the quality labels used for PBTDB samples; and the model architectures considered.

Experiment Identifier	**Experiment Part**	**Training Databases**	**Quality Labels**	**Test Databases**	**Model Architecture**	**Section**
1	-	PBTDB (175000)	Multiple TMIQA metric scores	ESPL-LIVE + TMID	RankIQA	5.1
2	-	PBTDB (175000)	TMQI scores	ESPL-LIVE + TMID	Base CNN feature extraction + one fully-connected layer	5.2
3	1	Private-PBTDB (2700)/ Crowd-PBTDB (2650)	Subjective MOS values	Private-PBTDB (300)/ Crowd-PBTDB (294)	Base CNN feature extraction + one fully-connected layer	5.3
	2	Private-PBTDB (3000)/ Crowd-PBTDB (2944)	Subjective MOS values + TMQI scores	ESPL-LIVE + TMID	Base CNN feature extraction + one fully-connected layer	5.3

5.1 Experiment 1: RankIQA Fine-Tuning

RankIQA model learns how to predict quality scores by first learning to rank image quality. In [17, 47], this model is trained in two stages, as summarized in Figure 5.1. In the first stage, a Siamese architecture [87], which consists of two identical networks that are jointly trained and share weights, is trained to correctly rank images from an input pair, based on their corresponding distortion levels. Both images from the pair are impaired by only one specific distortion, but in different levels for each image. Several distortion types are considered, such as blur, white noise, JPEG blocking artifacts and others. In the second stage, one network from the Siamese architecture is fine-tuned using datasets that contain images impaired by the same considered distortions, but each image is labeled with a subjective MOS. In the end, the network learns to predict a quality score for the input image based on the intensity of such distortions that may be present. The described architecture is the publicly available version of the RankIQA model, which has been used so far in this work. In this section, we refer to this version as the "base RankIQA model". Learning from ranking is a potentially useful idea for determining tone-mapped image quality. We hypothesize that ranking the image quality can be regarded as an indirect way of assessing how complex tone-mapping distortions, which are difficult to directly quantify, impact on the overall quality perception.

We fine-tune the base RankIQA model for the TMIQA task, using the PBTDB dataset. We use 90% of the samples for training (158327 samples), and 10% of the samples for validation (17592 samples). The following $\mathcal{L}_2$ loss function is used for training:

$$L = \frac{1}{B} \sum_{i=1}^{B} (y_i - \hat{y}_i)^2 , \tag{5.1}$$

where B denotes the mini-batch size (in this case, $B = 30$ images), y_i is the labeled quality score of the i-th image, and $\hat{y}_i$ is the quality score predicted by the model for the i-th image. We test two optimization algorithms for minimizing the loss function from Equation (5.1): the SGD (with momentum) algorithm [88], and the Adam algorithm [89]. For the SGD optimizer, the momentum parameter is set to 0.9, and the base learning rate is set to 10^{-6}. Weight decay regularization is also used [90], with decay factor equal to 5 $\times 10^{-4}$. For the Adam optimizer, a base learning rate is set to 10^{-6}, and no regularization is performed. For both optimization algorithms, learning rate is divided by half every 5000 iterations (the model weights and biases are updated at each iteration). Models trained with the Adam optimizer achieve better performance values, and these are ones reported in this section. For models trained with the Adam optimizer, we experiment with a step size of 2500 iterations

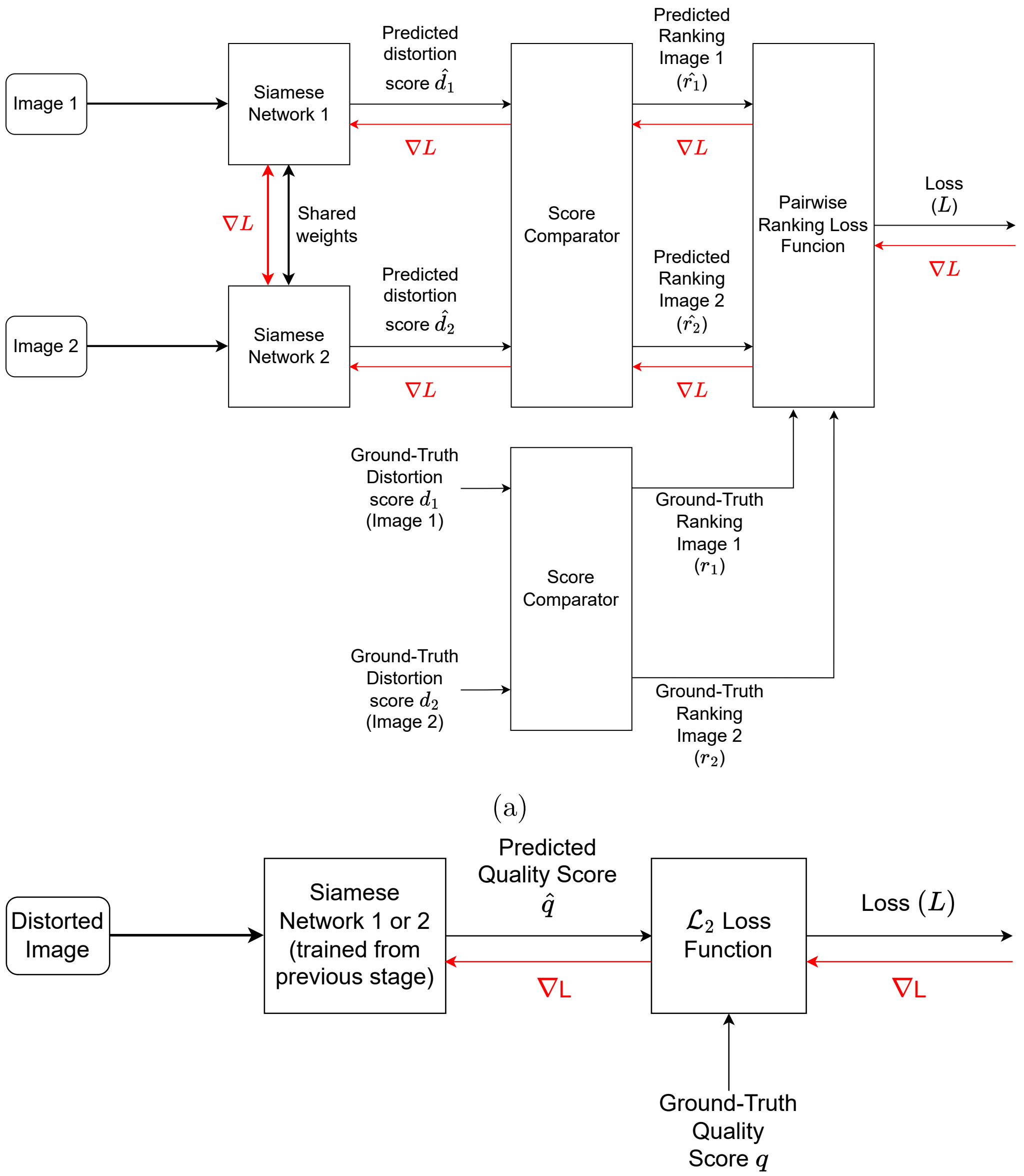


Image 1
Image 2
Siamese Network 1
Siamese Network 2
Shared weights
∇L
Predicted distortion score $\hat{d}_1$
Predicted distortion score $\hat{d}_2$
Score Comparator
Predicted Ranking Image 1 ($\hat{r_1}$)
Predicted Ranking Image 2 ($\hat{r_2}$)
Pairwise Ranking Loss Funcion
Loss (L)
Ground-Truth Distortion score d_1 (Image 1)
Ground-Truth Distortion score d_2 (Image 2)
Score Comparator
Ground-Truth Ranking Image 1 (r_1)
Ground-Truth Ranking Image 2 (r_2)
Distorted Image
Siamese Network 1 or 2 (trained from previous stage)
Predicted Quality Score $\hat{q}$
$\mathcal{L}_2$ Loss Function
Loss (L)
∇L
Ground-Truth Quality Score q

(a)

(b)

Figure 5.1: RankIQA training stages: (a) first stage; and (b) second stage. In the first stage, the model learns to rank images based on some distortion type. In the second stage, the model is fine-tuned to learn how to predict quality scores for the task at hand. Red arrows show the gradient loss value, with respect to network parameters, that is back-propagated to train the networks. Interested reader can find the pairwise ranking loss function definition in [17, 47]. The $\mathcal{L}_2$ loss function is defined in Equation (5.1).

for reducing the base learning rate in half, but better results are achieved with the 5000 iterations step size. We train the models for 30000, 40000 and 52780 iterations (which corresponds to, approximately, 5.5, 7.6 and 10 epochs, respectively). No performance improvement is observed after 10 epochs. Therefore, training lasts for 10 epochs, for all models.

Several RankIQA models are considered. Each model corresponds to fine-tuning

Table 5.2: RankIQA models trained with different quality scores as labels for the images in PBTDB.

Model	**PBTDB Sample Label**
RankIQA2	TMQI scores
RankIQA3	HIGRADE-2 scores
RankIQA4	(TMQI scores + HIGRADE-2 scores)/2
RankIQA5	Converted MOS from TMQI scores in TMID dataset
RankIQA6	Converted MOS from BTMQI scores in TMID dataset

the base RankIQA model using a different quality score as sample label $\hat{y}_i$ in Equation (5.1). We analyze how labels given by different TMIQA metrics influence the model performance in different test datasets. Table 5.2 lists the trained RankIQA models, along with the corresponding quality scores used as labels. The models are enumerated from "RankIQA2" to "RankIQA6". The first "RankIQA" corresponds to the base RankIQA model that is not fine-tuned.

We use the TMQI scores as the "official" labels for the PBTDB samples, and train RankIQA2. TMQI metric is commonly used in the literature both to objectively assess tone-mapped image quality [91, 92], and as baseline for comparison with other TMIQA metrics, [93, 94]. Also, we hypothesize that, since TMQI is an FR metric, its quality predictions are more reliable for "generic" tone-mapped images (*i.e.* tone-mapped images from any source) than quality predictions from HIGRADE and BTMQI metrics. In TMID dataset, TMQI scores reliably represent image quality, but it is unknown whether this is also the case in ESPL-LIVE dataset[1]. HIGRADE-2 scores represent well the image quality of samples in ESPL-LIVE dataset, and we train RankIQA3 using such scores as labels for PBTDB samples. For RankIQA4, we combine TMQI and HIGRADE-2 scores to create an average quality score, which serves as label. We map HIGRADE-2 scores into the same range as TMQI scores, before taking the average. For RankIQA5 and RankIQA6, labels correspond to MOS values converted from TMQI and BTMQI scores, respectively. These "converted MOS" values are obtained by applying functions that map TMQI and BTMQI scores into subjective MOS values in the TMID dataset. These functions are fit via non-linear regression. They are shown in Figure 5.2.

Training and validation loss curves for each model are shown in Figure 5.3. Table 5.3 presents the performance values of each trained model when applied in the ESPL-LIVE and TMID datasets, along with the base RankIQA model performance in both datasets. Scatter plots between subjective MOS and predicted MOS from each trained model, considering both datasets, can be found in Appendix A. In ESPL-LIVE dataset, all models perform poorly, whereas, in TMID dataset, per-

[1] TMQI performance can not be measured in ESPL-LIVE dataset because this dataset does not provide the original HDR images.

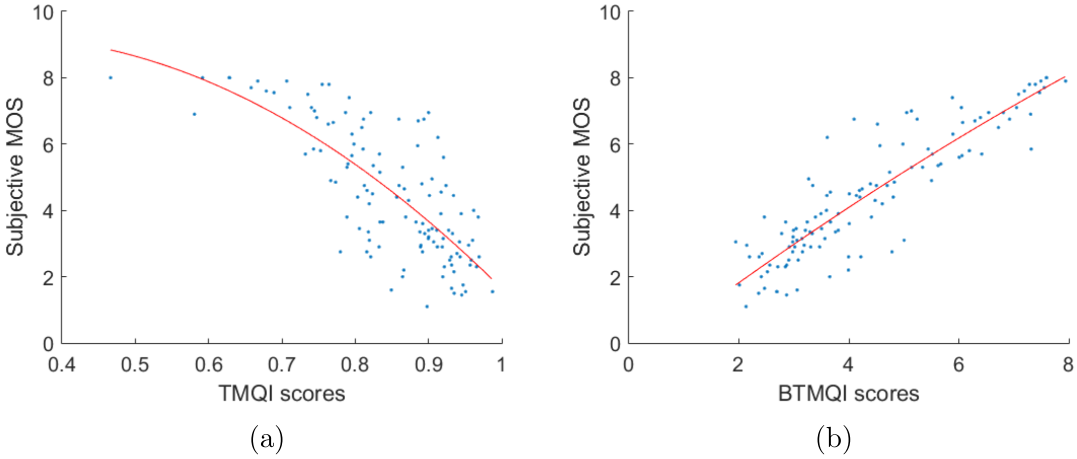


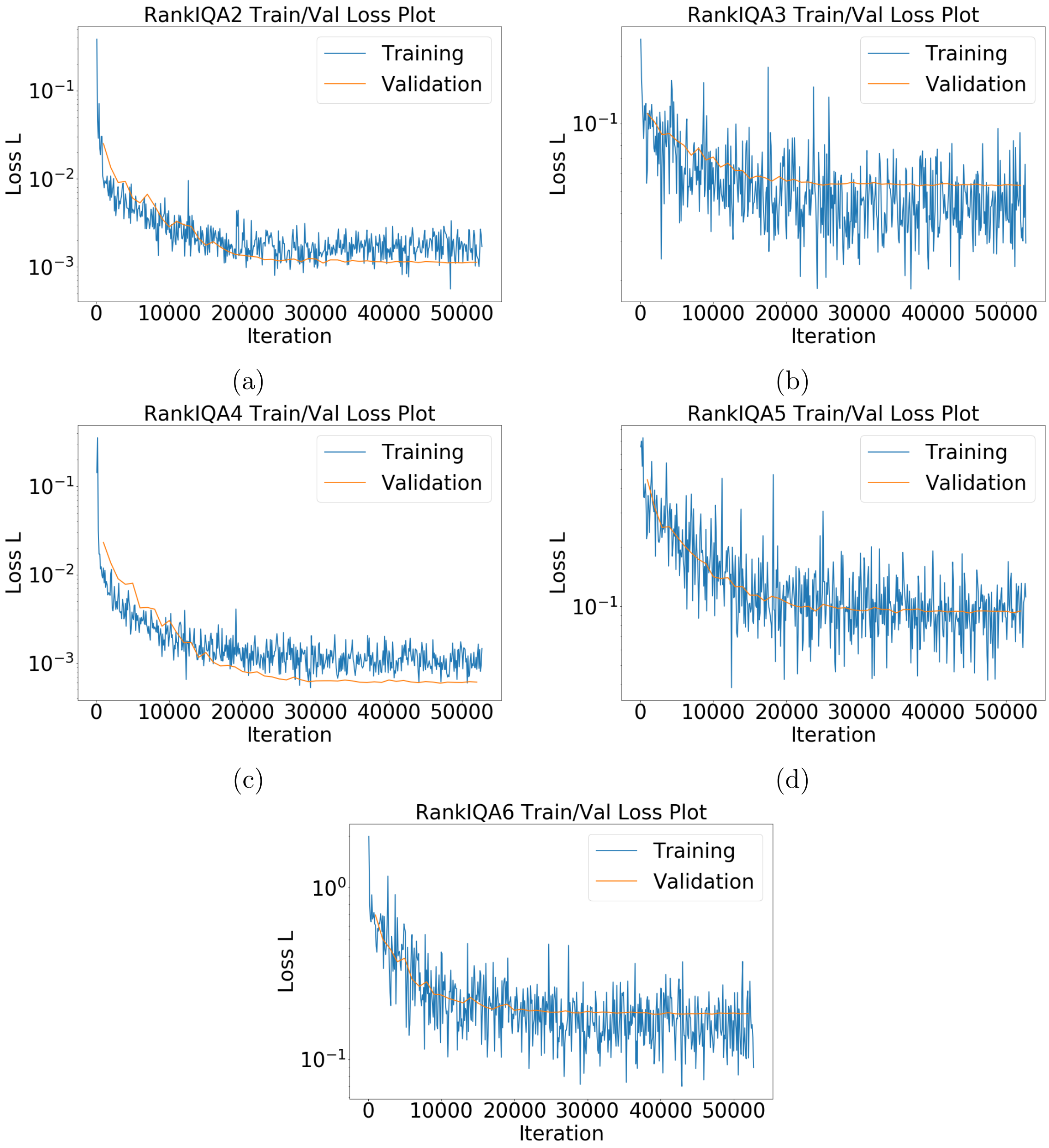
Figure 5.2: Scatter plots between subjective MOS values and (a) TMQI, and (b) BTMQI. The red line indicates the mapping function that is fit.

formance varies for each model. RankIQA2 achieves particularly low performance values in ESPL-LIVE dataset, which suggests that TMQI scores may not be adequate to represent overall quality of images from this dataset. In TMID dataset, where TMQI scores are well correlated with subjective MOS, RankIQA2 performs better. Considering all fine-tuned RankIQA models, RankIQA3, which is trained with HIGRADE-2 scores as labels, achieves the best performance in the ESPL-LIVE dataset, but the worst in TMID dataset. RankIQA4, whose training labels are the average of TMQI and HIGRADE-2 scores, performs a little better than RankIQA2 and RankIQA3 models in ESPL-LIVE and in TMID datasets, respectively. RankIQA5 and RankIQA6 present the best performance values in the TMID dataset, thus indicating that subjective quality impression is more closely related to converted MOS values than to TMQI (and, potentially, to BTMQI) scores. Such converted MOS values, however, are not good representations for quality of any tone-mapped image, as they do not seem appropriate, for instance, for the samples from ESPL-LIVE dataset (both models perform poorly in this dataset).

This experiment confirms that deep learning model performance highly depends on which TMIQA metric score is used as label to represent the image quality. It also reinforces the idea that state-of-the-art TMIQA metrics provide unreliable tone-mapping quality estimates for most tone-mapped images (that is, images that do not belong to any particular dataset).

5.2 Experiment 2: Architecture Exploration

From the previous experiment, it is unclear whether the specific RankIQA architecture is suited for the TMIQA task (and, hence, if the architecture has an important impact in the performance values observed). In this second experiment, we train



(a)

(b)

(c)

(d)

(e)

Figure 5.3: Training and validation loss curves of each RankIQA model in PBTDB: (a) RankIQA2, (b) RankIQA3, (c) RankIQA4, (d) RankIQA5, and (e) RankIQA6. For better visualization, log scale is used in y axis.

models with different architectures using the TMQI quality score as label for the PBTDB samples. The considered architectures are VGG-16, ResNet50, and Inception. These architectures are composed by two parts: the feature extraction part, composed by convolutional layers, and the classification part, composed by fully-connected layers. We maintain the feature extraction part (which we refer to as "Base CNN model"), and replace the original classification part of these models by one fully-connected (FC) layer, and one output layer, containing one neuron that yields the model prediction. Figure 5.4 depicts the overall model structure used in

Table 5.3: RankIQA model performance values in the ESPL-LIVE and TMID datasets. "RankIQA" denotes the base RankIQA model, that is not fine-tuned. Boldface text marks the best result for each performance metric in the corresponding dataset.

Model	ESPL-LIVE Dataset			TMID Dataset		
	PLCC	**SRCC**	**RMSE**	**PLCC**	**SRCC**	**RMSE**
RankIQA	0.252	0.235	9.712	0.108	0.137	1.922
RankIQA2	0.209	0.176	9.789	0.694	0.583	1.399
RankIQA3	**0.482**	**0.432**	**8.778**	0.430	0.370	1.751
RankIQA4	0.326	0.275	9.463	0.553	0.487	1.605
RankIQA5	0.260	0.245	9.668	0.786	**0.723**	1.196
RankIQA6	0.344	0.339	9.398	**0.816**	0.716	**1.136**

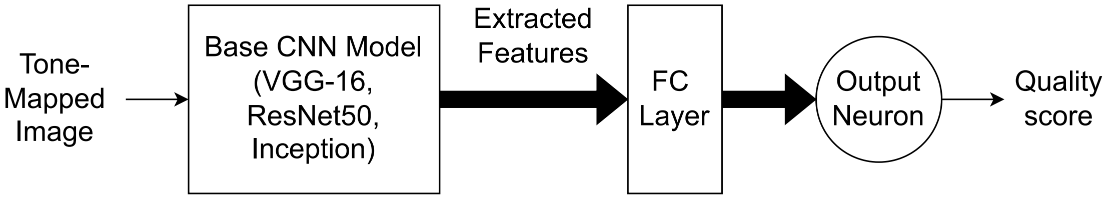


Figure 5.4: Overall model architecture trained to predict tone-mapped image quality. The models are trained using TMQI scores of PBTDB samples as target quality values.

this experiment. The VGG-16 architecture version that is used in the present work includes batch normalization layers [95] after every convolutional layer in the feature extraction part.

The training setup for each model is summarized in Table 5.4. We only train the newly introduced FC and output layers. The weights and biases of the feature extraction part are frozen. These frozen weights and biases are the ones learned from the ImageNet dataset for the image classification task. As in the previous experiment, 90%/10% of the PBTDB samples are used for training/validation. This corresponds to 158327 samples for training, and 17592 samples for validation. For each model, we vary the number of neurons in the FC layer along a sequence of powers of two, from 2 to 2^{14}, and choose the architecture that provides a good trade-off between number of neurons in this layer and validation loss. Activation functions for FC layer neurons are leaky rectified linear units (LeakyReLUs) [96], with negative slope set to 0.01. The $\mathcal{L}_2$ loss function from Equation (5.1) is used in this experiment to train the models. We choose Adam as the loss function optimizer, with learning rate lr initially set to 10^{-3}. This learning rate decays exponentially every epoch according to: $lr = lr(0.9)^n$, where n is the current epoch number, and 0.9 is the learning rate decay rate. For VGG-16 and ResNet50 models, input images (that have original resolution of 256×256 pixels) are resized to 224×224 pixels, whereas for Inception model, they are resized to 299×299 pixels. All resizing

Table 5.4: Training setup for deep learning models in this experiment.

Model	Input Image Size	Mini-Batch Size	Optimizer	Learning Rate Scheduler	FC Layer Neurons	Activation Units	Epochs
VGG-16	224 × 224	32	Adam ($lr = 10^{-3}$)	$lr = lr(0.9)^n$	64	LeakyReLU (negative slope = 0.01)	40
ResNet50	224 × 224	32	Adam ($lr = 10^{-3}$)	$lr = lr(0.9)^n$	4096	LeakyReLU (negative slope = 0.01)	40
Inception	299 × 299	32	Adam ($lr = 10^{-3}$)	$lr = lr(0.9)^n$	512	LeakyReLU (negative slope = 0.01)	40

Table 5.5: Performance values of each considered CNN architecture in ESPL-LIVE and TMID datasets. Boldface text marks the best performance values in each dataset. The "CNN-VGG16-FT" model denotes the fine-tuned version of the "CNN-VGG16" model.

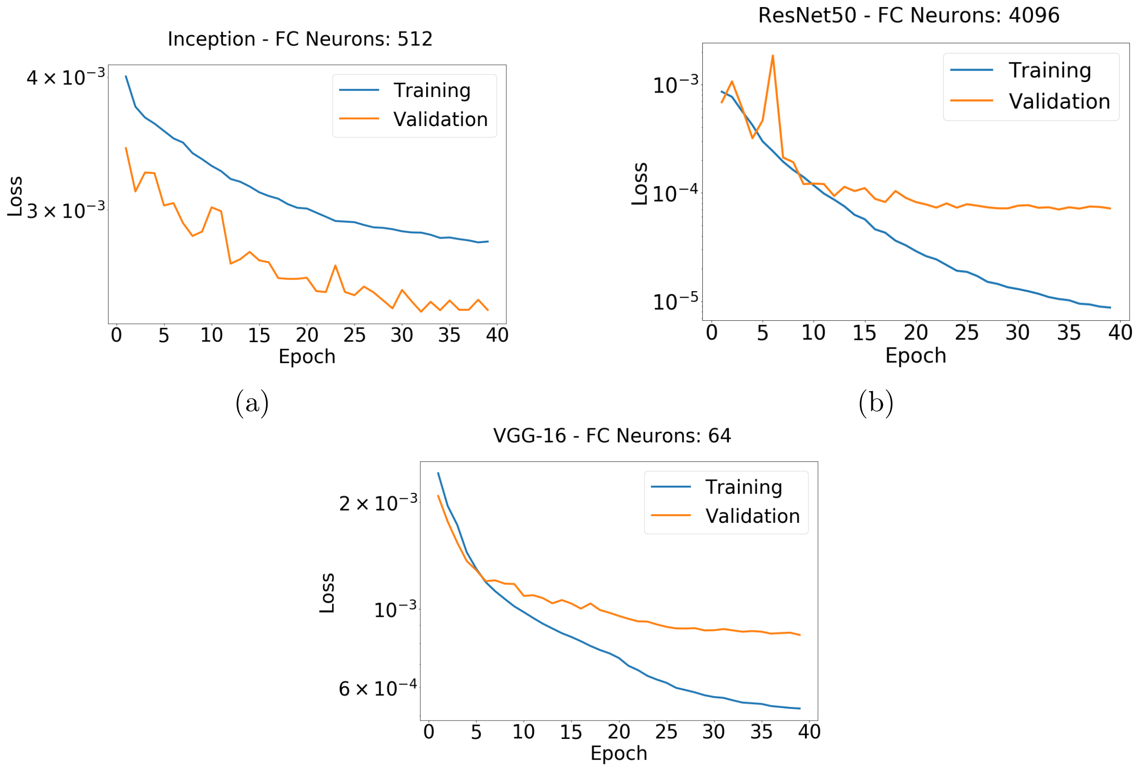
Model	ESPL-LIVE Dataset			TMID Dataset		
	PLCC	SRCC	RMSE	PLCC	SRCC	RMSE
CNN-INCEPTION	0.146	0.136	9.923	0.234	0.201	1.870
CNN-RESNET50	**0.368**	**0.367**	**9.327**	**0.763**	**0.728**	**1.243**
CNN-VGG16	0.243	0.233	9.729	0.610	0.397	1.525
CNN-VGG16-FT	0.314	0.314	9.523	0.681	0.624	1.409

operations use bilinear interpolation. These are the respective image sizes that were used to train the original models in the ImageNet dataset. At first, we trained models for 120 epochs. However, model validation loss value does not improve after 40 epochs. Then, for the experiments presented in this section, we set model training duration to 40 epochs, unless otherwise stated.

Figure 5.5 shows the training and validation curves for the chosen models, and Table 5.5 shows their performance metric values in TMID and ESPL-LIVE datasets. Corresponding scatter plots of each model in both datasets are found in Appendix A. All models achieve better performance values in TMID dataset than in ESPL-LIVE dataset. This is probably a consequence of using TMQI scores as labels, which is also observed in the previous experiment, specially for RankIQA2 and RankIQA5 models. The Inception architecture presents the worst performance in both datasets, with particularly low performance values. ResNet50 outperforms the VGG-16 architecture in both datasets. It also outperforms the RankIQA2 model from the experiment reported in Section 5.1. Similarly with respect to the models considered in this experiment, the RankIQA2 model was trained using TMQI scores as labels for image quality. These results suggest that, for the desired task, the architecture choice may have a significant impact on performance.

We fine-tune the entire VGG-16 and ResNet50 models[2] (that is, parameters from the respective feature extraction parts are now unfrozen and fine-tuned as well) from

[2] We did not fine-tune the Inception model as it performs very poorly in both test datasets, and it performs worse than the other considered models, as shown in Table 5.5.



(a)

(b)

(c)

Figure 5.5: Training and validation loss curves of each model in PBTDB dataset: (a) Inception, (b) ResNet50, and (c) VGG-16

the previous training using the PBTDB dataset. The models are initialized with weight and bias values learned from the previous training. Fine-tuning lasts for 120 epochs, using Adam optimizer with a learning rate set to 10^{-6}, which remains unchanged. Table 5.5 shows the results from the fine-tuned VGG-16 model. We do not show the results for the fine-tuned ResNet50 model because, in both test datasets, its performance values are similar to the ones from the previous ResNet50 model that is not fine-tuned (and is shown in Table 5.5). Minor performance gains over the “CNN-VGG16” model performance are observed, but the fine-tuned model still presents low performance values in ESPL-LIVE dataset. Also, the fine-tuned VGG-16 model performance is not better than the performance of the ResNet50 model. This indicates that the label choice that represents the image quality (in this case, TMQI scores) is likely limiting the model performance.

5.3 Experiment 3: VGG-16 Model Training in Subjectively Labeled PBTDB Subset

In the experiments presented so far in this chapter, we train deep learning models using objective TMIQA metric scores as target quality scores for PBTDB samples.

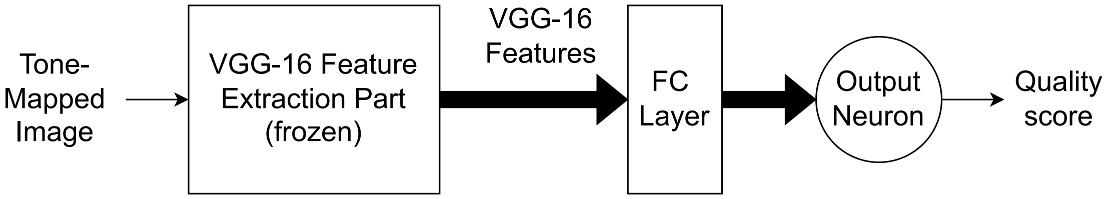
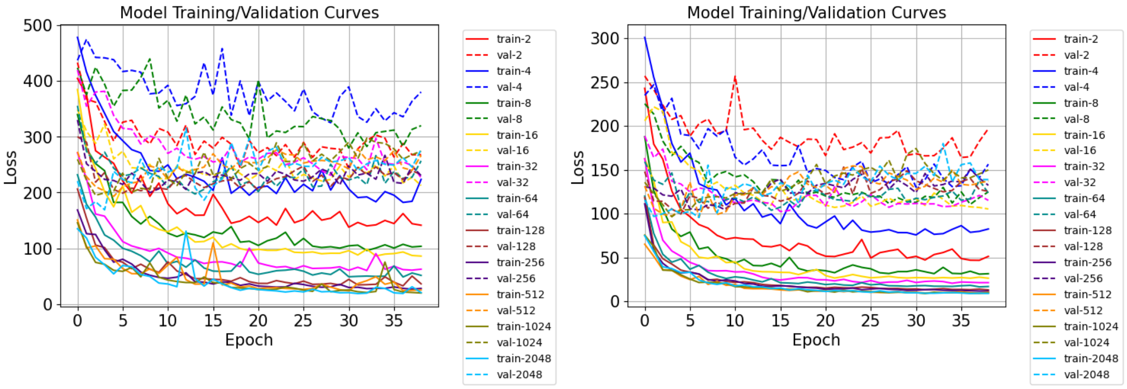


Figure 5.6: Overall model architecture that uses VGG-16 model as feature extractor. Only the FC layer is trained (using subjective MOS values from private-PBTDB and crowd-PBTDB samples as target quality values). Weights and biases from the feature extraction part are kept frozen.

In this section, models are trained using the PBTDB subsets presented in Chapter 4, in which subjective experiments were performed. They are tested in ESPL-LIVE and TMID datasets. Target quality scores correspond to subjective MOS values instead of objective metric scores. We use VGG-16 model as feature extractor, as shown in Figure 5.6, keeping its weights and biases frozen. We replace the original VGG-16 FC layer by a new one, and train this layer only. We also considered the ResNet50 model as feature extractor. ResNet50 model results in ESPL-LIVE and TMID datasets are similar to the ones obtained from the VGG-16 model. To avoid redundant result presentation, in this section, we show VGG-16 results only, as tests were performed with this model initially.

First, the model is trained and tested in the same corresponding PBTDB subsets (private-PBTDB and crowd-PBTDB). In both subsets, we divide samples into three disjoint sets that are used for training, validating and testing the model. We fix the following sample proportions for each set: 80% for training, 10% for validation, and 10% for test. Such division corresponds to 2400 training samples, 300 validation samples and 300 test samples in private-PBTDB, and to 2356 training samples, 294 validation samples and 294 test samples in crowd-PBTDB. Because several PBTDB samples represent repeated scenes mapped by different tone-mapping operators, we make sure all tone-mapped versions of the same scene integrate one set only (either training, validation or test). This avoids overestimating model performance caused by scene contents previously shown to the model during training. Hyperparameter values are the same ones shown in Table 5.4, except for FC layer number of neurons, which we vary from 2 to 2048 in powers of two. Figure 5.7 shows training and validation curves for each model in private-PBTDB and crowd-PBTDB. In both datasets, lowest validation loss is achieved by the model with 16 neurons in the FC layer (yellow dashed line in both plots). We use this configuration in the following experiments of this section.

We train HIGRADE and BTMQI metrics in each PBTDB subset, and compare their performance values with the VGG-16 proposed model. Table 5.6 reports the performance values in the test sets. In private-PBTDB, the proposed VGG-16 model



(a)

(b)

Figure 5.7: Training and validation curves for the models in (a) private-PBTDB and (b) crowd-PBTDB. Solid lines indicate training performance, whereas dashed lines indicate validation performance.

outperforms the best state-of-the-art TMIQA metric (HIGRADE-2) by roughly 4%, considering the SRCC metric. Analyzing the proposed model validation curve from Figure 5.7, we believe its performance can likely be improved by introducing regularization techniques, such as dropout, during training. Nevertheless, these results suggest that VGG-16 features are not well suited for the TMIQA task. This is because simpler TMIQA metrics, which extract features designed for tone mapping quality assessment, are able to achieve similar performance values to the ones from a more complex deep CNN model, which uses the VGG-16 features.

In crowd-PBTDB, the best metric is the HIGRADE-1, outperforming the proposed model by approximately 2% in SRCC metric. Besides the proposed model, this metric also outperforms HIGRADE-2 by a rather large margin (roughly 8% in SRCC). Such behavior is not observed in private-PBTDB, whose samples are almost the same as crowd-PBTDB, but with different MOS values. This result might be caused by experiment inconsistencies, as shown in Figure 4.20(b). We observe that, in crowd-PBTDB, the VGG-16 model improves all its performance values by approximately 4%, as compared with its counterpart in private-PBTDB. This suggests that having more quality labels per sample (on average, 51 labels per sample versus 16 labels per sample in private-PBTDB), which influence sample MOS calculation, leads to a better training process. More labels provide more different MOS values and thus smoother distributions.

Next, we train the chosen VGG-16 model using subjective MOS values from each PBTDB subset, and test it in different databases, namely ESPL-LIVE and TMID. In this experiment, we investigate how generic are the subjective MOS values. Comparatively, we train the same VGG-16 model, in each PBTDB subset, but using corresponding sample TMQI scores as target quality scores, instead of subjective MOS values. We examine the impacts on model generalization capabilities when

Table 5.6: Performance values of TMIQA models in the private-PBTDB and crowd-PBTDB datasets. The VGG-16 model is identified as "CNN-VGG16-16N" because it contains 16 neurons in the FC layer. Boldface text marks the best result for each performance metric in the corresponding dataset.

Model	**Private-PBTDB**			**Crowd-PBTDB**		
	PLCC	**SRCC**	**RMSE**	**PLCC**	**SRCC**	**RMSE**
CNN-VGG16-16N	**0.691**	**0.728**	13.324	0.734	0.768	9.8747
HIGRADE-1	0.663	0.671	13.322	**0.797**	**0.782**	**8.667**
HIGRADE-2	**0.691**	0.684	**12.860**	0.746	0.707	9.322
BTMQI	0.627	0.631	13.860	0.719	0.703	9.732

different target quality scores are used (*i.e.* subjective scores versus objective scores).

Table 5.7 shows the trained model performance values in each test database. We did not apply the non-linear fit function to the model predicted scores. Therefore, RMSE metric is not shown, as TMQI and subjective MOS values have different score ranges. The models present poor performance values in both databases, indicating that quality prediction of generic tone-mapped images is still unreliable. However, we observe that models trained with subjective MOS labels perform better than their counterparts trained with TMQI scores, in each respective database (private-PBTDB and crowd-PBTDB). Particularly, training with MOS values from the private group increases model performance values by about 6% in ESPL-LIVE and 11% in TMID in comparison with using TMQI scores as target values for training. This shows that subjective MOS values are more reliable quality labels than TMQI scores (and, potentially, other objective TMIQA metric scores). We hypothesize that a reason for poor performance, in this case, is likely the lack of more subjectively labeled samples, and not the sample quality score nature.

The VGG-16 model trained with subjective MOS values from crowd-PBTDB performs worse than the model trained with subjective MOS values from private-PBTDB in predicting quality of ESPL-LIVE and TMID samples. Considering the crowd-PBTDB and private-PBTDB VGG model tests in their own respective datasets (Table 5.6), we observed that crowd-PBTDB VGG model achieved better results. Despite these results, such model has weaker generalization power than the model from private-PBTDB. This strengthens the idea that MOS values from crowd group experiment are less reliable for predicting generic tone-mapped image quality than the ones from private group experiment. Crowd MOS unreliability is arguably caused by inconsistencies still present in such experiment, as opposed to private MOS values, which come from an entirely consistent experiment.

Finally, we investigate the addition of one extra FC layer to the deep learning model. The idea is to verify if manipulating VGG features leads to a feature space that facilitates the regression task. The model regression part is now represented

Table 5.7: Performance values of VGG-16 models trained in PBTDB subsets, and tested in ESPL and TMID datasets. Quality scores used to train VGG-16 models can correspond to subjective MOS values or TMQI score values. Predicted scores are not adjusted by a non-linear fit function, so better performance values are close to 1 in ESPL-LIVE dataset, and -1 in TMID dataset. Boldface text marks the best result for each performance metric in the corresponding dataset.

Model	**ESPL-LIVE Dataset**		**TMID Dataset**	
	PLCC	**SRCC**	**PLCC**	**SRCC**
CNN-VGG16-16N-private-MOS	**0.203**	**0.195**	**-0.336**	**-0.318**
CNN-VGG16-16N-private-TMQI	0.141	0.129	-0.228	-0.201
CNN-VGG16-16N-crowd-MOS	0.191	0.179	-0.298	-0.244
CNN-VGG16-16N-crowd-TMQI	0.023	0.020	-0.222	-0.197

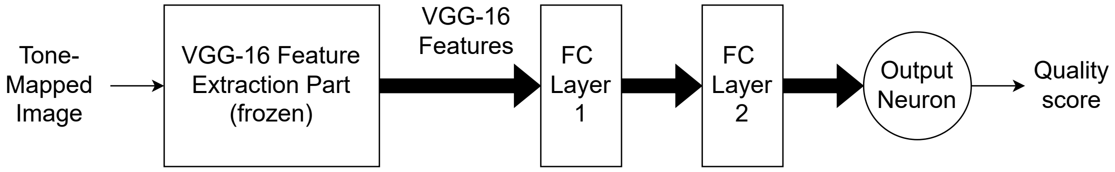


Figure 5.8: Overall model architecture that uses VGG-16 model as feature extractor, and two FC layers as the regression part. Only the two FC layers are trained (using private-PBTDB subjective MOS values as target quality values). Weights and biases from the feature extraction part are kept frozen.

by two FC layers, as shown in Figure 5.8. We fix 16 neurons in the first FC layer, and vary the number of neurons in the second FC layer from 2 to 128 in powers of two. All models are trained in the private-PBTDB, using subjective MOS values as target scores. Only model FC layers are trained (the feature extraction part is kept frozen). Other training hyperparameters values are the ones shown in Table 5.4. Figure 5.9 shows the training and validation curves for different number of neurons in the second FC layer. The chosen model for this experiment is the one with 128 neurons in the second FC layer, as it achieves the lowest validation error after 40 epochs (brown dashed line).

Table 5.8 shows the performance values in the test subset of private-PBTDB, and in ESPL-LIVE and TMID databases. Comparing these results with the ones reported in Tables 5.6 and 5.7, we note that, in all considered databases, the model with two FC layers performs similarly as the model with one FC layer only. This reinforces the previous observation that VGG features are not suited for TMIQA, and they limit model performance.

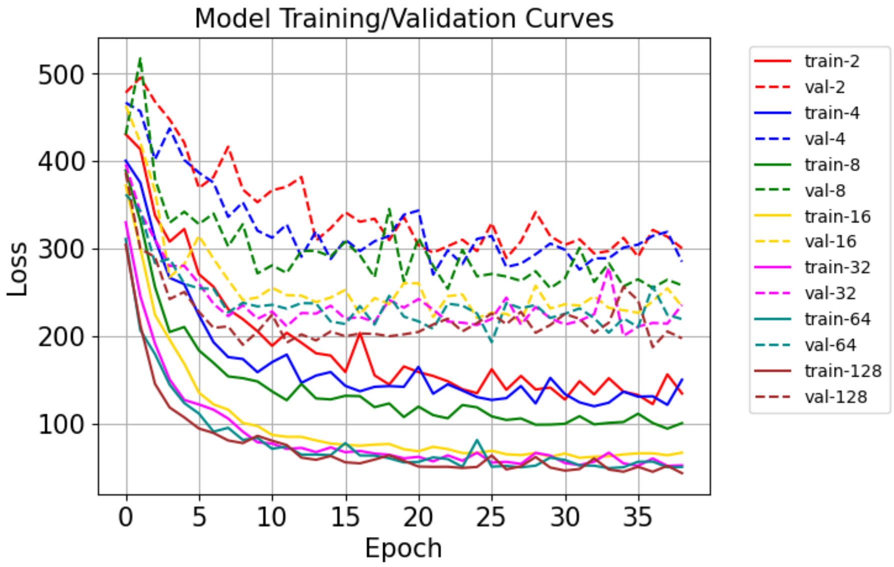


Figure 5.9: Training and validation curves for the 2-FC layer models in private-PBTDB. Each curve corresponds to a different number of neurons in the second FC layer, varying from 2 to 128 in powers of two (all models have 16 neurons in the first FC layer). Solid lines indicate training performance, whereas dashed lines indicate validation performance.

Table 5.8: Test performance values of the VGG-16 model that has two FC layers in its regression part. The FC layer contains 16 neurons, and the second FC layer has 128 neurons. Predicted scores are not adjusted by a non-linear fit function when model is tested in ESPL-LIVE and TMID datasets. Therefore, better performance values are close to 1 in ESPL-LIVE dataset, and -1 in TMID dataset.

Database	**PLCC**	**SRCC**
Private-PBTDB	0.700	0.730
ESPL-LIVE	0.226	0.218
TMID	-0.365	-0.325

5.4 Chapter Summary

In this chapter, we presented three experiments in which PBTDB was used to train several deep learning architectures. In the first experiment, we fine-tuned different versions of the RankIQA model. Each version considered scores from a different TMIQA metric as target quality values. In the second experiment, we trained networks based on traditional deep CNN architectures (namely VGG-16, Resnet50 and Inception) for the TMIQA task. We used TMQI scores as target quality values. We removed the original FC layers from such models, and trained a newly introduced FC layer for the task at hand. In the third experiment, we trained deep learning models based on VGG-16 using subjective MOS values from private-PBTDB and crowd-PBTDB subsets as target quality values.

We draw three conclusions from the results presented in this chapter. The first one is that the objective quality score used to train the deep learning model impacts on its performance. Performance values from the deep learning models are not better than the performance values from the respective individual TMIQA metric whose scores are used as target quality values in model training. In fact, deep learning model performance values reflect the limitations from the considered TMIQA metrics in each case. The network architecture may also have an impact on its performance, as shown in Table 5.5.

The second conclusion is that the lack of more subjectively labeled samples limits performance of deep learning models that are trained with subjective MOS values as target quality scores. This is most likely the main cause for the overall poor performance values in TMID and ESPL-LIVE datasets, rather than limitations intrinsic to MOS values from private-PBTDB and crowd-PBTDB subsets. Nevertheless, models trained with subjective MOS values from private-PBTDB achieve better performance values, when evaluating the quality of ESPL-LIVE and TMID samples, than models trained with subjective MOS values from crowd-PBTDB (Table 5.7). This suggests that private-PBTDB subjective MOS values are better tone-mapping quality indicators than crowd-PBTDB subjective MOS values. Arguably, this is because all samples in private-PBTDB are consistent (*i.e.* have typical label distributions), as opposed to crowd-PBTDB, which contains some inconsistent samples. Using private-PBTDB MOS values as target quality values in training, instead of crowd-PBTDB MOS values, leads to models with better generalization capabilities.

The last conclusion is that VGG-16 features (and, possibly, features from other traditional CNN architectures, such as ResNet50) do not seem suited for the TMIQA task. New architectures should be explored specifically for TMIQA, so that more adequate features to this task can be discovered and learned by the models.

Chapter 6

Conclusions and Future Work

Currently available state-of-the-art TMIQA metrics yield reliable quality scores for many tone-mapped image categories, but there is significant room for improvement. In particular, for tone-mapped images that do not belong to specific databases, the TMIQA metrics do not yield suitable quality labels. Only two benchmark databases designed for TMIQA are available in the literature (ESPL-LIVE and TMID), and they contain a relatively small number of samples. This likely causes the poor generalization capabilities of state-of-the-art TMIQA metrics, which are trained in such databases. Other IQA metrics, designed for evaluating generic distortions or aesthetics, do not perform well in the TMIQA task either.

We presented experiments in which regression and classification models are trained using quality scores from several selected IQA metrics as features to assess image quality. The assumption was that a "commitee" of IQA metrics can provide a more reliable quality assessment than any individual metric. We showed that models achieve reasonable performance values when they are trained and tested with tone-mapped images that come from the same database (in this case, ESPL-LIVE). However, these models perform poorly when trained with samples from one database (ESPL-LIVE) and tested with samples from another one (TMID). Such limitation is also observed in individual IQA metrics, and the IQA metric "commitee" we considered is also unable to overcome it. Slightly better performance values in cross-database experiments presented in this work are achieved when regression models combine quality features extracted from different TMIQA metrics. Still, best performance values in TMID database come from a single TMIQA metric (BTMQI), and this metric greatly outperforms these regression models.

We repeated the regression model experiments, but now considering that corresponding training and test databases contain samples from both ESPL-LIVE and TMID databases. We observed that regression models trained with a mixed database achieve better test performance values than their counterparts that are trained using ESPL-LIVE samples only, and tested in TMID samples only. This indicates that samples from each database contain different properties that are relevant for assessing their quality, and likely explains the model generalization power improvement when mixed training and test databases are considered. However, such regression models achieve performance values that are similar to or inferior than the ones from individual TMIQA metrics that are trained and tested in the same respective mixed databases. These results do not support the claim that using more complex regression models in replacement of a single (and simpler) TMIQA metric leads to more reliable tone-mapping quality evaluations.

We introduced a new tone-mapped database for TMIQA, called "PBTDB", which contains approximately 175000 samples. Each sample has four quality scores obtained from different state-of-the-art TMIQA metrics (HIGRADE-1, HIGRADE-2, BTMQI and TMQI). We showed that PBTDB contains a wide scene type diversity, which is measured both objectively (in terms of brightness, colorfulness and scene complexity attributes) and subjectively (in terms of TMIQA metric score distributions). We selected 3009 representative samples from PBTDB, and performed remote subjective tests to assess their quality. Two subject groups were considered: a smaller "private" group, in which participants are directly invited by authors, and a larger "crowd" group, in which participants are recruited from an AWS online "crowdsourcing" platform. We followed the procedure in [82] to check if the subjective experiments are consistent, and thus ensure that labels from both subject groups are trustworthy. Inconsistency is defined by a case when, for a given p-value threshold, the observed proportion of samples with atypical label distributions exceed the maximum expected proportion of such samples for that threshold value. Each detected inconsistent sample was treated individually by taking one out of three pre-defined actions. After performing such actions, we showed that labels from the private group are entirely consistent, whereas labels from the crowd group still presented some minor inconsistencies. We conclude that additional mechanisms that raise subject attention to the task at hand, other than the simple GSQ approach adopted in this work, may be required when designing large-scale crowdsourcing subjective experiments.

Finally, we conducted three experiments that involve training deep learning models for TMIQA using PBTDB. In the first experiment, we fine-tuned several versions of the base architecture defined in the RankIQA metric for the TMIQA task. Each version was fine-tuned with PBTDB samples using a different TMIQA metric scores to represent sample target quality value. All versions were then tested in ESPL-LIVE and TMID databases. None of the models was able to achieve good performance values in both databases simultaneously. In fact, their performances highly correlate with the performances from individual TMIQA metrics whose scores are used to fine-tune the respective RankIQA models. Therefore, generalization capability of RankIQA models is probably limited by the nature of the scores that represent sample quality. In each database, the corresponding best RankIQA versions are still outperformed by a single TMIQA metric (HIGRADE-2 in ESPL-LIVE, and BTMQI in TMID).

In the second deep learning experiment, we replaced the FC layers of traditional deep CNN models (VGG-16, ResNet50, and Inception) by one new FC layer. We kept convolutional layers frozen, and trained the FC layer only for the TMIQA task, using PBTDB samples. TMQI scores were chosen as sample target quality values. We noted that all models achieved better results in TMID than in ESPL-LIVE. This reinforces the idea that model performance relies heavily on the metric used to represent sample quality, as TMQI metric scores are highly correlated with TMID MOS values. Also, in both databases, ResNet50 model performed consistently better than other models, whereas Inception model performed particularly poorly. This suggests that model architecture may significantly influence performance.

In the last deep learning experiment, we followed the same training procedure from the second experiment, but considered as training databases the PBTDB subsets in which subjective experiments were performed. We considered the VGG-16 model as the base feature extractor architecture. We showed that using subjective MOS values either from crowd group or from private group as sample quality scores to train the models led to better performance values in test databases (ESPL-LIVE and TMID) than using TMQI scores as quality indicators. However, performance values are very poor in both databases. We believe that, in this case, lack of more subjectively labeled samples is compromising model performance. Moreover, models trained with private group MOS values performed better than models trained with crowd group MOS values in both test databases. MOS values computed from fewer but more consistent quality labels (private group case) are more reliable quality indicators for generic tone-mapped samples than MOS values obtained from many but less consistent quality labels (crowd group case). This indicates that large-scale subjective experiments design should primarily focus on strict mechanisms that discourage "cheating" behaviors (thereby ensuring more consistent participants), rather than recruiting a large number of subjects while using no such mechanisms or very simple ones.

As future work directions, one possibility is to explore autolabeling techniques, such as the one proposed in [97], to increase the number of PBTDB samples labeled with more reliable quality scores. Private-PBTDB subset can be used as a starting point for training models that learn to predict their subjective MOS values. Then, an iterative training process would take place as follows. Samples that have their quality reliably predicted by the best trained model from the current iteration are integrated in the training database that contains the private-PBTDB samples. The resulting database is used to train regression models in the next iteration.

Another possibility is to change the methodology for treating inconsistent samples in the crowd experiment. In particular, because private and crowd groups evaluated the same sample set, we can assign private group MOS values as the expected sample quality values. Then, samples with crowd MOS values that strongly deviate from the expected ones are filtered out. Such procedure is similar to the one adopted in [50] to identify and remove outlier samples. Such approach may lead to an entirely consistent crowd group experiment, which arguably provides more accurate sample MOS values. This is because the MOS values from the crowd group are computed over more quality labels (that are now consistent) than the MOS values from the private group. Models trained with consistent crowd MOS values might achieve better results in test databases.

Other direction is to investigate new architectures that are not necessarily based on traditional deep CNN models, as VGG features (and, potentially, ResNet50 features) do not seem suited for TMIQA. Ranking tone-mapped samples according to their quality may be a promising training approach. Samples in the private-PBTDB and crowd-PBTDB subsets can be ranked according to their subjective MOS values. In this training approach, a “sample” actually corresponds to a pair of images. Therefore, an effective large training database can be obtained, if we consider all possible image pairs from the aforementioned PBTDB subset. Architectures that are adapted to this training method, such as the Siamese networks from RankIQA, can be explored.

References

[1] H. SEETZEN AND W. HEIDRICH AND W. STUERZLINGER AND G. WARD AND L. WHITEHEAD AND M. TRENTACOSTE AND A. GHOSH AND A. VOROZCOVS. "High dynamic range display systems". In: *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, pp. 760–768, New York, NY, USA, 2004. ACM. doi: 10.1145/1186562.1015797.

[2] E. REINHARD AND G. WARD AND S. PATTANAIK AND P. DEBEVEC. *High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting (The Morgan Kaufmann Series in Computer Graphics)*. San Francisco, CA, USA, Morgan Kaufmann Publishers Inc., 2005. ISBN: 0125852630.

[3] M. ČADÍK AND M. WIMMER AND L. NEUMANN AND A. ARTUSI. "Evaluation of HDR tone mapping methods using essential perceptual attributes", *Computers & Graphics*, v. 32, n. 3, pp. 330 – 349, 2008. ISSN: 0097-8493. doi: 10.1016/j.cag.2008.04.003. Disponível em: <`http://www.sciencedirect.com/science/article/pii/S0097849308000460`>.

[4] WARD, G. "Graphics Gems IV". Academic Press Professional, Inc., cap. A Contrast-based Scalefactor for Luminance Display, pp. 415–421, San Diego, CA, USA, 1994. ISBN: 0-12-336155-9. Disponível em: <`http://dl.acm.org/citation.cfm?id=180895.180934`>.

[5] FERWERDA, J. A., PATTANAIK, S. N., SHIRLEY, P., et al. "A Model of Visual Adaptation for Realistic Image Synthesis". In: *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, pp. 249–258, New York, NY, USA, 1996. ACM. ISBN: 0-89791-746-4. doi: 10.1145/237170.237262. Disponível em: <`http://doi.acm.org/10.1145/237170.237262`>.

[6] LARSON, G. W., RUSHMEIER, H., PIATKO, C. "A visibility matching tone reproduction operator for high dynamic range scenes", *IEEE Transactions on Visualization and Computer Graphics*, v. 3, n. 4, pp. 291–306, Oct 1997. ISSN: 1077-2626. doi: 10.1109/2945.646233.

[7] KUANG, J., JOHNSON, G. M., FAIRCHILD, M. D. "iCAM06: A refined image appearance model for HDR image rendering", *Journal of Visual Communication and Image Representation*, v. 18, n. 5, pp. 406 – 414, 2007. ISSN: 1047-3203. doi: https://doi.org/10.1016/j.jvcir.2007.06.003. Disponível em: <`http://www.sciencedirect.com/science/article/pii/S1047320307000533`>. Special issue on High Dynamic Range Imaging.

[8] X. CERDA-COMPANY AND C. A. PARRAGA AND X. OTAZU. "Which tone-mapping operator is the best? A comparative study of perceptual quality", *J. Opt. Soc. Am. A*, v. 35, n. 4, pp. 626–638, Apr 2018. doi: 10.1364/JOSAA.35.000626. Disponível em: <`http://josaa.osa.org/abstract.cfm?URI=josaa-35-4-626`>.

[9] E. REINHARD AND M. STARK AND P. SHIRLEY AND J. FERWERDA. "Photographic tone reproduction for digital images", *ACM Trans. Graph.*, v. 21, n. 3, pp. 267–276, jul. 2002. ISSN: 0730-0301. doi: 10.1145/566654.566575.

[10] TADE, S. L., VYAS, V. "Tone Mapped High Dynamic Range Image Quality Assessment Techniques: Survey and Analysis", *Archives of Computational Methods in Engineering*, Apr 2020. ISSN: 1886-1784. doi: 10.1007/s11831-020-09428-y. Disponível em: <`https://doi.org/10.1007/s11831-020-09428-y`>.

[11] CHANDLER, D. M., LI, S., LIN, C. S., et al. "Seven Challenges in Image Quality Assessment: Past, Present, and Future Research", *ISRN Signal Processing*, v. 2013, pp. 905685 – 905737, 2013. doi: 10.1155/2013/905685.

[12] H. YEGANEH AND Z. WANG. "Objective quality assessment of tone-mapped images", *IEEE Transactions on Image Processing*, v. 22, n. 2, pp. 657–667, Feb 2013. ISSN: 1057-7149. doi: 10.1109/TIP.2012.2221725.

[13] GU, K., WANG, S., ZHAI, G., et al. "Blind Quality Assessment of Tone-Mapped Images Via Analysis of Information, Naturalness, and Structure", *IEEE Transactions on Multimedia*, v. 18, n. 3, pp. 432–443, 2016.

[14] KUNDU, D., GHADIYARAM, D., BOVIK, A. C., et al. "No-reference quality assessment of tone-mapped HDR pictures", *IEEE Transactions on Image Processing*, v. 26, n. 6, pp. 2957–2971, jun 2017. ISSN: 10577149. doi: 10.1109/TIP.2017.2685941.

[15] NUNES, G. M. S., OLIVEIRA, F. D. V. R., FARIAS, M. C. Q., et al. "Comparison between Digital Tone-Mapping Operators and a Focal-Plane Pixel-Parallel Circuit", *Signal Processing: Image Communication*, v. 88, pp. 115937, 2020. ISSN: 0923-5965. doi: https://doi.org/10.1016/j.image.2020.115937. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0923596520301235>.

[16] D. HASLER AND S. SÜSSTRUNK. "Measuring colourfulness in natural images", *Proc. IS&T/SPIE Electronic Imaging 2003: Human Vision and Electronic Imaging VIII*, v. 5007, pp. 87–95, 2003. doi: 10.1117/12.477378. Disponível em: <http://infoscience.epfl.ch/record/33994>.

[17] LIU, X., VAN DE WEIJER, J., BAGDANOV, A. D. "RankIQA: Learning From Rankings for No-Reference Image Quality Assessment". In: *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017.

[18] TALEBI, H., MILANFAR, P. "NIMA: Neural Image Assessment", *IEEE Transactions on Image Processing*, v. 27, n. 8, pp. 3998–4011, 2018.

[19] REN, J., SHEN, X., LIN, Z., et al. "Personalized Image Aesthetics". In: *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 638–647, 2017.

[20] HE, Q., LI, D., JIANG, T., et al. "Quality Assessment for Tone-Mapped HDR Images Using Multi-Scale and Multi-Layer Information". In: *2018 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pp. 1–6, 2018.

[21] RAVURI, C. S., SUREDDI, R., REDDY DENDI, S. V., et al. "Deep No-Reference Tone Mapped Image Quality Assessment". In: *2019 53rd Asilomar Conference on Signals, Systems, and Computers*, pp. 1906–1910, 2019.

[22] HE, K., ZHANG, X., REN, S., et al. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[23] RONNEBERGER, O., FISCHER, P., BROX, T. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, v. 9351, *LNCS*, pp. 234–241. Springer, 2015. Disponível em: <http://lmb.informatik.uni-freiburg.de/Publications/2015/RFB15a>. (available on arXiv:1505.04597 [cs.CV]).

[24] AYDIN, T. O., MANTIUK, R., MYSZKOWSKI, K., et al. "Dynamic Range Independent Image Quality Assessment", *ACM Trans. Graph.*, v. 27, n. 3, pp. 1–10, ago. 2008. ISSN: 0730-0301. doi: 10.1145/1360612.1360668. Disponível em: <`https://doi.org/10.1145/1360612.1360668`>.

[25] SMOLA, A. J., SCHÖLKOPF, B. "A tutorial on support vector regression", *Statistics and Computing*, v. 14, n. 3, pp. 199–222, Aug 2004. ISSN: 1573-1375. doi: 10.1023/B:STCO.0000035301.49549.88. Disponível em: <`https://doi.org/10.1023/B:STCO.0000035301.49549.88`>.

[26] YUE, G., HOU, C., ZHOU, T. "Blind Quality Assessment of Tone-Mapped Images Considering Colorfulness, Naturalness, and Structure", *IEEE Transactions on Industrial Electronics*, v. 66, n. 5, pp. 3784–3793, may 2019. ISSN: 02780046. doi: 10.1109/TIE.2018.2851984.

[27] YUE, G., HOU, C., GU, K., et al. "Biologically inspired blind quality assessment of tone-mapped images", *IEEE Transactions on Industrial Electronics*, v. 65, n. 3, pp. 2525–2536, mar 2018. ISSN: 02780046. doi: 10.1109/TIE.2017.2739708.

[28] JIANG, Q., SHAO, F., LIN, W., et al. "BLIQUE-TMI: Blind Quality Evaluator for Tone-Mapped Images Based on Local and Global Feature Analyses", *IEEE Transactions on Circuits and Systems for Video Technology*, v. 29, n. 2, pp. 323–335, 2019.

[29] HUANG, G., ZHU, Q., SIEW, C. "Extreme learning machine: Theory and applications", *Neurocomputing*, v. 70, n. 1, pp. 489 – 501, 2006. ISSN: 0925-2312. doi: https://doi.org/10.1016/j.neucom.2005.12.126. Disponível em: <`http://www.sciencedirect.com/science/article/pii/S0925231206000385`>. Neural Networks.

[30] ZHANG, Y., CHANDLER, D. M. "Opinion-Unaware Blind Quality Assessment of Multiply and Singly Distorted Images via Distortion Parameter Estimation", *IEEE Transactions on Image Processing*, v. 27, n. 11, pp. 5433–5448, 2018.

[31] MOORTHY, A. K., BOVIK, A. C. "A Two-Step Framework for Constructing Blind Image Quality Indices", *IEEE Signal Processing Letters*, v. 17, n. 5, pp. 513–516, 2010.

[32] SAAD, M. A., BOVIK, A. C., CHARRIER, C. "Blind Image Quality Assessment: A Natural Scene Statistics Approach in the DCT Domain", *IEEE Transactions on Image Processing*, v. 21, n. 8, pp. 3339–3352, 2012.

[33] LIU, L., HUA, Y., ZHAO, Q., et al. "Blind image quality assessment by relative gradient statistics and adaboosting neural network", *Signal Processing: Image Communication*, v. 40, pp. 1 – 15, 2016. ISSN: 0923-5965. doi: https://doi.org/10.1016/j.image.2015.10.005. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0923596515001708>.

[34] MITTAL, A., MOORTHY, A. K., BOVIK, A. C. "No-Reference Image Quality Assessment in the Spatial Domain", *IEEE Transactions on Image Processing*, v. 21, n. 12, pp. 4695–4708, 2012.

[35] ZHANG, Y., MOORTHY, A. K., CHANDLER, D. M., et al. "C-DIIVINE: No-reference image quality assessment based on local magnitude and phase statistics of natural scenes", *Signal Processing: Image Communication*, v. 29, n. 7, pp. 725 – 747, 2014. ISSN: 0923-5965. doi: https://doi.org/10.1016/j.image.2014.05.004. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0923596514000836>.

[36] YE, P., KUMAR, J., KANG, L., et al. "Unsupervised feature learning framework for no-reference image quality assessment". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1098–1105, 2012.

[37] ZHANG, Y., CHANDLER, D. M. "An algorithm for no-reference image quality assessment based on log-derivative statistics of natural scenes". In: Burns, P. D., Triantaphillidou, S. (Eds.), *Image Quality and System Performance X*, v. 8653, pp. 156 – 165. International Society for Optics and Photonics, SPIE, 2013. doi: 10.1117/12.2001342. Disponível em: <https://doi.org/10.1117/12.2001342>.

[38] MOORTHY, A. K., BOVIK, A. C. "Blind Image Quality Assessment: From Natural Scene Statistics to Perceptual Quality", *IEEE Transactions on Image Processing*, v. 20, n. 12, pp. 3350–3364, 2011.

[39] CHEN, X., ZHANG, Q., LIN, M., et al. "No-reference color image quality assessment: from entropy to perceptual quality", *EURASIP Journal on Image and Video Processing*, v. 2019, n. 1, pp. 77, Sep 2019. ISSN: 1687-5281. doi: 10.1186/s13640-019-0479-7. Disponível em: <https://doi.org/10.1186/s13640-019-0479-7>.

[40] GHADIYARAM, D., BOVIK, A. C. "Perceptual quality prediction on authentically distorted images using a bag of features approach", *Journal of Vision*, v. 17, n. 1, pp. 32–32, 01 2017. ISSN: 1534-7362. doi: 10.1167/17.1.32. Disponível em: <https://doi.org/10.1167/17.1.32>.

[41] XUE, W., MOU, X., ZHANG, L., et al. "Blind Image Quality Assessment Using Joint Statistics of Gradient Magnitude and Laplacian Features", *IEEE Transactions on Image Processing*, v. 23, n. 11, pp. 4850–4862, 2014.

[42] ZHANG, L., ZHANG, L., BOVIK, A. C. "A Feature-Enriched Completely Blind Image Quality Evaluator", *IEEE Transactions on Image Processing*, v. 24, n. 8, pp. 2579–2591, 2015.

[43] GU, K., ZHAI, G., YANG, X., et al. "Using Free Energy Principle For Blind Image Quality Assessment", *IEEE Transactions on Multimedia*, v. 17, n. 1, pp. 50–63, 2015.

[44] MITTAL, A., SOUNDARARAJAN, R., BOVIK, A. C. "Making a Completely Blind Image Quality Analyzer", *IEEE Signal Processing Letters*, v. 20, n. 3, pp. 209–212, 2013.

[45] GU, K., LIN, W., ZHAI, G., et al. "No-Reference Quality Metric of Contrast-Distorted Images Based on Information Maximization", *IEEE Transactions on Cybernetics*, v. 47, n. 12, pp. 4559–4565, 2017.

[46] GOLESTANEH, S. A., CHANDLER, D. M. "No-Reference Quality Assessment of JPEG Images via a Quality Relevance Map", *IEEE Signal Processing Letters*, v. 21, n. 2, pp. 155–158, 2014.

[47] LIU, X., VAN DE WEIJER, J., BAGDANOV, A. D. "Exploiting Unlabeled Data in CNNs by Self-supervised Learning to Rank", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2019. ISSN: 0162-8828. doi: 10.1109/TPAMI.2019.2899857.

[48] GU, K., ZHAI, G., YANG, X., et al. "Hybrid No-Reference Quality Metric for Singly and Multiply Distorted Images", *IEEE Transactions on Broadcasting*, v. 60, n. 3, pp. 555–567, 2014.

[49] LIU, L., LIU, B., HUANG, H., et al. "No-reference image quality assessment based on spatial and spectral entropies", *Signal Processing: Image Communication*, v. 29, n. 8, pp. 856 – 863, 2014. ISSN: 0923-5965. doi: https://doi.org/10.1016/j.image.2014.06.006. Disponível em: <`http://www.sciencedirect.com/science/article/pii/S0923596514000927`>.

[50] KUNDU, D., GHADIYARAM, D., BOVIK, A. C., et al. "Large-Scale Crowdsourced Study for Tone-Mapped HDR Pictures", *IEEE Transactions on Image Processing*, v. 26, n. 10, pp. 4725–4740, Oct 2017. ISSN: 1941-0042. doi: 10.1109/TIP.2017.2713945.

[51] VQEG. "Final Report from the Video Quality Experts Group on Validation of Objective Models of Video Quality Assessment, Phase II (FR-TV2)". `https://www.its.bldrdoc.gov/vqeg/projects/frtv-phase-ii/frtv-phase-ii.aspx`, . Last Accessed 11 August 2020.

[52] ROHALY, A. M., CORRIVEAU, P. J., LIBERT, J. M., et al. "Video Quality Experts Group: current results and future directions". In: Ngan, K. N., Sikora, T., Sun, M.-T. (Eds.), *Visual Communications and Image Processing 2000*, v. 4067, pp. 742 – 753. International Society for Optics and Photonics, SPIE, 2000. doi: 10.1117/12.386632. Disponível em: <`https://doi.org/10.1117/12.386632`>.

[53] VQEG. "Tutorial: Objective Perceptual Assessment of Video Quality: Full Reference Television". `https://www.itu.int/ITU-T/studygroups/com09/docs/tutorial_opavc.pdf`, . Last Accessed 09 September 2020.

[54] HASTIE, T., TIBSHIRANI, R., FRIEDMAN, J. *The Elements of Statistical Learning*. 2 ed. Manhattan, New York City, Springer-Verlag New York, 2009. ISBN: 978-0-387-84858-7. doi: 10.1007/978-0-387-84858-7.

[55] CHOUDHURY, A., DALY, S. "Combining Quality Metrics using Machine Learning for improved and robust HDR Image Quality Assessment", *Electronic Imaging*, v. 2019, n. 10, pp. 307–1–307–7, 2019. ISSN: 2470-1173. doi: doi:10.2352/ISSN.2470-1173.2019.10.IQSP-307. Disponível em: <`https://www.ingentaconnect.com/content/ist/ei/2019/00002019/00000010/art00009`>.

[56] CHOUDHURY, A., DALY, S. "Advantages of Incorporating Perceptual Component Models into a Machine Learning framework for Prediction of Display Quality", *Electronic Imaging*, v. 2018, n. 12, pp. 299–1–299–6, 2018. ISSN: 2470-1173. doi: doi:10.2352/ISSN.2470-1173.2018.12.IQSP-299. Disponível em: <`https://www.ingentaconnect.com/content/ist/ei/2018/00002018/00000012/art00013`>.

[57] WHITNEY, A. W. "A Direct Method of Nonparametric Measurement Selection", *IEEE Transactions on Computers*, v. C-20, n. 9, pp. 1100–1103, 1971.

[58] REFAEILZADEH, P., TANG, L., LIU, H. "Cross-Validation". In: LIU, L., ÖZSU, M. T. (Eds.), *Encyclopedia of Database Systems*, pp. 532–538, Boston, MA, Springer US, 2009. ISBN: 978-0-387-39940-9. doi: 10.1007/978-0-387-39940-9_565.

[59] BUITINCK, L., LOUPPE, G., BLONDEL, M., et al. "API design for machine learning software: experiences from the scikit-learn project". In: *ECML PKDD Workshop: Languages for Data Mining and Machine Learning*, pp. 108–122, 2013.

[60] VORAN, S. D. "Iterated Nested Least-Squares Algorithm for Fitting Multiple Data Sets". Technical Report, PB2003-101416; NTIA-TM-03-397 Assistant Secretary for Communications and Information, out. 2002.

[61] SIMONYAN, K., ZISSERMAN, A. "Very Deep Convolutional Networks for Large-Scale Image Recognition". In: *International Conference on Learning Representations*, 2015.

[62] MANTIUK, R. K., MYSZKOWSKI, K., SEIDEL, H. "High Dynamic Range Imaging". In: *Wiley Encyclopedia of Electrical and Electronics Engineering*, pp. 1–42, American Cancer Society, 2015. ISBN: 9780471346081. doi: 10.1002/047134608X.W8265. Disponível em: <`https://onlinelibrary.wiley.com/doi/abs/10.1002/047134608X.W8265`>.

[63] NEMOTO, H., KORSHUNOV, P., HANHART, P., et al. "Visual attention in LDR and HDR images", 2015. Disponível em: <`http://www.epfl.ch/labs/mmspg/hdr-eye`>. Last Accessed 15 July 2020.

[64] XIAO, F., DICARLO, J. M., CATRYSSE, P. B., et al. "High dynamic range imaging of natural scenes". In: *in The Tenth Color Imaging Conference*, pp. 337–342. Elsevier, Morgan Kaufmann, 2002. Disponível em: <`http://scarlet.stanford.edu/~brian/hdr/hdr.html`>. Last Accessed 15 July 2020.

[65] "Ward's High Dynamic Range Images Examples". `http://www.anyhere.com/gward/hdrenc/pages/originals.html`. Last Accessed 15 July 2020.

[66] FUNT, B. V., SHI, L. "The Rehabilitation of MaxRGB." In: *Color Imaging Conference*, pp. 256–259. IS&T - The Society for Imaging Science and Technology, 2010. ISBN: 978-0-89208-294-0. Disponível em: <`https://www2.cs.sfu.ca/~colour/data/funt_hdr/`>. Last Accessed 15 July 2020.

[67] "HDRI Haven Database". `https://hdrihaven.com/hdris/`. Last Accessed 15 July 2020.

[68] FAIRCHILD, M. "HDR Photographic Survey Dataset". `http://rit-mcsl.org/fairchild//HDR.html`. Last Accessed 15 July 2020.

[69] "sIBL Archive". http://www.hdrlabs.com/sibl/archive.html. Last Accessed 15 July 2020.

[70] "EMPA HDR Dataset". http://www.pads.ufrj.br/~fernanda/SPIC%20Additional%20Material/. Last Accessed 15 July 2020.

[71] BANTERLE, F., ARTUSI, A., DEBATTISTA, K., et al. *Advanced High Dynamic Range Imaging (2nd Edition)*. Natick, MA, USA, AK Peters (CRC Press), July 2017. ISBN: 9781498706940.

[72] WINKLER, S. "Analysis of Public Image and Video Databases for Quality Assessment", *IEEE Journal of Selected Topics in Signal Processing*, v. 6, n. 6, pp. 616–625, 2012. doi: 10.1109/JSTSP.2012.2215007.

[73] HULUSIC, V., VALENZISE, G., PROVENZI, E., et al. "Perceived dynamic range of HDR images". In: *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, pp. 1–6, 2016. doi: 10.1109/QoMEX.2016.7498953.

[74] HIRTH, M., HOSSFELD, T., TRAN-GIA, P. "Analyzing costs and accuracy of validation mechanisms for crowdsourcing platforms", *Mathematical and Computer Modelling*, v. 57, n. 11, pp. 2918–2932, 2013. ISSN: 0895-7177. doi: https://doi.org/10.1016/j.mcm.2012.01.006. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0895717712000076>. Information System Security and Performance Modeling and Simulation for Future Mobile Networks.

[75] HOSSFELD, T., SEUFERT, M., HIRTH, M., et al. "Quantification of YouTube QoE via Crowdsourcing". In: *2011 IEEE International Symposium on Multimedia*, pp. 494–499, 2011. doi: 10.1109/ISM.2011.87.

[76] REPORT, I.-T. T. "PSTR CROWDS - Subjective evaluation of media quality using a crowdsourcing approach". https://www.itu.int/en/publications/Pages/publications.aspx?lang=en&media=electronic&parent=T-TUT-QOS-2018.

[77] P.913, I.-T. "Methods for the subjective assessment of video quality, audio quality and audiovisual quality of Internet video and distribution quality television in any environment". https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-P.913-202106-I!!PDF-E&type=items.

[78] HOSSFELD, T., KEIMEL, C., HIRTH, M., et al. "Best Practices for QoE Crowdtesting: QoE Assessment With Crowdsourcing", *IEEE Transac-*

tions on Multimedia, v. 16, n. 2, pp. 541–558, 2014. doi: 10.1109/TMM.2013.2291663.

[79] DAI, P., RZESZOTARSKI, J. M., PARITOSH, P., et al. "And Now for Something Completely Different: Improving Crowdsourcing Workflows with Micro-Diversions". In: *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, CSCW '15, p. 628–638, New York, NY, USA, 2015. Association for Computing Machinery. ISBN: 9781450329224. doi: 10.1145/2675133.2675260. Disponível em: <`https://doi.org/10.1145/2675133.2675260`>.

[80] NADERI, B., WECHSUNG, I., MÖLLER, S. "Effect of being observed on the reliability of responses in crowdsourcing micro-task platforms". In: *2015 Seventh International Workshop on Quality of Multimedia Experience (QoMEX)*, pp. 1–2, 2015. doi: 10.1109/QoMEX.2015.7148091.

[81] GADIRAJU, U., KAWASE, R., DIETZE, S., et al. "Understanding Malicious Behavior in Crowdsourcing Platforms: The Case of Online Surveys". In: *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, CHI '15, p. 1631–1640, New York, NY, USA, 2015. Association for Computing Machinery. ISBN: 9781450331456. doi: 10.1145/2702123.2702443. Disponível em: <`https://doi.org/10.1145/2702123.2702443`>.

[82] NAWALA, J., JANOWSKI, L., CMIEL, B., et al. "Describing Subjective Experiment Consistency by P-Value P–P Plot". In: *Proceedings of the 28th ACM International Conference on Multimedia*, p. 852–861, New York, NY, USA, Association for Computing Machinery, 2020. ISBN: 9781450379885. Disponível em: <`https://doi.org/10.1145/3394171.3413749`>.

[83] JANOWSKI, L., ĆMIEL, B., RUSEK, K., et al. "Generalized Score Distribution". 2019. Disponível em: <`https://arxiv.org/abs/1909.04369`>.

[84] MCDONALD, J. H. *Handbook of Biological Statistics*. 3 ed. Baltimore, Maryland, Sparky House Publishing.

[85] SZEGEDY, C., VANHOUCKE, V., IOFFE, S., et al. "Rethinking the Inception Architecture for Computer Vision". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, 2016.

[86] DENG, J., DONG, W., SOCHER, R., et al. "ImageNet: A large-scale hierarchical image database". In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

[87] BROMLEY, J., GUYON, I., LECUN, Y., et al. “Signature Verification Using a “Siamese” Time Delay Neural Network”. In: *Proceedings of the 6th International Conference on Neural Information Processing Systems*, NIPS’93, p. 737–744, San Francisco, CA, USA, 1993. Morgan Kaufmann Publishers Inc.

[88] RUDER, S. “An overview of gradient descent optimization algorithms”, *CoRR*, v. abs/1609.04747, 2016. Disponível em: <http://arxiv.org/abs/1609.04747>.

[89] KINGMA, D. P., BA, J. “Adam: A Method for Stochastic Optimization”. In: Bengio, Y., LeCun, Y. (Eds.), *3rd International Conference on Learning Representations, ICLR, Conference Track Proceedings*, San Diego, CA, USA, May 2015. Disponível em: <http://arxiv.org/abs/1412.6980>.

[90] GOODFELLOW, I., BENGIO, Y., COURVILLE, A. *Deep Learning.* Cambridge, MA, MIT Press, 2016. http://www.deeplearningbook.org.

[91] RANA, A., SINGH, P., VALENZISE, G., et al. “Deep Tone Mapping Operator for High Dynamic Range Images”, *IEEE Transactions on Image Processing*, v. 29, pp. 1285–1298, 2020.

[92] MA, K., YEGANEH, H., ZENG, K., et al. “High dynamic range image tone mapping by optimizing tone mapped image quality index”. In: *2014 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, 2014.

[93] VISAVAKITCHAROEN, A., KINOSHITA, Y., KOBAYASHI, H., et al. “Quality improvement of tone mapped images by TMQI-II based optimization for the JPEG XT standard”. In: *2016 International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS)*, pp. 1–5, 2016.

[94] ZIAEI NAFCHI, H., SHAHKOLAEI, A., FARRAHI MOGHADDAM, R., et al. “FSITM: A Feature Similarity Index For Tone-Mapped Images”, *IEEE Signal Processing Letters*, v. 22, n. 8, pp. 1026–1029, 2015.

[95] IOFFE, S., SZEGEDY, C. “Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift”. In: *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML’15, p. 448–456. JMLR.org, 2015.

[96] XU, B., WANG, N., CHEN, T., et al. “Empirical Evaluation of Rectified Activations in Convolutional Network”, *CoRR*, v. abs/1505.00853, 2015. Disponível em: <http://arxiv.org/abs/1505.00853>.

[97] YU, F., ZHANG, Y., SONG, S., et al. "LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop", *arXiv preprint arXiv:1506.03365*, 2015.

Appendix A

Scatter Plots of Individual IQA Metrics

In this appendix, we show the scatter plots between the subjective MOS and predicted MOS from each individual IQA metric in the ESPL-LIVE and TMID datasets. We also show the scatter plots between subjective MOS and predicted MOS from the fine-tuned RankIQA models, and from the different CNN architectures, in both datasets. The black line shown in each plot corresponds to the non-linear function fit to data in each case. Table A.1 identifies the metric according to their position in the 7 × 5 scatter plot grid that is shown in Figures A.1 and A.2.

Table A.1: Identification of each metric in the scatter plot grid from Figures A.1 and A.2.

	Column 1	**Column 2**	**Column 3**	**Column 4**	**Column 5**
Row 1	BIQI	BLIINDS-2	BPADA	BRISQUE	BTMQI
Row 2	C-DIIVINE	CORNIA	DESIQUE	DIIVINE	ENIQA
Row 3	FRIQUEE	GM-LOG	HIGRADE-1	HIGRADE-2	ILNIQE
Row 4	CNN-INCEPTION	CNN-RESNET50	CNN-VGG16	NFERM	NIMA
Row 5	NIQE	NIQMC	NJQA	PAM	RANKIQA
Row 6	RANKIQA2	RANKIQA3	RANKIQA4	RANKIQA5	RANKIQA6
Row 7	SISBLIM-SFB	SISBLIM-SM	SISBLIM-WFB	SISBLIM-WM	SSEQ

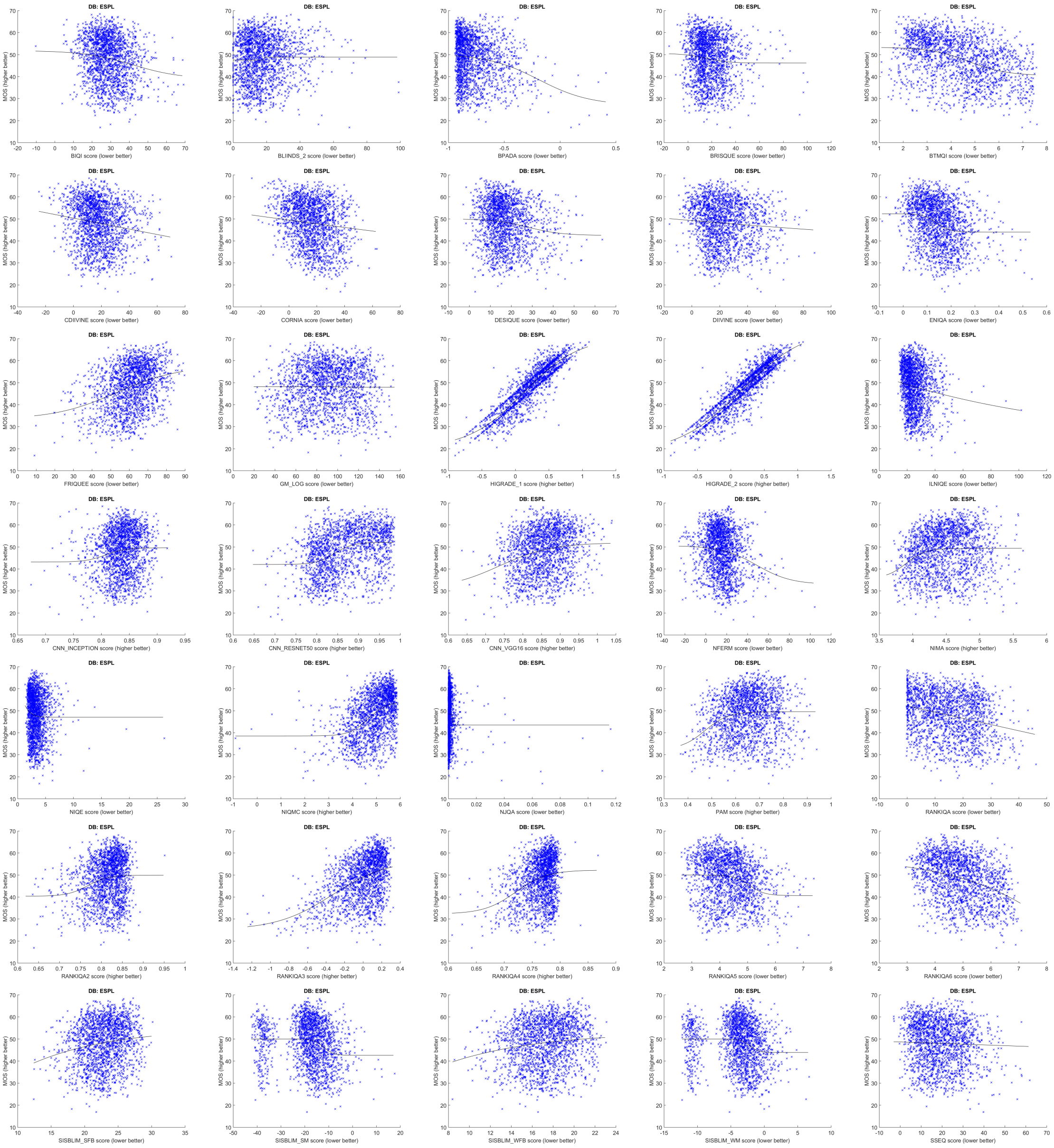


Figure A.1: Scatter plots showing subjective MOS versus predicted MOS from each IQA metric in the ESPL-LIVE dataset. The black line corresponds to the non-linear function fit to data.

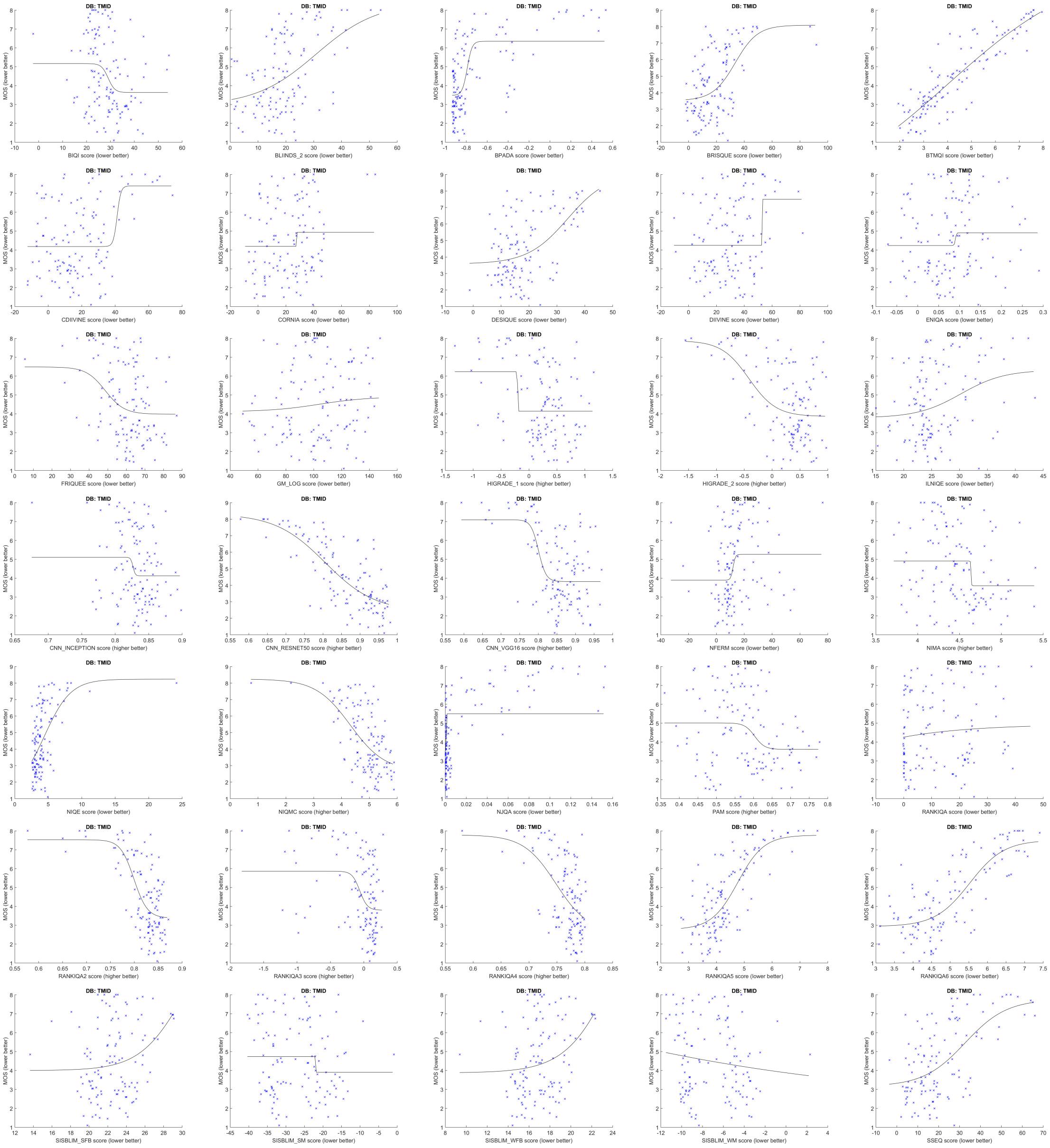


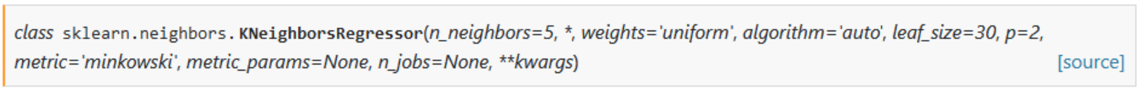
Figure A.2: Scatter plots showing subjective MOS versus predicted MOS from each IQA metric in the TMID dataset. The black line corresponds to the non-linear function fit to data.

Appendix B

Model Hyperparameters in Scikit-Learn Package

In this appendix, we list all hyperparameters that are defined in regression model implementations from the Python *scikit-learn* package [59]. Default values shown for each hyperparameter are the ones set in version 0.24.2 of this package.

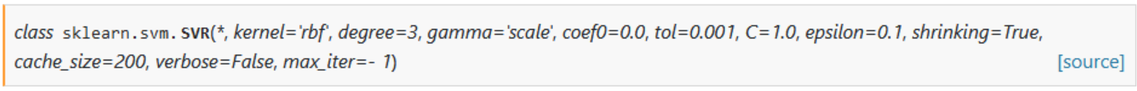
- K-Nearest Neighbors (KNN)



```
class sklearn.neighbors.KNeighborsRegressor(n_neighbors=5, *, weights='uniform', algorithm='auto', leaf_size=30, p=2,
metric='minkowski', metric_params=None, n_jobs=None, **kwargs)                                                [source]
```

Figure B.1: Hyperparameters and corresponding default values defined in version 0.24.2 of *scikit-learn* Python package for the KNN regression model. Picture taken from the package official website: https://scikit-learn.org/0.24/. Last accessed: August 17th, 2022.

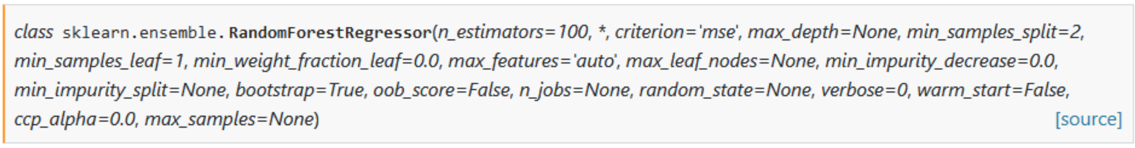
- Support Vector Regression Machine (SVR)



```
class sklearn.svm.SVR(*, kernel='rbf', degree=3, gamma='scale', coef0=0.0, tol=0.001, C=1.0, epsilon=0.1, shrinking=True,
cache_size=200, verbose=False, max_iter=- 1)                                                                    [source]
```

Figure B.2: Hyperparameters and corresponding default values defined in version 0.24.2 of *scikit-learn* Python package for the SVR regression model. Picture taken from the package official website: https://scikit-learn.org/0.24/. Last accessed: August 17th, 2022.

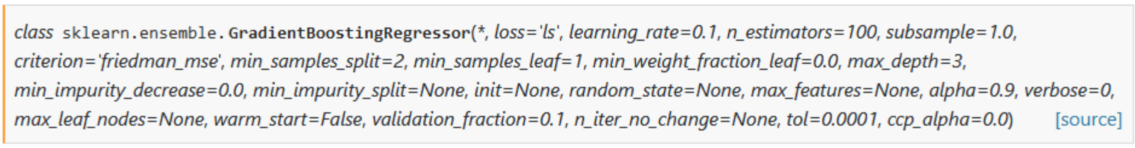
- Random Forest



```
class sklearn.ensemble.RandomForestRegressor(n_estimators=100, *, criterion='mse', max_depth=None, min_samples_split=2,
min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_features='auto', max_leaf_nodes=None, min_impurity_decrease=0.0,
min_impurity_split=None, bootstrap=True, oob_score=False, n_jobs=None, random_state=None, verbose=0, warm_start=False,
ccp_alpha=0.0, max_samples=None)                                                                              [source]
```

Figure B.3: Hyperparameters and corresponding default values defined in version 0.24.2 of *scikit-learn* Python package for the Random Forest regression model. Picture taken from the package official website: https://scikit-learn.org/0.24/. Last accessed: August 17th, 2022.

- Gradient Tree Boosting (GTB)



```
class sklearn.ensemble.GradientBoostingRegressor(*, loss='ls', learning_rate=0.1, n_estimators=100, subsample=1.0,
criterion='friedman_mse', min_samples_split=2, min_samples_leaf=1, min_weight_fraction_leaf=0.0, max_depth=3,
min_impurity_decrease=0.0, min_impurity_split=None, init=None, random_state=None, max_features=None, alpha=0.9, verbose=0,
max_leaf_nodes=None, warm_start=False, validation_fraction=0.1, n_iter_no_change=None, tol=0.0001, ccp_alpha=0.0)      [source]
```

Figure B.4: Hyperparameters and corresponding default values defined in version 0.24.2 of *scikit-learn* Python package for the GTB regression model. Picture taken from the package official website: https://scikit-learn.org/0.24/. Last accessed: August 17th, 2022.