University of Brasília
Faculty of Technology
Department of Electrical Engineering

Deep Learning Based Objective Quality Assessment of Multidimensional Visual Content

Sana Alamgeer

A thesis submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy

Supervisor
Prof. Dr. Mylene Christine Queiroz de Farias

Brasília
2022

University of Brasília
Faculty of Technology
Department of Electrical Engineering

Deep Learning Based Objective Quality Assessment of Multidimensional Visual Content

Sana Alamgeer

A thesis submitted in fulfilment of the requirements for the degree of
Doctor of Philosophy

Prof. Dr. Mylene Christine Queiroz de Farias (Supervisor)
FT/UnB

Prof. Dr. Li Weigang
CIC/UnB

Prof. Dr. João Luiz Carvalho
ENE/UnB

Prof. Dr. Carla Pagliari
Instituto Militar de Engenharia

Brasília, July 01, 2022

FICHA CATALOGRÁFICA

SANA ALAMGEER
Deep Learning Based Objective Quality Assessment of Multidimensional Visual Content [Distrito Federal] 2022.
xvi, 56 p., 210 x 297 mm (ENE/FT/UnB, Doctorate, Electrical Engineering, 2022).
Dissertation - University of Brasília, Faculty of Technology.
Department of Electrical Engineering

1. Visual Quality Assessment	2. Visual Attention
3. Deep Learning	4. 4D Light Fields
I. ENE/FT/UnB	II. Title (series)

BIBLIOGRAPHIC REFERENCE

A. SANA (2022). *Deep Learning Based Objective Quality Assessment of Multidimensional Visual Content*. Dissertation, Department of Electrical Engineering, University of Brasília, Brasília, DF, 56 p.

ASSIGNMENT OF RIGHTS

AUTOR: Sana Alamgeer
TÍTULO: Deep Learning Based Objective Quality Assessment of Multidimensional Visual Content.
GRAU: Doctorate in Electrical Engineering ANO: 2022

Sana Alamgeer
Department of Electrical Engineering - ENE - FT
University of Brasília (UnB)
Campus Darcy Ribeiro
CEP: 70919-970 - Brasília - DF - Brasil

Acknowledgements

I am truly grateful to my supervisor Prof. Dr. Mylene C. Q. Farias for her consistent guidance, support, cooperation, and inspiration during my entire journey of Doctor of Philosophy. My sincere appreciation and gratitude also goes to the team of Group of Digital Signal Processing (GPDS) laboratory, specially Dr. Alessandro Silva, and Dr. Rafael Diniz, for their prompt help in maintaining the GPU server for my research projects.

I am obliged to my husband, Irshad, and specially my parents, Tahira and Alamgeer, who have always believed in me, and have never expected nothing less from me.

I am also thankful to all my colleagues in the Laboratory of the Group of Digital Signal Processing (GPDS), specifically Dário Morais, Vinícius Oliveira, Gustavo Sandri, Henrique Garcia, Kerlla Luz, Thayane Viana, Max Vizcarra, Pedro Freitas and Priscila Andrade for being so kind and cooperative, and for making me feel at home.

Last but not least, I hereby express my sincere gratitude to the University of Brasília, for giving me the opportunity to do my doctorate while exploring the Brazilian culture. I am also thankful to the Fundacção de Apoio a Pesquisa do Distrito Federal, and Coordenacção de Aperfeiçoamento de Pessoal de Nível Superior for their financial support, without which, my research would not have been possible.

To myself, without whose determination and tireless efforts,
this work would not have been possible.

Abstract

In the last decade, there has been a tremendous increase in the popularity of multimedia applications, hence increasing multimedia content. When these contents are generated, transmitted, reconstructed and shared, their original pixel values are transformed. In this scenario, it becomes more crucial and demanding to assess visual quality of the affected visual content so that the requirements of end-users are satisfied. In this work, we investigate effective spatial, temporal, and angular features by developing no-reference algorithms that assess the visual quality of distorted multi-dimensional visual content. We use machine learning and deep learning algorithms to obtain prediction accuracy.

For two-dimensional (2D) image quality assessment, we use multiscale local binary patterns and saliency information, and train / test these features using Random Forest Regressor. For 2D video quality assessment, we introduce a novel concept of spatial and temporal saliency and custom objective quality scores. We use a Convolutional Neural Network (CNN) based light-weight model for training and testing on selected patches of video frames.

For objective quality assessment of four-dimensional (4D) light field images (LFI), we propose seven LFI quality assessment (LF-IQA) methods in total. Considering that LFI is composed of dense multi-views, Inspired by Human Visual System (HVS), we propose our first LF-IQA method that is based on a two-streams CNN architecture. The second and third LF-IQA methods are also based on a two-stream architecture, which incorporates CNN, Long Short-Term Memory (LSTM), and diverse bottleneck features. The fourth LF-IQA is based on CNN and Atrous Convolution layers (ACL), while the fifth method uses CNN, ACL, and LSTM layers. The sixth LF-IQA method is also based on a two-stream architecture, in which, horizontal and vertical EPIs are processed in the frequency domain. Last, but not least, the seventh LF-IQA method is based on a Graph Convolutional Neural Network. For all of the methods mentioned above, we performed intensive experiments, and the results show that these methods outperformed state-of-the-art methods on popular quality datasets.

Keywords: Visual Quality Assessment, 4D Light Fields, Visual Attention, Deep Learning, Bottleneck Features

Contents

List of Figures

List of Tables

Acronyms

2D 2-dimensional.

4D 4-dimensional.

ACL-LSTM Atrous Convolutional Layers with Long Short-Term Memory layers.

CNN Convolutional Neural Network.

CNN-ACL CNN with Atrous Convolutional layers.

DL Deep learning.

DNN Deep Neural Network.

DNNF-LFIQA Deep Learning-Based light field image quality assessment using Frequency domain inputs.

EPI Epipolar-plane image.

FR Full-reference.

GCNN Graph Convolutional Neural Network.

GCNN-LFIQA Graph Convolutional Neural Network-based Light Field image quality assessment.

GPU graphics processing unit.

HVS Human Visual System.

HVS-CNN Human Visual System-based multi-stream Convolutional Neural Network.

LF Light Field.

LF-IQA Light Field image quality assessment.

LFI Light Field image.

LR Linear Regression.

LSTM Long-Short-Term Memory Network.

LSTM-DNN Long-Short-Term Memory-based two-stream Deep Neural Network.

LSTM-DP Long-Short-Term Memory-based two-stream Neural Network with diverse parameters.

ML Machine learning.

MLI Micro-lens image.

MOS mean opinion scores.

MSE Mean Square Error.

NR No-reference.

NSS Natural Scene Statistics.

OQA Objective quality assessment.

PBM Pixel-based method.

PLCC Pearson's linear correlation coefficient.

PSNR Peak Signal-to-Noise Ratio.

QM Quality metrics.

RFR Random Forest regression.

RR Reduced-reference.

SAI Sub-aperture image.

SGD Stochastic Gradient Descent.

SQA Subjective quality assessment.

SROCC Spearman's rank-order correlation coefficient.

SVR Support Vector regression.

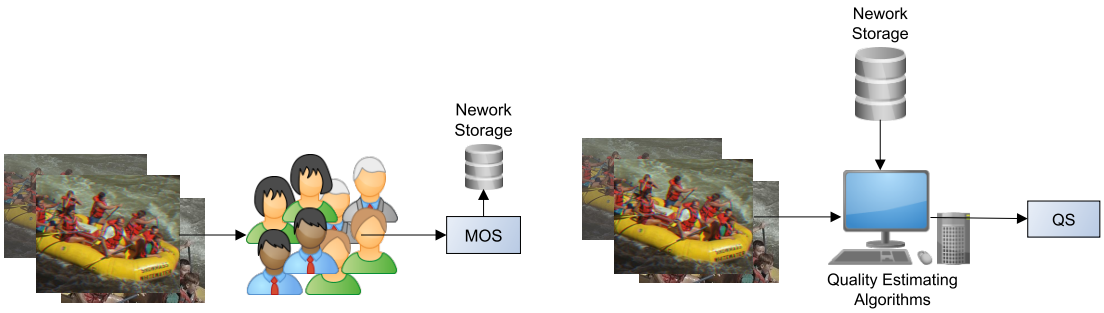
TL Transfer learning.

Chapter 1

Introduction

1.1 Overview

In the last decades, there has been a tremendous increase in the popularity of multimedia applications, especially smartphones, tablets, and personal computers. Image and Video services provided by these applications have become an integral part of consumers' life. The excess usage of multimedia applications has caused a huge growth in Internet traffic. According to a current report by Cisco^{TM} [8], every second, global Internet traffic increases by one hundred thousand GB, of which 82% are made up of videos. When visual content[1] (produced by the multimedia applications mentioned above) is compressed, transmitted, decoded, and displayed, its pixel values are transformed. In this scenario, multimedia applications require quantifying to what extent the content is affected by these operations. For this purpose, applications assess the visual quality of affected (distorted) visual content to ensure that the delivered content meets the requirements of end users.



(a) Block diagram of a typical subjective quality assessment (SQA method).

(b) Block diagram of a typical objective quality assessment (OQA) method.

Figure 1.1.1: Illustration of different ways of assessing visual quality

[1] Multidimensional images and videos.

Generally, there are two types of methods that can be used to assess visual quality. The first type of method is known as the Subjective quality assessment (SQA) method. SQA methods estimate the quality of visual contents by psychophysical experiments, which are performed in a controlled-laboratory environment. In this experiment, a number of human observers (subjects), who are usually naive in terms of visual quality analysis, analyzes the visual quality of displayed content. The experiment follows a standardized Recommendation ITU-R BT.500 [9]. In most types of experiments, the subjects are asked to rate the quality or another attribute of the displayed content (e.g., colorfulness, sharpness, noise, etc.) by giving a score. An estimate of the quality can be given by the mean opinion scores (MOS), which is computed by averaging the scores given by all subjects to a test visual content. Recommendation ITU-T P.800.1 [10] describes experimental methodologies that are used to estimate the quality of visual content. Figure 1.1(a) shows a block diagram of a typical SQA method. Although SQA methods are considered ground truth in visual quality assessment, these methods are expensive, time consuming, and usually not easy to repeat.

The second type of methods for visual quality assessment are called Objective quality assessment (OQA) methods. OQA methods use computational models, known as Quality metrics (QM), to estimate the quality of a visual content. OQA methods are faster, cheaper, and can be more easily incorporated into multimedia applications. In other words, given the limitations of SQA methods, OQA methods are often preferred. For this reason, great efforts have been done to develop fast and high accuracy OQA methods. Figure 1.1(b) shows a block diagram of a typical OQA method.

OQA methods are either *dedicated* quality metrics, which assess a specific type of distortion, or *generic quality* metrics (also known as general-purpose metrics) that estimate the overall perceived quality. Depending on the amount of reference information used, both dedicated and general-purpose OQA methods are further divided into three types, i.e., Full-reference (FR), Reduced-reference (RR) and No-reference (NR). FR-OQA methods work with the assumption that reference content is fully available and require information from both pristine and test visual content. RR-OQA methods require information from the test visual content, and a set of features from references. NR-OQA methods require information only from the test content. NR-OQA methods have no prior knowledge of references, which is why they are often called Blind Quality Metrics (BQM). Figure 1.1.2 shows a block diagram representing the information extraction steps of FR, RR, and FR OQA methods. All types of OQA methods mentioned above rely on MOS values i.e., the effectiveness of an OQA method is quantified by to what extent its quality prediction is in agreement with human judgements.

Since the Human Visual System (HVS) is the ultimate estimator of visual quality, researchers have integrated the characteristics of the HVS into the design of OQA methods. Some OQA methods [11–25] take into account the lower-level aspects of HVS, such as con-

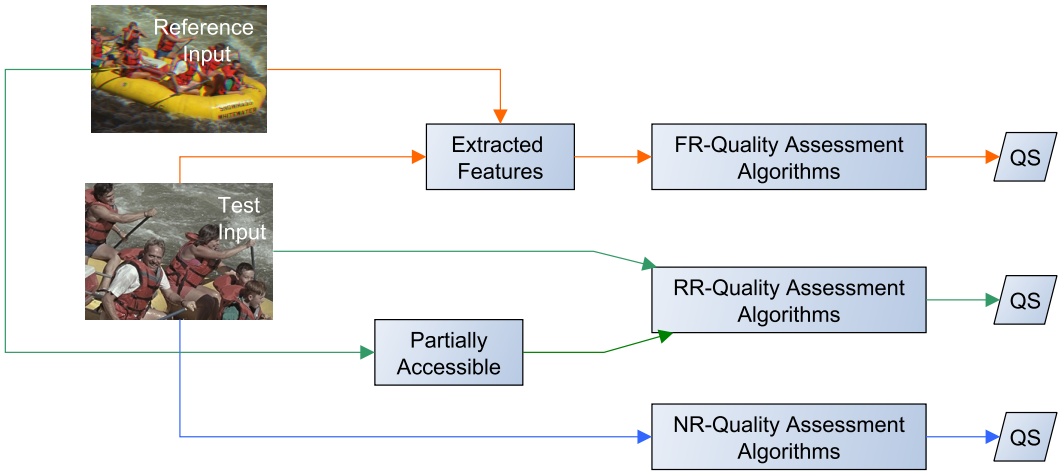


Figure 1.1.2: Categories of objective quality assessment (OQA) methods.

trast sensitivity, luminance masking, and texture masking. These HVS-based OQA methods are allegedly more reliable than purely pixel-based FR-OQA methods, such as Peak Signal-to-Noise Ratio (PSNR) and Mean Square Error (MSE). Other FR-OQA methods [26–29] incorporate HVS characteristics by using feature extraction approaches. Webster *et al.* [30] proposed one of the first RR-OQA methods. This method uses spatial and temporal features to assess the quality of videos. For some multimedia applications, it is difficult to acquire information from reference visual contents. In this scenario, the NR-OQA methods are the only available option. Although these methods are generally less accurate than the FR methods, they are less complex. Some of NR-OQA methods [31–34] use a distortion-specific approach. Despite of the fact that, NR-OQA methods have gained a lot of attention, their design is still a challenge [35, 36].

To further improve the reliability of OQA methods, the current research trend consists in investigating the impact of integrating visual attention into their designs [37, 38]. Researchers incorporate visual attention aspects into OQA methods to optimize the ability to predict quality [39, 40]. This approach assumes that, if a distortion occurs in an area that attracts the viewer's attention, it is more annoying than if it occurs in any other area. The algorithm weighs local distortions with local saliency.

The advancement of imaging technologies in the last decade has allowed for more faithful representations of tridimensional (3D) scenes to create immersive experiences that are indistinguishable from the real world. These technological advancements have produced plenoptic devices that can capture and display visual information to describe objects in 3D space from any point-of-view. Depending on the capturing device, this visual information can correspond to hologram, Light Field (LF), or point cloud imaging formats. In the particular case of LF contents, the 4-dimensional (4D) Light Field image (LFI) describe the dis-

tribution of light rays in a free space[2], including their spatial and angular dimensions. The high-dimensionality of the LFI data represents challenges to compression, transmission, and reconstruction algorithms, which are often the source of degradations that alter the quality of LFIs. In this scenario, accurate Light Field image quality assessment (LF-IQA) methods are important tools, that play a vital role in the design of these algorithms. The 2D OQA methods mainly focus on spatial information, but for an LF content, an OQA method needs to focus on both the spatial and angular information. Therefore, it is highly demanded to design an OQA method to automatically and accurately measure the quality of LFIs.

1.2 Problem Statement

To develop a visual quality assessment method, three major steps are necessary; measuring, pooling, and mapping, as named by Hemani and Reibman [41]. The measuring step consists of determining a set of features that are relevant to visual quality. For example, edge sharpness [42], Prewitt filters [43], Natural Scene Statistics (NSS) [44], statistics on the curvelet domain [45], discrete cosine transform domain [46] or gradient domain [47], spatial and spectral entropies [48], subband statistics in the wavelet-packet domain [49] are measurements used to extract features. Each set of measurements generate one feature vector. The second step is to develop a pooling strategy to assess the quality of a content that varies over time. For example, Minkowski summation [33] and average pooling [50, 51] strategies are often used in this step. Pooling strategies are also used for dimensionality reduction [52]. It is worth mentioning that the chosen pooling strategy should consider how the HVS perceives temporal signals.

The third step in developing an OQA method consists of creating a model that maps the pooled data into quality estimates. The mapping model can be a predefined function, as adopted in the Structural Similarity (SSIM) [11] and Gradient Magnitude Similarity Deviation (GMSD) [53] algorithms, or automatically learned from the pooled features, as adopted in most Machine learning (ML) or Deep learning (DL) based methods. In these methods, the most commonly used ML methods are Support Vector regression (SVR) machines, Linear Regression (LR), Random Forest regression (RFR) machines, and one Deep learning (DL)-based method which is Convolutional Neural Network (CNN). The block diagram in Figure 1.2.1 (a) shows the way in which the test input is processed by the ML methods in previous studies. In this figure, hand-crafted features are extracted from test input. After pooling these features, ML algorithms are used to map and predict quality scores. Figure 1.2.1 (b) represents another way to use the ML-based approach that consists of extraction of learned features using

[2] In electrical engineering, free space means the air.

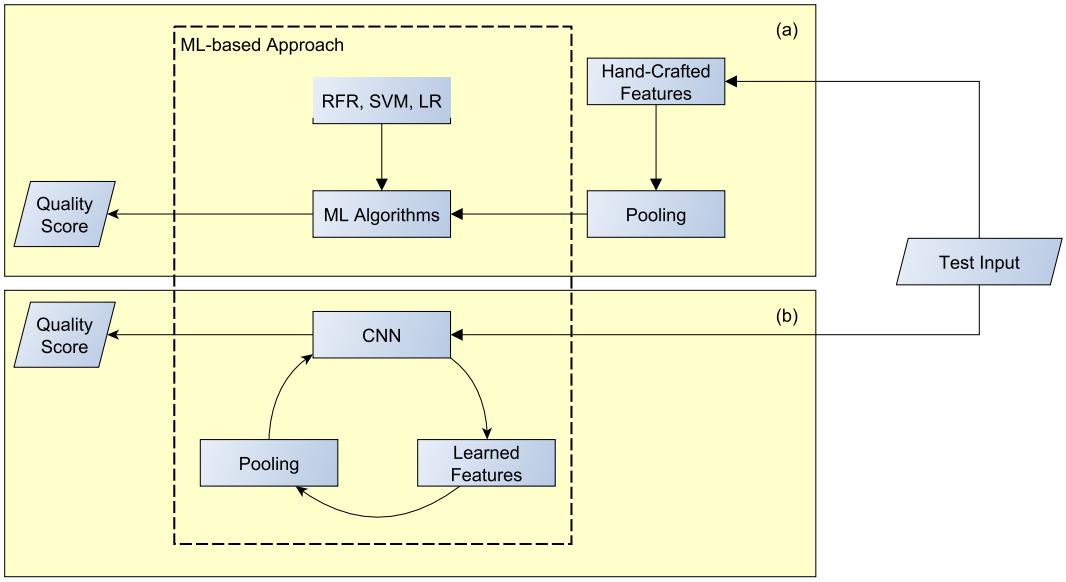


Figure 1.2.1: Ways of processing input by machine learning methods in previous studies

a neural network. After pooling, these features are fed to CNN for quality prediction. It can be noticed that in this type of approach, a neural network is used as a third-party algorithm.

To assess the quality of videos, the most popular way is to use the average of frame level quality measures, and pool them to the overall video quality, represented in Figure 1.2.2. Then, these pooled features are mapped to MOS by ML-based methods [54–56]. One other possibility is to use the OQA methods to measure spatial quality and incorporate a temporal factor, e.g., by using similarity between the motion vectors [57–61] or the quality variation along the time axis [62, 63]. After pooling, these spatio-temporal factors are mapped by the ML-based methods [55].

Most Pixel-based method (PBM) are complex because they analyze the visual content by extracting features directly from pixels. PBM methods can use these features to predict the presence, and strength of common distortions or to analyze the impact of distortions on NSS. In this case, the quality values depend on the type of single distortion or a combination of distortions. The distortion-specific methods may be unable to assess the overall quality in the presence of other types of distortions. Given these limitations, researchers have been working on PBM methods that do not make assumptions about specific distortions [64–67]. These methods are unbiased, and generally perform an analysis of the statistical characteristics found in original visual content.

Great efforts have been made for the development of objective and subjective quality assessment of 4D LFI, and their quality assessment datasets. For measuring stage, often a set of spatial features are extracted using GMSD, Morphological Wavelet Peak Signal-to-Noise Ratio (MW-PSNR) [68], and image contours (NICE) [69]. The angular features are ex-

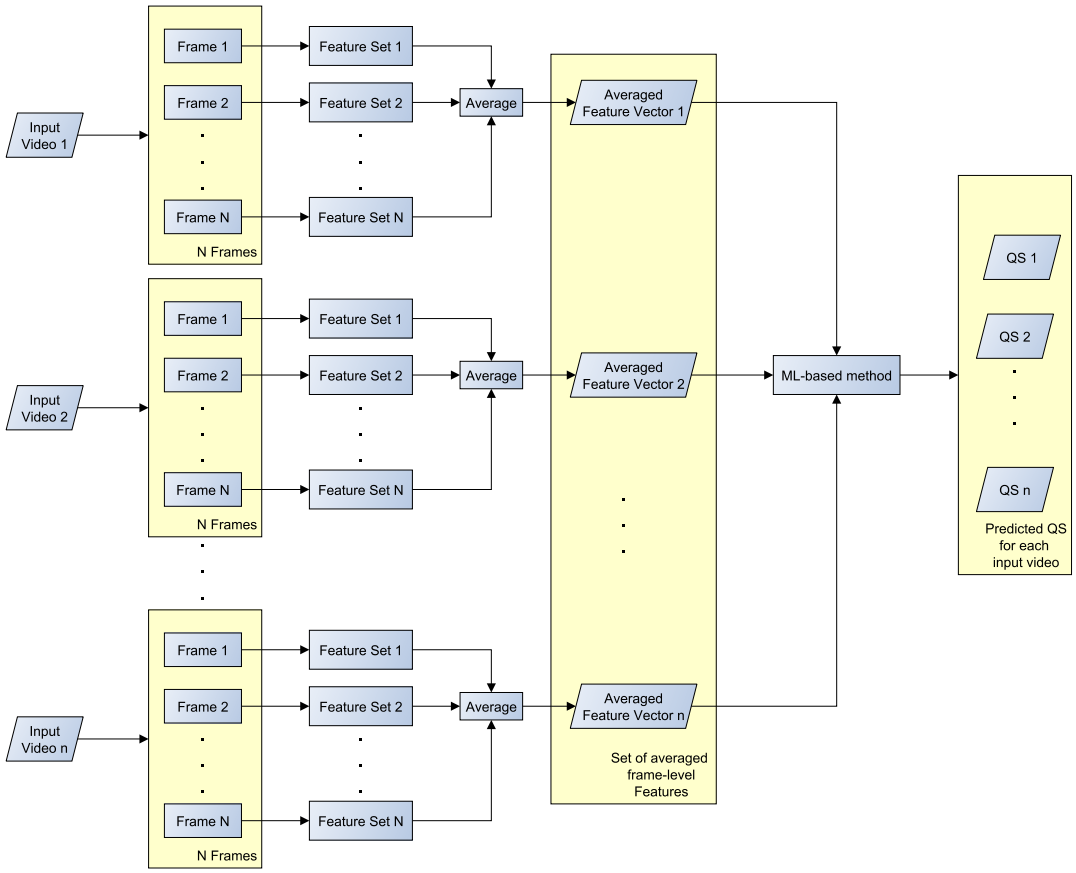


Figure 1.2.2: Most popular ML-based Objective quality assessment (OQA) method for videos using averaged frame-level features.

tracted from Epipolar-plane image (EPI) of LFI [70]. For pooling and mapping the extracted features, either FR-OQA or NR-OQA metrics of 2D visual content (as mentioned above) are used [71–76]. Some of the LF-IQA methods [77–80] rely on hand-crafted features extracted from different formats of LF image, and these features are fed to SVR algorithm for quality prediction.

Deep learning based methods, such as CNNs, have been proven highly efficient in computer vision, and image processing domains. Fujita *et al.* [81] have investigated in what format 4D LFI should be processed by CNNs for the signal restoration problem. Through their experiments, they found that the EPI domain benefits the learning accuracy of CNN, which arises an intuition towards the inspection of EPIs, and derive features from it for quality assessment of LFIs. From several CNN-based LF-IQA methods [82–84], only a few methods [85, 86] have incorporated horizontal EPIs as input for a single-stream neural network. Considering the capabilities of an EPI, its bidirectional information (horizontal and vertical) might be useful for identifying several types of spatial and angular domain distortions.

Despite of the advancement in objective quality assessment methods of distorted LF images, there are some limitations. Majority of LF-IQA methods use traditional 2D image quality assessment techniques or rely on low-level spatial features. The architectures of CNN-based LF-IQA methods are shallow, i.e. not deep enough to take advantage of the angular features in both horizontal and vertical EPIs. Also, up to our knowledge, there is a limited work available [85, 86] which employs the features from only the horizontal EPIs. Moreover, the traditional CNN-based approaches are unable to exploit the self-similar patterns. For example, if we randomly shuffle the pixels of an image, then the traditional CNN-based LF-IQAapproach fails to recognize it. Additionally, there is lack of research on image quality assessment methods that work in frequency domain of the LF images. The main advantage of using frequency domain is that, it gives complete control over intense characteristics of the image that are difficult to be seen in spatio-angular domain.

Some of LF-IQA datasets, such as LFDD [4], have complex content in terms of depth and focus with cluttered background and foreground having the same range of intensities. LF images in such datasets are difficult to analyze for quality assessment either subjectively or objectively. Observers in subjective experiment cannot easily differentiate between the good quality and bad quality of complex LF images because it is difficult to visualize distortions. Previous study [87] shows that such datasets become more challenging for quality assessment methods to provide better quality of experience. To the best of our knowledge, there is a lack of quality assessment methods for such challenging LF-IQA datasets.

The existing LF-IQA methods do not ascertain long-term dependencies, and relationships among distortion-related characteristics of distorted LFIs. For example, spatial and angular information strongly depend on each other for the perceived quality of an LF image. Traditional neural networks are limited by their localized receptive fields, and cannot persist important information (long-term) to learn dependencies between the data elements along with their relationships, such as points appearing too close or far from the center in a corresponding field of view, or invalid pixels at the boundaries due to misalignment of sub-views.

In the Human Visual System (HVS), the human visual cortex responds in a different way to process multiview stimuli. Specifically, the HVS has two parallel hierarchical sequences, or processing streams, which are known as dorsal and ventral streams. The dorsal stream begins in the low-level visual area of the visual cortex, known as the primary visual cortex (V1), interconnecting the ventral stream that runs into the areas of the high-level visual cortex of V2 to V5 [88, 89]. While processing a stimuli gathered from multiviews, binocular fusion and disparity responses are formed and weighted in V1, and then passed on to V2 for further processing by the dorsal and ventral streams. These two streams filter the processed information, and send it to V5. The existing LF-IQA methods do not ascertain this multistream-based processing of multiview stimuli.

1.3 Proposed Approach

This research proposes a set of dedicated quality assessment methods for 2D and 4D visual contents based on deep learning methods. Our aim is overcome the challenges mentioned in previous section, and achieve the accurate and robust outcomes when running the proposed quality assessment methods as real-world applications without reference information.

First, we inspect texture and saliency-based measurements for the NR quality assessment of 2D images. We extract the textures from distorted images and weight these features with saliency information. Then, we pass these weighted features on to a machine learning algorithm that performs a regression operation for quality prediction. Although machine learning methods are capable of learning trends and patterns, they are highly prone to error, i.e., the data we push in the models for training must be clean and accurate. Therefore, to reduce the risk of error, we employ a CNN-based approach to assess the quality of 2D videos. The method assesses the quality of distorted videos in a frame-by-frame manner. For training, we pass to the network a selected number of patches generated from spatio-temporal domains of a video. To select patches, we exploit saliency information so that we can obtain the patches with high perceptual relevance.

Second, inspired by the multi-dimension (spatial and angular) information of 4D LF images, it is worth investigating the features of EPIs for performing the quality assessment. Therefore, in this work, we propose deep learning-based methods that take horizontal and vertical EPIs as inputs, and learn important features from them. We have adapted multi-layered streams CNN architectures, so that the network can learn long-term dependent, and distortion-related characteristics from EPIs, not only in spatio-angular domain, but also in the frequency domain.

Third, to overcome the limitations of traditional CNN networks, we adapt the networks with an expanded field-of-view. Specifically, without increasing the number of parameters, we increase the receptive field of a network so that it can learn more dense features from the EPIs of distorted LF images. Most importantly, to obtain an adequate amount of features and increase the performance accuracy for training, we also adapt bottleneck features generated from popular pre-trained networks. These approaches are ideal when hardware resources are limited.

Fourth, we propose a deep learning-based method that is dedicated to assess the quality of compressed LF images. After an image is compressed, it loses certain data contents of the image, that in consequence, breaks up the actual structure of the image. To efficiently identify and learn from such compressed and unstructured data, we adapt graph convolutions for LF-IQA. This method is based on a deep single-stream network architecture which takes horizontal EPI as input. In this scenario, we take full advantage of graph convolutions by

assuming that our data is unordered and irregular, and we aim to achieve good prediction performance for such data.

1.4 Contributions

Based on the knowledge provided in previous section, hence, our contributions are as follows:

- For NR quality assessment of 2D images, a machine learning based method is proposed that employs texture information weighted by spatial saliency.
- For NR quality assessment of 2D videos, a CNN-based method is proposed, which incorporates the selected number of patches from video frames. The method uses spatio-temporal saliency to select the most relevant patches. Most importantly, the method is independent of subjective quality scores to quantify the efficiency of the predicted quality.
- Seven novel NR and deep learning based methods are proposed to assess the quality of LF images. The methods are composed of diverse parameters, and explore dense features of spatial and angular dimensions in horizontal and vertical EPIs. Specifically:
  - NR LF-IQA method is proposed that is based on HVS-inspired two-streams CNN. The method uses a novel technique to generate images of multiple epipolar planes.
  - Two NR LF-IQA methods are proposed that are based on Long Short-Term Memory Network, and diverse parameters to learn long-term and distortion-related characteristics from EPIs.
  - Two NR LF-IQAmethods are proposed that are based on CNN and Dense Atrous Convolutions. The networks expand the receptive field to capture dense features from EPIs, and learn long-term dependencies among data points.
  - A NR LF-IQA method is proposed that is based on multi-stream neural network which incorporates frequency domain inputs of EPIs.
  - A NR LF-IQA method is proposed that is based on Deep Graph Convolutional Neural Network. The method identifies unstructured data points in EPIs and learns important features for good quality predictions.
- The significance of the proposed methods is proved on using publicly accessible corresponding datasets for both 2D and 4D visual contents.

1.5 Thesis Outline

This document is divided into eight chapters. Chapter 2 describes the basic concepts that have been employed in this work with a brief literature review. In Chapter 3, we present the proposed IQA and VQA methods that blindly estimate image and video quality, respectively. In Chapters 4 to 7, we present our methods for quality assessment of 4D light field images. Finally, In Chapter 8, we summarize the results presented in this work, with concluding remarks, and future directions. A high-level overview of Chapters 3 to 7 is as follows:

- Chapter 3: This chapter presents general-purpose no-reference image and video quality assessment methods. The image quality assessment method is based on the textural statistics of multiscale local binary patterns. The method incorporates texture and saliency information. Quality is predicted after training a random forest regressor algorithm. The video quality assessment method uses a single-stream CNN model, and selects the most perceptually relevant patches using spatial and temporal saliency models. The method does not require subjective quality scores to train the CNN; rather, it uses computed objective quality scores as target quality scores for the video frames.

- Chapter 4: This chapter presents no-reference LF image quality assessment methods, that are based on a two-stream CNN architectures. First method HVS-CNN is inspired by the human visual system which is able to extract rich distortion-related spatial and angular LF characteristics, and predict the LF quality. The method uses a novel approach to generate multiple epipolar plane images. Second LF-IQA method DNNF-LFIQA in this chapter is novel and based on a deep neural network. It incorporates inputs in frequency domain of angular and spatial information of LF images. The method is also composed of two processing streams employing the CNN layers with different set of parameters.

- Chapter 5: This chapter presents two novel no-reference LF image quality assessment methods. The first method is composed of Long-Short-Term Memory-based two-stream Deep Neural Network (LSTM-DNN), while the second method is composed of Long-Short-Term Memory-based two-stream Neural Network with diverse parameters (LSTM-DP). Both the LSTM-DNN and LSTM-DP methods incorporate bottleneck features generated from different pre-trained networks.

- Chapter 6: This chapter presents two novel no-reference LF image quality assessment methods. First method uses CNN with Atrous Convolutional layers (CNN-ACL), and explores dense features of spatial and angular information of LF images. The second method uses Atrous Convolutional Layers with Long Short-Term Memory layers (ACL-

LSTM). Both CNN-ACL and ACL-LSTM methods are independent of reference information, and based on two-streams architectures.

- Chapter 7: This chapter presents a novel no-reference LF image quality assessment method, which is based on the Graph Convolutional Neural Network-based Light Field image quality assessment (GCNN-LFIQA). GCNN-LFIQA not only takes into account both LF angular and spatial information, but also learns the order of pixel information from input graphs for quality prediction.

Chapter 2

Basic Concepts and Literature Review

In this chapter, we present a background to the basic concepts that have been used in the development of this work, including objective quality assessment (OQA) methods, 4D Light Field images, machine learning (ML), deep learning (DL) methods, and Visual Attention (VA). The main purpose of this chapter is to familiarize the reader with the research topics and introduce some relevant information with respect to the literature review.

2.1 Objective Quality Assessment Methods

Objective quality assessment (OQA) methods are computational algorithms that are known as Quality metrics (QM). The goal of OQA methods is to predict the perceived quality of visual content. As mentioned above, the effectiveness of OQA method is generally quantified by to what extent its quality prediction is in agreement with human judgements (known as subjective quality scores or MOS), i.e., comparing the predicted quality scores with subjective quality scores. The algorithms that are used to compare the results are called performance evaluation metrics.

In visual quality assessment, the most common and widely used performance evaluation metrics are Spearman's rank-order correlation coefficient (SROCC) and Pearson's linear correlation coefficient (PLCC). SROCC is a non-parametric algorithm that is used to measure the degree of association between two variables. PLCC measures the degree of relationship between linearly related variables. The difference between SROCC and PLCC is that SROCC describes a monotonic relationship between two variables, while PLCC describes a linear relationship. SROCC is computed as follows [90, 91]:

$$r_s = 1 - \frac{6\sum_{i=1}^{n} d_i^2}{n(n^2-1)} \tag{2.1.1}$$

where r_s denotes SROCC, d_i is the difference between the ranks of the corresponding variables and n is the number of observations. LCC is computed as follows [90, 91]:

$$r_{x,y} = \frac{\sum_{i=1}^{n}(x_i - x)(y_i - y)}{\sqrt{\left[\sum_{i=1}^{n}(x_i - \bar{x})^2\right]\left[\sum_{i=1}^{n}(y_i - \bar{y})^2\right]}} \tag{2.1.2}$$

where $r_{x,y}$ represents LCC between x and y, n is the number of observations, x_i is the value of x at the i^{th} observation and similarly y_i is the value of y at the i^{th} observation.

2.2 4D Light Field Images

A Light Field image (LFI) describes the angular distribution of light rays in free space, and it allows one to capture richer information from our world. LF model was first defined by Gershun [92] in 1936. Later, Adelson and Bergen [93] introduced a complete version of the model in 1991, and it is known as the plenoptic function. The plenoptic function is a multidimensional function that describes the set of light rays traveling in every direction through every point in the 3D space, from the geometric optics perspective. The original plenoptic function is obtained by measuring the light rays at every possible location (x, y, z), from every possible angle (θ, ϕ), at every wavelength λ, and at every time t. This way, a 7D plenoptic function can be denoted as:

$$P = L(x, y, z, \theta, \phi, \lambda, t), \tag{2.2.1}$$

To handle such high dimensional data occupied by 7D plenoptic function, it requires a large number of computational power and resources. Therefore, to reduce the computational complexity, Levoy and Hanrahan [94] and Gortler *et al.* [95] introduced a 4D plenoptic function by assuming that the light field image is monochromatic, time-invariant, and measured in free space where the radiance remains constant along a straight line. Thus, a 4D plenoptic function is used to parameterize the light rays by the coordinates of their intersections with two planes that are placed at arbitrary positions. The coordinate system is denoted by (u, v) for the first plane, and (s, t) for the second plane. As shown in Figure 2.2.1(a), an oriented light ray defined in the two-plane system first intersects the uv plane at coordinate (u, v), and then intersects the st plane at coordinate (s, t). Henceforth, a 4D plenoptic function can be denoted as:

$$P = L(u, v, s, t), \tag{2.2.2}$$

where s and t dimensions are referred to as the angular dimensions, and u and v dimensions are referred to as the spatial dimensions of a light field image.

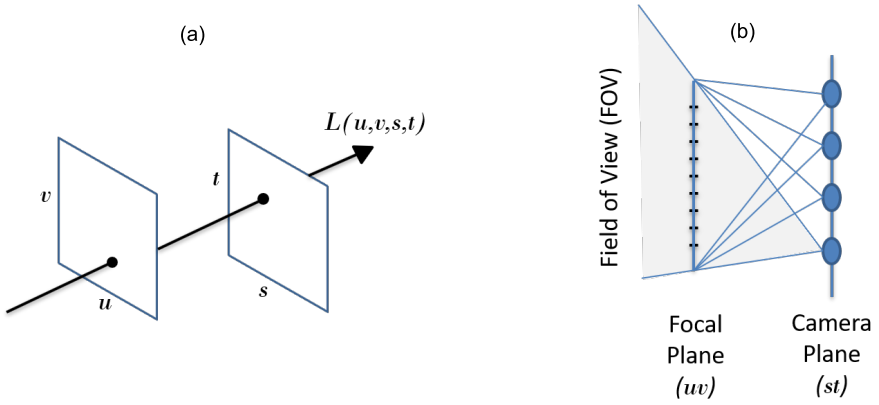


Figure 2.2.1: (a) A two-plane plenoptic to parameterize a 4D light field, and (b) Spatial multiplexed imaging system to acquire a 4D light field.

Figure 2.2.2: Spatial multiplexed imaging system to acquire 4D light field images: (a) A Lytro Illum 1.0 light field camera [1], and (b) Raytrix R29 3D plenoptic light field camera [2].

Acquisition of a light field image requires specially designed imaging systems. Sensors in conventional cameras can only measure the information from the spatial dimensions of a scene at a single moment. But to acquire a 4D light field image, we need to capture multiple samples along the angular dimensions. The most commonly used approach for light field acquisition is known as a multiplexed imaging system. This system encodes the 4D light field image into a 2D sensor plane, by multiplexing the angular domain into the spatial domain. The multiplexed imaging system is further categorized into spatial multiplexing [96]. In spatial multiplexing, an interspersed array of 2D slices of the light field image are captured by the sensor. This approach is implemented using an array of micro-lenses or lenslet array which is statically placed in front of the photosensor. Figure 2.2.1(b) shows an implementation of a spatial multiplexed imaging system where the st plane represents a set of cameras or micro-lens array, and the uv plane represents the focal plane of micro-lenses on the st plane. Figure 2.2.2(a) and (b) show examples of spatially multiplexed imaging systems (Lytro

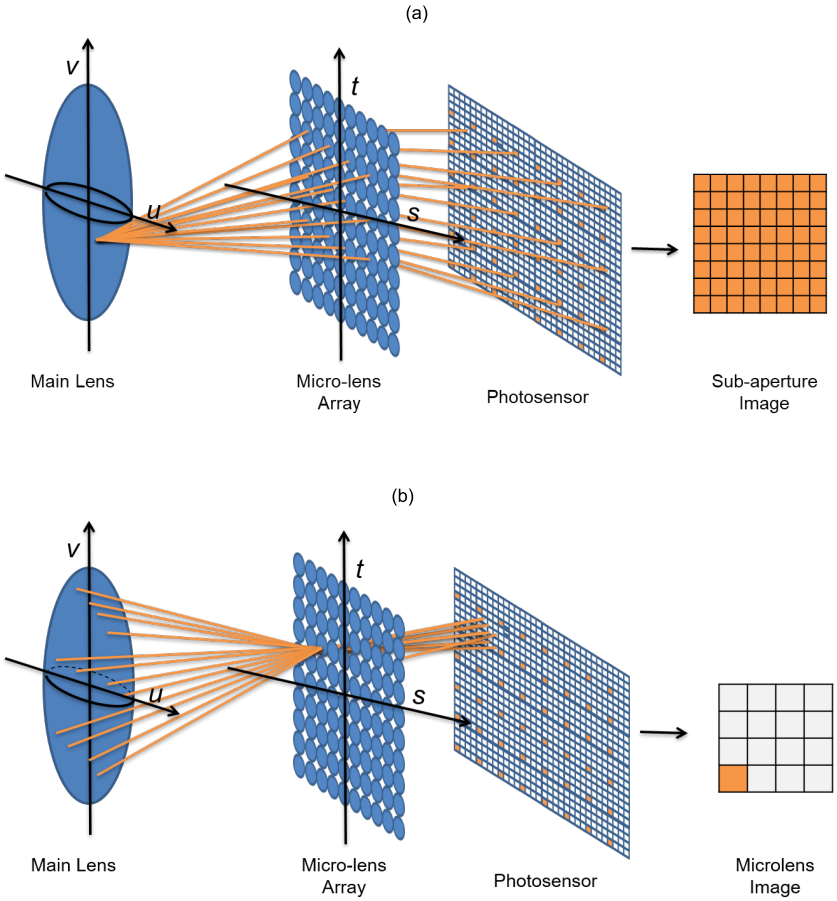


Figure 2.2.3: Different 2D representation of a 4D LFI. (a) Sub-aperture image representation with the given viewpoint (u^*, v^*), and (b) Micro-lens image representation with the given location (s^*, t^*).

Illum 1.0 light field camera [1], and Raytrix R29 3D plenoptic light field camera [2]) to capture a 4D light field and are available for commercial and consumer use. Both Lytro and Rayrix cameras are categorized as the "plenoptic camera 1.0". Each microlens in a "plenoptic camera 1.0" captures the angular distribution of the radiance. By gathering pixels in the same coordinate of each sub-view, we can obtain an image located at a certain viewpoint.

For visualization, two-plane parameterization is used to generate different 2D representations from a 4D light field image. Considering that the st plane represents an array of micro-lenses, and the uv plane represents the focal plane of micro-lenses, we can obtain 2D sub-aperture, micro-lens, and epipolar-plane images from a 4D light field image. As illustrated in Figure 2.2.3(a), a Sub-aperture image (SAI) represents the incoming rays from a given angular position u^*, v^* which are received by all micro-lens regions on the st plane. As illustrated in Figure 2.2.3(b), a Micro-lens image (MLI) represents a set of all incoming light rays from the uv plane intersected with a given micro-lens location (s^*, t^*). Hence, we can obtain SAI or MLI by gathering either two spatial dimensions (uv) or two angular dimensions

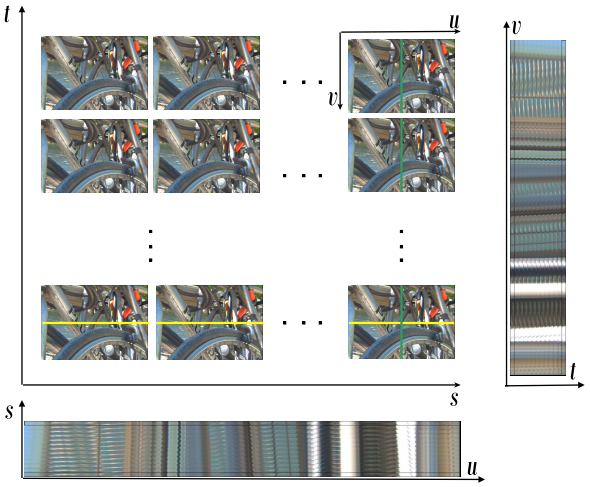


Figure 2.2.4: Illustration of epipolar-plane image (EPI) of a light field image: A 9×9 grid of 81 SAIs of the Bikes-LFI from Win5-LID dataset [3] with corresponding Vertical (extracted from green line) and Horizontal (extracted from yellow line) EPIs.

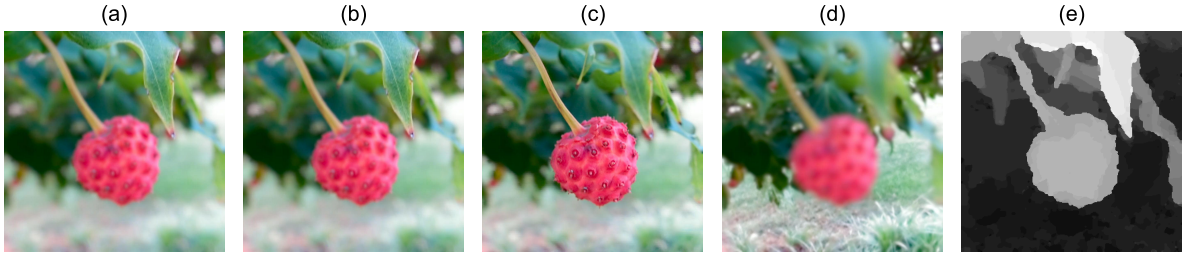


Figure 2.2.5: Representation of depth map for LFI: (a), (b), (c), and (d) four SAIs with focus at different depth levels and (e) the corresponding depth map.

(st) respectively.

By combining spatial and angular dimensions of a light field image, we can obtain another important 2D representation, which is called the Epipolar-plane image (EPI). EPIs are produced by gathering the light field samples with a fixed spatial coordinate v or u and angular coordinate t or s. Specifically, the vertical EPIs can be obtained by fixing the coordinates u and s, while the horizontal EPIs can be obtained by fixing the coordinates v and t. The slopes of lines in the EPI reflect the depth of the scene captured by the light field image. The structured information represented by the EPIs is widely exploited to infer scene geometry. Figure 2.2.4 shows a grid of SAIs of the image 'Bikes' from the Win5-LID dataset [3], along with the corresponding horizontal (obtained from green line fixing the coordinates (u,s)) and vertical EPIs (obtained from yellow line fixing the coordinates (t,v)).

The 4-dimensional LFI representation contains multiple views of the scene that are used

to estimate depth maps. The baseline between adjacent views in a light field image is narrow, which makes it difficult to recover the disparity between two views using traditional stereo matching methods. Therefore, instead of using stereo matching methods, constraints and cues which take advantage of all the views together are used to estimate the depth map from a light field image. For example, Figures 2.2.5(a), (b), (c), and (d) illustrate 4 sub-views of LFI focusing at different depth levels [97]. We observe that each sub-view shows only a certain region in focus at the corresponding depth level. By taking advantage of refocusing feature of the light field, we can estimate the depth of each ray of light recorded in the sensor via measuring pixels in focus [98]. As shown in Figure 2.2.5(e), a depth map provides additional information for the perceived depth of a scene. It can greatly help to separate objects from similar and cluttered backgrounds.

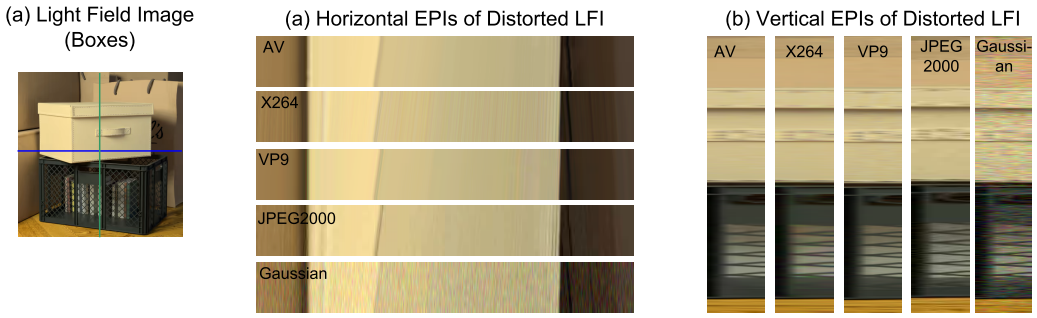
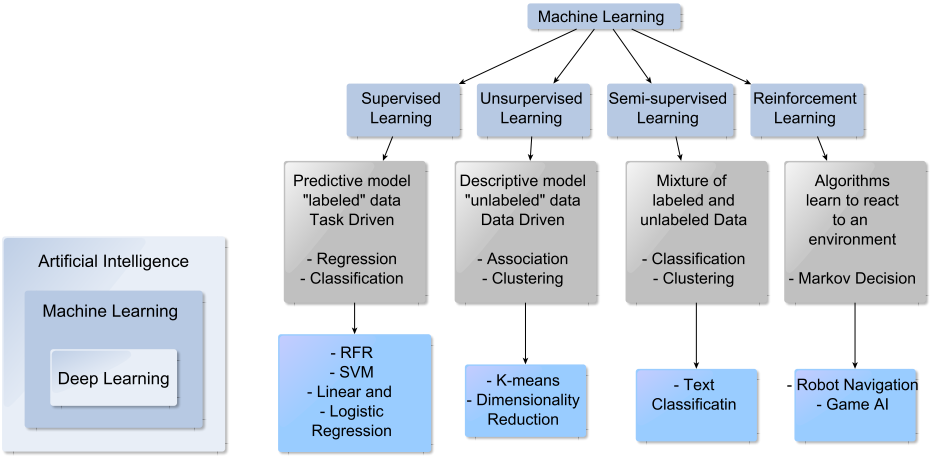


Figure 2.2.6: Example of horizontal and vertical EPIs of LFI distorted by different types of degradations: (a) LFI (Boxes) from taken from the LFDD dataset [4], (b) Horizontal EPI obtained from the blue line, and (c) Vertical EPI obtained from the green line.

From acquisition to display, LFIs go through several processing stages (e.g., acquisition, compression, transmission, rendering, and display). At every stage, distortions may be introduced that may affect the LFI visual quality [99–101]. Figure 2.2.6 shows an example of horizontal and vertical EPIs extracted from LFIs of the LFDD [4] dataset, and distorted by different types of degradations. Taking into account the texture information contained in each distorted EPI around the edge area, as displayed Figure 2.2.6, we can see a clear difference in distorted regions in every distorted EPI. Specifically, the EPIs with Gaussian distortion show more visible distortions than the AV, X264, VP9, and JPEG2000 distortions. This suggests that EPIs are sensitive to distortions [72, 80, 102]. In this scenario, accurate light field image quality assessment (LF-IQA) methods are important tools that play a vital role in the design of these algorithms exploring the dense structured features of EPIs.

In recent years, several LF-IQA methods have been developed, most of them relying on hand-crafted features extracted from MLIs [78, 101, 103–105], SAIs [73, 77, 99, 102, 106, 107], and EPIs [80, 108, 109]. These features are mapped on the corresponding subjective mean opinion scores (MOS) using machine learning (ML) based regression algorithms.

2.3 Machine Learning Methods



(a) Domain of artificial intelligence

(b) Categories of machine learning algorithms

Figure 2.3.1: Machine learning-based models for training and testing.

As shown in Figure 2.3.1(a), Machine learning (ML) is a subset knowledge domain of Artificial Intelligence (AI). ML is defined as algorithms that process data, learn from those data, and then apply what they have learned to make decisions [110]. Alpaydin [111] has described machine learning as the process of programming computers to optimize a performance criterion using example data or previous experience. We have a model defined up to some parameters, and learning is the execution of a computer program to optimize the parameters of the model using the training data. The model may be *predictive* to make predictions, or *descriptive* to gain knowledge from data, or both [111]. For example, by analyzing sample face images of a person, a learning program captures the pattern specific to a person, and then recognizes them by checking for this pattern in a given image. This is one example of pattern recognition problem [111].

Based on the types of application, ML algorithms can be divided into four categories: supervised, unsupervised, semi-supervised, and reinforcement learning. As shown in Figure 2.3.1(b), supervised learning is a predictive model that processes labeled data to achieve a specific task. For example, regression and classification are supervised learning applications. In regression, manually engineered features as input and a scalar number as outcome are provided to the supervised learning-based algorithm, which, then, finds a relationship between these input and output variables, which allows it to predict the outcome. Some of the regression methods are the Random Forest Regressor (RFR), the Support Vector Machine (SVM), the Linear Regression (LrR), and the Logistic Regression (LgR).

Unsupervised learning is a descriptive model that processes unlabeled data. The model learns from hidden structure of data. One of the unsupervised learning methods is to learn associations of different attributes of data. Semi-supervised learning is the combination of supervised and unsupervised learning methods. For example, learning from unstructured data to define tags and types of content in text classification problem is one of the semi-supervised methods. In reinforcement learning, the model assesses the policies (rules) and learns from past good action sequences to be able to generate a policy. For example, in some applications, the output of the system is a sequence of actions. In such a case, a single action is not important; the policy is important, which is the sequence of correct actions to reach the goal. An action is good if it is part of a good policy. Robotic cars are one of the reinforcement learning methods. In this work, we have used supervised machine learning approach to perform a regression. In this approach, we train a regression algorithm using input images and the subjective quality scores as targets or labels.

In OQA methods for visual content, the most commonly used ML methods are Support Vector Regressor Machines (SVR), Linear Regression (LrR), and Random Forest Regressors (RFR) that are based on a supervised learning approach of machine learning. SVR is a kernel-based regression method that uses variants of kernel functions for learning. For example, CORNIA [64], CQA [45], SSEQ [48], BRISQUE [112], LTP [113], DIIVINE [114], and MLBP [115] and GWH-GLBP [47] are NR image quality methods, while V-BLINDS [46], and SSDCT [62] are NR video quality methods. In these methods, SVR is used for quality predictions. RFR is based on ensemble learning, in which multiple decision trees are grouped [111]. For example, the FR video quality method FREITAS2018 [55] has used the RFR model for quality prediction. The LrR method aims to obtain a line that best fits the instances. The best fit line is the one for which the total prediction error is as small as possible [116]. Recently, Freitas *et al.* [55] proposed a FR video quality assessment method, which extracts spatial and temporal features, and uses an RFR model to predict video quality scores. Hui *et al.* [50] used hand-made spatial and temporal features and the SVM model to blindly predict video quality.

2.4 Deep Learning Methods

Traditional ML-based algorithms rely on hand-crafted features obtained from feature engineering task. But with the increase in unstructured data (text, images, videos, audios, etc.) in the current digital world, feature engineering with efficient third-party algorithms is time-consuming, unfeasible, and prone to error. On the other hand, Deep learning (DL)-based methods do both feature engineering and learning from these features without human intervention, which is why the DL has become more popular over time.

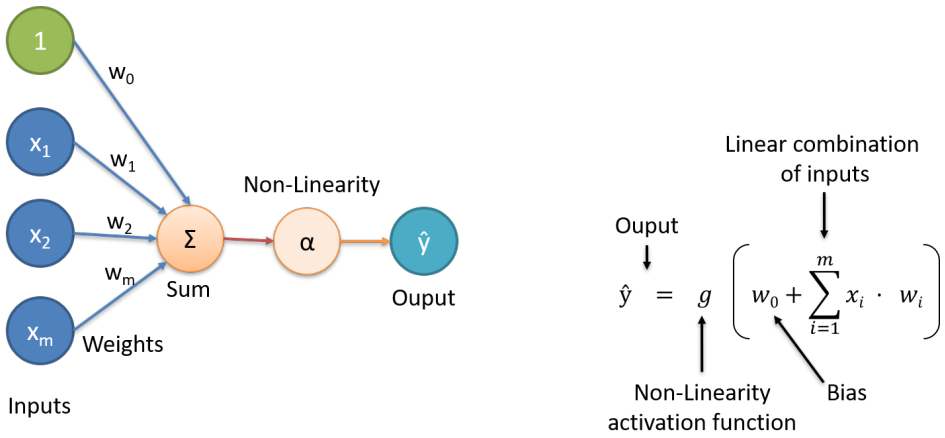


Figure 2.4.1: A perceptron in forward propagation.

Deep learning (DL) is a subset knowledge domain of machine learning. It consists of Neural Networks. The formulation of Neural Networks (also known as artificial neural networks (ANN)) is inspired by the human brain. The human brain consists of billions of neurons interconnected to each other. Each neuron receives the signal, processes it, and passes it to the other neurons. This is how the information is passed on in our brain. Likewise, deep learning focuses on using neural networks to automatically extract patterns in raw data and then using these patterns or features to learn how to perform a task. Traditionally, machine learning algorithms define a set of features in the data. Usually, these features are handcrafted or hand-engineered, and, as a result, they tend to be pretty brittle in practice. The key idea of deep learning is to learn these features directly from data in a hierarchical manner, i.e., to detect a face for example, start by detecting the edges in the image, composing these edges together to detect middle-level features, such as an eye, or a nose or mouth, and then, going deeper, composing these features into structural or facial features to finally recognize the corresponding face. This hierarchical way of thinking is really a core to deep learning.

An important question arises, "why we are considering deep learning now?". The answer to this question is that, the data has become much more pervasive now. Deep learning models are extremely hungry for data, and we are able to get huge amount of data easily from different online sources. Second, now we have powerful GPU hardware to run deep learning algorithms in parallel processing. And finally, due to open source toolboxes like TensorFlow, Keras, and PyTorch, building and deploying these models has become streamlined.

The fundamental building block of deep learning is a single neuron (also known as a perceptron). As shown in Figure 2.4.1, a single neuron works in a format of forward propagation of information passing through it. We can divide a set of inputs x_i to x_m, and each of these inputs or each of these numbers is multiplied by their corresponding weights w_i

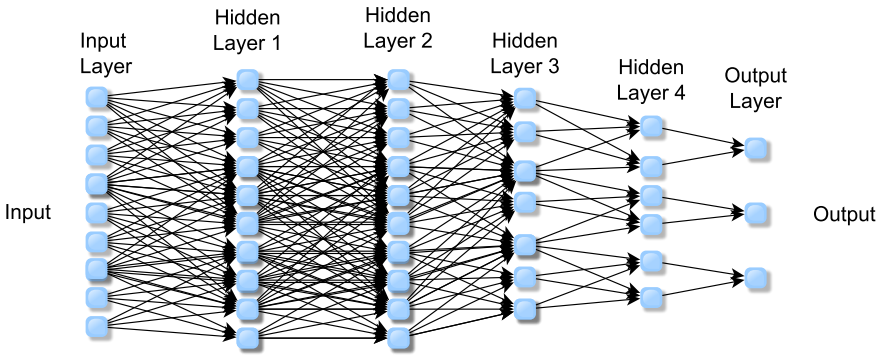


Figure 2.4.2: Deep Neural Network (DNN).

and then added together. We take this single number, which is the result of addition, and pass it through a non-linear activation function to produce our final output $\hat{y}$. We also have a bias function, which is a shift activation function. The right-hand side of the figure illustrates the forward propagation of a perceptron, and left-hand side illustrates mathematical representation of a perceptron. We can re-write this concept in more concise way as follows:

$$\hat{y} = g(w_0 + \mathbf{X}^{\mathbf{T}}\mathbf{W}) \tag{2.4.1}$$

where $\mathbf{X}$ represents a vector of inputs x_1 to x_m at time T, $\mathbf{W}$ represents a vector of weights from w_1 to w_m. The output $\hat{y}$ is simply obtained by taking the dot product of $\mathbf{X}$ and $\mathbf{W}$, adding a bias w_0, and then applying a non-linearity g.

Using one perceptron, we can build a Deep Neural Network (DNN), by simply stacking the layers of perceptrons to create more and more hierarchical models, where the final output is computed by going deeper and deeper into the network. Figure 2.4.2 shows an example of a deep neural network with many hidden layers and many nodes in every hidden layer. Each layer has perceptrons or neurons interconnected to the neurons in the next layer. Input layer is the layer in which we feed the input. Number of nodes in this layer depends on the number of the dimensions of the data. Output layer is the layer in which the output is generated. The number of nodes depends on the number of classes in the classification problem or of the scalar values in the regression problem. The hidden layer is like the "black box" in which the feature extraction takes place. The number of hidden nodes and number of hidden layers are arbitrary. For example, in image classification, every hidden layer extracts features that help in identifying the images. The first hidden layer may extract features such as edges. The second hidden layer builds upon the features extracted from the first layer and may extract features related to the objects, e.g., the structure of different faces. The more we increase hidden layers, the more complex features are extracted [110].

To mathematically represent this deep neural network, first we define the dot product, summation of input vectors, and their corresponding weights:

$$z_{k,i} = w_{0,i}^{k} + \sum_{j=1}^{n_{k-1}} g(z_{k-1,j})\, w_{j,i}^{(k)}, \tag{2.4.2}$$

where k is the number of layers, n is the number of inputs, $w_{j,i}$ is the i^{th} weight of the perceptron of the j^{th} input, $w_{0,i}^{k}$ is the bias of the i_{th} input of the k_{th} layer, and z represents the dot product and summation of the input vectors and their corresponding weights right before applying the nonlinearity, and it can be written as follows:

$$z_i = w_{0,i}^{k} + \sum_{j=1}^{m} x_j\, w_{j,i}^{k}. \tag{2.4.3}$$

Then, we can obtain our output $\hat{y}$ as follows:

$$\hat{y} = g\left(w_{0,i}^{k} + \sum_{j=1}^{d_k}\right) g(z_j)\, w_{j,i}^{k}, \tag{2.4.4}$$

where where k is the number of layers, $w_{j,i}^{k}$ is the weight of the j^{th} perceptron of the i^{th} input of the k_{th} layer, z_j is the output of the j^{th} perceptron, d_k represents the desired output value of the perceptron in layer k, and g is a nonlinear activation function.

The non-linear activation function allows us to deal with nonlinear data because, in the real world, data are always non-linear. In quality assessment methods, the Exponential Linear Unit (ELU) activation function [117] is commonly used because this function tends to converge faster and produces accurate results. When training a neural network, we want to find a network that minimizes the empirical loss (average loss over entire dataset) between the predictions and the ground truths (MOS in the case of quality assessment). For quality assessment methods, Mean Squared Error (MSE) loss is a commonly used loss, which can be computed as follows:

$$MSE = \frac{1}{n} \sum_{i=1}^{n} (Y_i - \hat{Y}_i)^2, \tag{2.4.5}$$

where n is the number of data points, Y_i represents ground truth values, and $\hat{Y}_i$ represents the predicted values.

To find the weights of the neural network, that will minimize the loss of training dataset, optimization functions are used, such as Stochastic Gradient Descent (SGD) [118], which is commonly used for loss optimization. In the training process, the number of steps by which the loss optimization function achieves local minima is defined by specifying the learning rates.

For quality assessment methods, the most commonly used deep learning methods are the Convolutional Neural Network (CNN) and the Long-Short-Term Memory Network (LSTM). CNN consists of convolutional layers that perform a convolution operation on multidimensional input images. Let us take an example of a 2D image. Suppose that we have a 4×4 patch or filter, which will consist of 16 weights. We are going to apply this same 4×4 filter in the input and use the result of that operation to define the state of the neuron in the next layer. So, the neuron in the next layer will be defined by applying this patch with a filter of equal size and learned weights. Then, we are going to shift that patch on the input image by one pixel to grab the next patch and compute the next output neuron. This is how the convolution operation works. The idea of convolution is to preserve the spatial relationship between pixels by learning the features of the image using small patches of the image.

Each neuron in the CNN layer will compute a weighted sum of each of its patch inputs. We apply and activate the neuron with some nonlinear activation function, so that we can handle nonlinear data relationships. We also need to add a bias in summation operation, that allows shifting the activation function. In other words, we can say that, each neuron in the hidden layer only sees a very specific patch of its inputs. It does not see all input neurons. In this case, each neuron output observes only a very local connected patch as input. We take a weighted sum of those patches, we apply a bias, and then we obtain a feature map (FM) as a result of a convolution layer. The feature map represents the state of the neuron in the next layer. We can define the convolutional layer mathematically as follows:

$$y_{FM} = g\left(\sum_{i=1}^{m}\sum_{j=1}^{n} w_{i,j}\, x_{(r(i)+p,r(j)+q} + b\right) \tag{2.4.6}$$

where $w_{i,j}$ represents $i \times j$ filter or patch matrix, $x_{i+p,j+q}$ represents the patch of size $p \times q$ in the input image x and r is the dilation rate (or Atrous rate). Specifically, using this equation, an element-wise multiplication is performed using every element in w by the corresponding elements in the input x. We add the bias b and activate it with non-linearity g. For each neuron in the hidden layer, y_{FM} becomes the input of the neurons in the next layer.

The Atrous or dilation rate is used to effectively enlarge the receptive field of kernel and capture abundant features, without increasing the number of parameters. For example, a 3×3 kernel with a dilation rate of 2 will have the same field of view as a 5×5 kernel, while only using 9 parameters. Traditional CNN layers use the default Atrous rate = 1, which means that no dilation is performed. By specifying a Atrous rate greater than 1, zeroes are added between the weights of the convolution kernel. Dilated convolutions are particularly popular in the field of real-time computer vision problems [119].

The general layers in DNN-based deep learning methods are as follows [120–122]:

- Input Layer: The input layer is the layer associated with the input image data. The

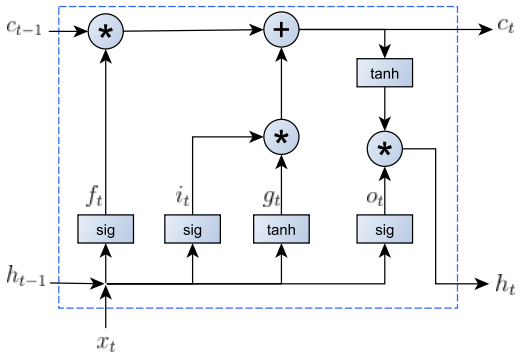


Figure 2.4.3: General structure of LSTM unit.

input layer is usually a tensor with dimensions, such as the dimensions of the input image, namely the length, width and number of channel images or their transformations.

- Convolution Layer: Convolutional layers are layers that carry out the convolution process from the previous layer. This layer stores the parameters or weights of the training results. The output of this layer (in the form of a tensor, often referred to as a feature map) usually has length and width smaller than the input layer but a greater depth. The movement of the filter in the image is controlled by the *stride* parameter [122].
- Activation Layer: It is an activation function that decides the final value of a neuron, as described in equation 2.4.1. These functions convert linear input signals into non-linear output signals, which aids the learning of deep networks.
- Pooling Layer: This layer is responsible for reducing the spatial size of the convolved feature. This is to decrease the computational power required to process the data through a dimensionality reduction, yet extracting dominant features that are rotation and position invariant.
- Fully Connected Layer: It is the final layer that functions as a classifier or a regressor. This layer generally uses artificial neural networks that can be trained. This layer stores the weight of the training results.

The Long Short-Term Memory (LSTM) network is a special kind of recurrent neural network, which is widely used in many tasks such as text generation and speech recognition [123, 124]. The LSTM takes one-dimensional (1D) vector (often formatted by a Reshape layer) $r^{(t)}$ as input and generates another 1D vector $h^{(t)}$. A conventional LSTM unit includes an input activation function, a single memory cell, and 3 gates, named as input gate i_t, forget gate f_t and output gate o_t. The sigmoid non-linearity is set as $\sigma(x) = (1 + e^{-x})^{-1}$, which maps the input data to the interval [0, 1]. The hyperbolic tangent nonlinearity is set as

$\varphi(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = 2\varphi(2x)$, which maps the input data into the intervals [-1, 1]. The mathematical representation of a general LSTM unit is as follows [125]:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i) \tag{2.4.7}$$

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f) \tag{2.4.8}$$

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o) \tag{2.4.9}$$

$$g_t = \tanh(W_c[h_{t-1}, x_t] + b_c) \tag{2.4.10}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{2.4.11}$$

$$h_t = o_t \odot \sigma(c_t) \tag{2.4.12}$$

where W_{xi} is the relevant weight matrix between layers, b_i is the bias, c_t is the memory cell unit that is a summation of the previous memory cell unit c_{t-1} controlled by the forget gate f_t, while the input modulation gate g_t is controlled by input gate i_t, h_t is the hidden unit, and $\odot$ is the element-wise product with the gate values. Figure 2.4.3 illustrates a general LSTM unit where the direction of arrows show input, output, and forget gates operations. The main advantage of the LSTM layer is that its memory cells extract long-term dependent distortion-related features from input.

Deep CNN architectures typically demand a sufficient amount of data for effective training. Transfer learning (TL) enables the re-utilization of computationally intensive deep CNNs that are already pre-trained on a bench-marked dataset (also called as source domain) having a large number of images, for a new problem (known as target-domain) that is comprised of the small training dataset. State-of-the-art deep CNN models with optimized filter weights, that are learned from the source domain (r.g., ImageNet dataset), are fine-tuned on multidimensional images (target domain) to effectively learn the target-domain-specific features from a limited amount of input samples. TL helps provide a useful set of feature descriptors learned from the source domain to effectively apply in a target domain by adapting them to the target task via fine-tuning. These feature descriptors are called bottleneck features. The bottleneck features are the last activation maps before the fully connected layers in the source network.

In the literature, several methods have been proposed that employed the deep learning models mentioned above for quality prediction. For example, Kang *et al.* [126] proposed the first NR image quality assessment (IQA) (2D images) method that used a CNN. In their work, 32×32 patches are used as input to the CNN. Apart from input and output layers, in the hidden layer, there is one convolution layer, one pooling layer, and two fully-connected layers. SACONVA [54] is a NR image quality assessment metric, in which hand crafted features are mapped by CNN. Most recently, Domonkos [127] has developed an NR-VQA method that uses frame-level features, which were obtained from a pre-trained CNN using transfer

learning. A temporal pooling using a regression algorithm (SVR) is used to aggregate these frame-level features for each video and predict the overall quality. Singh and Aggarwal [56] have proposed an NR-VQA model in which spatial and temporal features are extracted by a three-dimensional Local Binary Pattern (LBP) operator. These features are mapped to a single scalar quantity using a simple two-layer feed-forward artificial neural network (ANN) with a single hidden layer of four neurons. Ahn and Lee [51] have proposed an NR Deep Blind Video Quality Assessment (DeepBVQA) method, in which spatial features are extracted by a CNN, named BIECON [128], while temporal features are hand-crafted. The final video quality score is computed generating a feature vector by aggregating the pooled frame-level features. Domonkos and Szirányi [129] developed an NR-VQA model based on a long short-term memory (LSTM) network. The method considers the video frames as a time series of deep features, extracted with the help of a CNN, and uses an LSTM network to predict the video quality scores. Recently, the use of deep Convolutional Neural Network (CNN) architectures have also become very popular for LF-IQA [83–85]. Up to our knowledge, there is a limited work available [85, 86], which trains a single stream CNN architecture using only the horizontal EPIs.

The connection structure between the layers makes deep neural networks suitable for processing signals in tensor forms, where the tensor elements are arranged in a meaningful order. This fixed input order is a cornerstone for neural networks to extract higher-level features. For example, if we randomly shuffle the pixels of an image, then traditional CNN networks will fail to recognize it. Although images and many other types of data are naturally presented with order, there is another major category of structured data, namely graphs, which usually lack a tensor representation with fixed ordering. Dedicated Graph Convolutional Neural Networks [130, 131] (GCNN) have been developed, that learn from structured data and perform predictions. We generalize the images by their pixels in the graph, so each pixel indicates a particular node in the picture. If pixels have a relationship (connection), their edges will connect in a particular pair of nodes. Therefore in GCNN, the number of edges vary, and the nodes are unordered. In GCNN layer, convolution is applied to discover each pixel, and then their edges, and gradually the whole of neighbors, that are connected to a specific node. Having said that, the main role of the convolution is to cover the neighborhood of each node. To sum up, we tend to initialize the kernels by message passing with the neighbors of a corresponding node, and share this information by weight sharing. To the best of our knowledge, there is a lack of GCNN based quality assessment method for multi-dimensional visual contents.

2.5 Visual Attention

Visual attention (VA) is a technique of the HVS. When observing a scene, the human eye filters the large amount of visual information available, focusing on selected (salient) regions [132]. Figure 2.5.1(a) represents the VA technique, with the red circles depicting the salient regions that attract the human attention. This selection process is actively controlled through oculomotor techniques. These techniques allow the gaze of attention to hold on a particular location (fixation) or to shift to a preferred location when sufficient information has been collected from the current focus (saccades). Fixations are instinctively concentrated on highly informative areas. As a consequence, the amount of data to be further processed by the brain is minimized, while maximizing the quantity of useful information.

Based on how attention is stimulated, VA techniques are categorized as Bottom-up or Top-down. The bottom-up attention technique is stimuli-driven and based on salient features of the input image, such as orientation, colour, intensity and motion. The bottom-up attention technique is the outcome of a feature extraction across the whole visual field. Therefore, a highly salient region of a given input visual content can capture the focus of human attention. For example, flashing points of light on a dark night, sudden motion of objects in a static environment, and red followers on a green background can involuntarily and automatically attract human attention. The top-down attention technique, on the other hand, is task-driven and refers to the set of processes used to bias the visual perception based on task or intention. For example, when an observer is assigned a task of finding a black pen in a scene of a room crowded with many other things, he/she uses his/her prior knowledge, experience, and current goal to complete the task, which are mostly controlled by the high-level cortex that helps selecting the best region candidates. Bottom-up attention pops out only the candidate regions where targets are likely to appear, while top-down attention can depict the exact position of the target.

The algorithms or metrics that detect salient regions automatically, without human intervention, are called saliency computational models. For example, ITTI [133] and GBVS [134] are the most common and widely used bottom-up saliency models. They use conventional programming strategies to compute saliency, in which saliency is generated in three steps. In the first step *Extraction*, the lower level features (contrast, luminance, and textures) are extracted from the images, and each feature is converted into a corresponding vector. In the second step *Activation*, saliency maps are generated from each feature vector. Finally, in the third step *Combination*, saliency maps are combined into one final saliency map. For example, the Boolean Saliency Map (BMS), which is a conventional bottom-up saliency model, first generates all possible Boolean maps of an image, and then applies a threshold to them to create activation maps and a final saliency map is generated by computing the mean of all these maps.

(a) Visual attention (red circles depict the salient regions).

(b) Saliency Weighted Quality Assessment

Figure 2.5.1: Incorporation of visual attention into OQA methods.

Visual attention plays an important role in quality assessment (QA). Any distortion that occurs in a salient area is more important to the overall perceived quality. That is why distortions that occurs in the salient areas should be treated differently from the distortions in less salient areas. For this purpose, visual attention is used to improve the accuracy of image quality assessment metrics, i.e., to make the assessment results closer to the subjective score. In most existing works, visual attention is used as a weighting factor to spatially pool the objective quality scores from the quality map [135]. Figure 2.5.1(2) depicts this process of weighing salient regions. In this figure, spatial features are extracted using SSIM, and the saliency map is extracted using the BMS. The weighted map is generated as follows:

$$W_{map} = \frac{\sum_{x,y} \varepsilon(x,y) \bullet \omega(x,y)}{\sum_{x,y} \omega(x,y)}, \tag{2.5.1}$$

where, $\varepsilon(x,y)$ is the error map and $\omega(x,y)$ is the saliency map at pixel positions x and y. W_{map} denotes the weighted map and $\bullet$ denotes the weighting operator.

Recently, a variety of conventional 2D image OQA methods [136, 137] and video OQA methods [17, 18, 39, 40] and [138–141], have incorporated the saliency information to improve their quality predictions. The local distortions are weighted by the local saliency, with larger weights for salient areas and smaller weights for non-salient areas. For example, Zhang *et al.* [37, 38] have presented detailed statistical evaluations on the performance of saliency-weighted OQA methods for both images and videos. Using a CNN model, an NR-IQA method [5] has been proposed which incorporates saliency information into the quality prediction of images. And, since the eye is naturally attracted to moving objects, several works have tried to estimate the amount of motion in a video, often using optical flow algorithms. For example, Gujjunoori and Orungati [60] and Aabed and Al-Regib [61] proposed FR-VQA and RR-VQA metrics, respectively, that used optical flow features and conventional

pooling and mapping strategies to estimate video quality.

Considering the importance of incorporating saliency information in 2D quality assessment methods, several works have also been proposed for saliency prediction in LF images [119, 142–144] using different formats of LF images. In research, it is still a question that, to improve the prediction performance of quality assessment methods, which format of LF images we should consider, or where should we look for saliency in quality assessment field. For example, Lamichhane *et al.* [84] presented a full-reference LFI quality assessment method that is based on a CNN network. In this method, the impact of the use of saliency map has been addressed. The method extracted saliency information from the LFIs, and passed this information to a CNN network for training The results achieved show a high correlation between a measure of distortion in an image and the saliency map, in accordance with subjective quality scores. Although this work has shown significant improvement in comparison with the quality prediction without saliency, it has certain limitations. For example, the saliency models used (ITTI [133], GBVS [134], Geometry [145], BMS [146] and EBMS [147]) are designed for 2D images. Also, only one type of distorted LFIs representation is considered. Finally, there is a lack of research that incorporates salience information extracted by LF image saliency models, considering both spatial and angular information.

2.6 Visual Quality Databases

Table 2.6.1: Summary of 2D Image and Video Quality Datasets.

Dataset	Year	Type	Source/Test[1]	Resolution[2]	Distortion Types	Availability
CSIQ [148]	2006	Videos	12/216	832x480	H.264/AVC, H.264/PLR, MJPEG, WC, WN, and HEVC	Not Available
LIVE [149]	2018	Videos	10/150	768x432	TE, IP error, H264, and MPEG2	`https://live.ece.utexas.edu/research/Quality/`
TID2013 [150]	2015	Images	25/3000	512x384	AGN, AGC, SCN, MN, HFN, IN, QN, GB, ID, JPEG2000, JPEGTE, JPEG2000TE, NEPN, LBD, IS, CC, CCS, MGN, CN, LC, ICQ, CA, SSR	`https://ponomarenko.info/tid2013.htm`
CSIQ [151]	2010	Images	30/866	512x512	JPEG, JPEG2000, WN, GB, CD, and PN	Not Available
LIVE [152]	2006	Images	29/982	480x720, 610x488, 618x453, 627x482, 632x505, 634x438, 634x505, 640x512, and 768x512	JPEG, JPEG2000, WN, GB, and FF	`https://live.ece.utexas.edu/research/quality/subjective.htm`

[1]Data in this column shows the number of source (reference) content / total number of test contents.

[2]This column represents the resolution in the following format: width x height.

2.6.1 2D Images and Videos

For 2D image quality assessment, the most commonly used databases are as follows:

- Laboratory for Image and Video Engineering (LIVE) Image Database version 2 [152]: The database presents 982 test images, including 29 originals and 5 categories of distortions. These images are in uncompressed BMP format at several dimensions, including 480×720, 610×488, 618×453, 627×482, 632×505, 634×438, 634×505, 640×512, and 768×512. The distortions include JPEG, JPEG 2000 (JPEG2k), white noise (WN), Gaussian blur (GB), and fast fading (FF).
- Computational and Subjective Image Quality (CSIQ) Database [151]: The database contains thirty reference images obtained from public-domain sources and 6 categories of distortions. The distortions include JPEG, JPEG 2000 (JPEG2k), white noise (WN), Gaussian blur (GB), global contrast decrements (CD), and additive Gaussian pink noise (PN). In total, there are 866 distorted images.
- Tampere Image Database 2013 (TID2013) [150]: The database 25 reference images and 3,000 distorted images (25 reference images×24 types of distortions×5 levels of distortions). These images are in 512×384×24 uncompressed BMP format. The distortions include Additive Gaussian noise (AGN), Additive noise in color components (AGC), Spatially correlated noise (SCN), Masked noise (MN), High frequency noise (HFN), Impulse noise (IN), Quantization noise (QN), Gaussian blur (GB), Image denoising (ID), JPEG, JPEG2k, JPEG transmission errors (JPEGTE), JPEG2k transmission errors (JPEG2kTE), Non eccentricity pattern noise (NEPN), Local block-wise distortions (LBD), Intensity shift (IS), Contrast change (CC), Change of color saturation (CCS), Multiplicative Gaussian noise (MGN), Comfort noise (CN), Lossy compression (LC), Image color quantization with dither (ICQ), Chromatic aberration (CA), and Sparse sampling and reconstruction (SSR).

For 2D video quality assessment, the most commonly used databases are as follows:

- Computational and Subjective Image Quality (CSIQ) [148]: This database has 12 reference videos and 216 distorted videos. There are six types of distortions in this dataset, namely, AVC compression (AVC), PLR video with packet loss rate (PLR), MJPEG compression (MJPEG), Wavelet compression (WC), White noise (WN) and HEVC compression (HEVC).
- Laboratory for Image and Video Engineering (LIVE) [149]: This database has 10 reference 150 distorted videos, distorted by wireless (TE), IP error, H264 and MPEG2 types of distortions.

Table 2.6.1 shows a summary of the main characteristics of these 3 image and 2 video quality datasets, including details of their availability (e.g. site for download).

2.6.2 4D Light Field Images

Table 2.6.2: Summary of 4D Light Field Image and Video Quality Datasets.

Dataset	Year	Type	Nof Subjects	Source/Test[1]	Resolution[2]	Distortion Types	Availability
VALID [153]	2018	Images	22 (Average)	5/140	625x434x13x13	HEVC, JPEG2000, and VP9 @ different bitrates.	https://www.epfl.ch/labs/mmspg/downloads/valid/
Win5-LID [3]	2018	Images	2 (Aged: 19-26)	10/220	Real Scenes =625x434x9x9; Synthetic Scenes =512x512x9x9	JPEG2000, HEVC, bilinear and nearest-neighbour interpolation @ different quantization parameters.	http://staff.ustc.edu.cn/~chenzhibo/resources/2018/win5_lid.html
MPI [7]	2017	Images	40 (Aged: 24-40)	14/336	960X720X101	HEVC, bilinear and nearest-neighbor interpolation, Gaussian blur, quantized depth maps, and image warping using optical flow estimation.	http://lightfields.mpi-inf.mpg.de/Dataset.html
SMART [154]	2016	Images	19	16/256	625x434x14x14	JPEG, JPEG2000, HEVC intra, SSDC @ different quantization parameters	http://www.comlab.uniroma3.it/SMART.html
LFDD [4]	2019	Images	20 (Aged: 20-36)	8/480	512x512x9x9	JPEG, JPEG2000, X264, BPG, VP9, AV1, AVC, HEVC, Gaussian, Impulse, Pincushion and Unsharp mask	https://sites.google.com/fel.cvut.cz/lfdd

[1]Data in this column shows the number of source (reference) content / total number of test contents.
[2]This column represents the resolution in the following format: u, v, s, t.

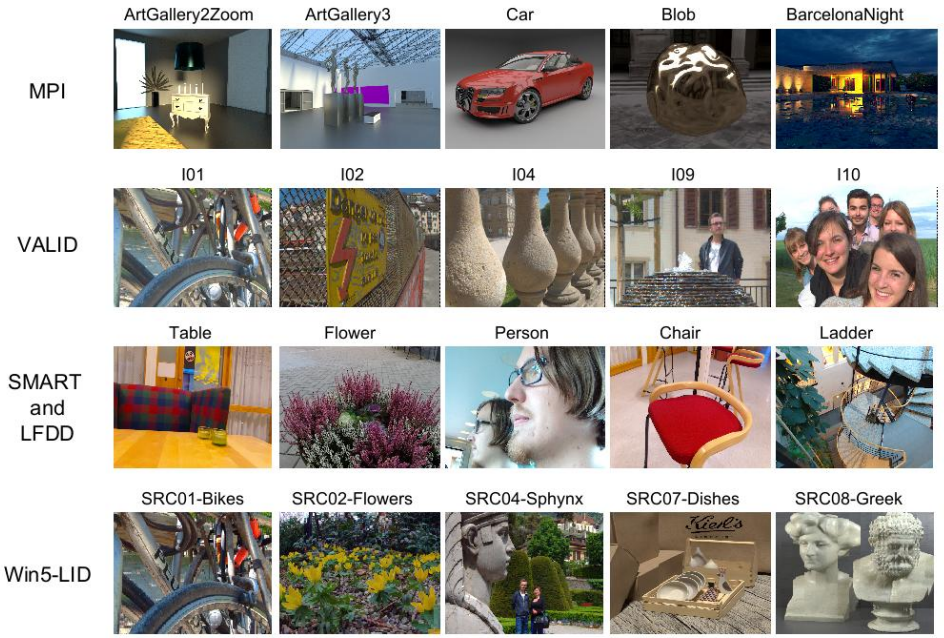


Figure 2.6.1: Sample images taken from 5 LF image quality datasets: MPI, VALID, SMART, Win5-LID, and LFDD.

Recently, researchers have developed several quality assessment methods to assess the visual quality of distorted LF images. These methods are tested on recently established datasets of distorted LF images, which also contain corresponding MOS values. In this work, we have used 5 light field image quality datasets. We have chosen these datasets because of the diversity of their visual contents, types of distortions and the availability of the corresponding subjective quality scores, as described next.

- The MPI dataset [7] contains 13 different source scenes and 336 distorted LFIs, which contain 6 distortion types (HEVC, linear and nearest-neighbor (NN) interpolation, Gaussian blur (GB), quantized depth maps, and image warping using optical flow estimation) with 7 degradation levels each. This dataset has light field distortions that are specific to transmission, reconstruction, and display. For each distortion, multiple test scenes have been generated by varying the distortion severity level. All LF images have the same spatial and angular resolution ($960 \times 720 \times 101$). The subjective experimental method considers the horizontal parallax of the LF images.
- The VALID dataset [153] contains 5 source contents, taken from the EPFL [155] light field image dataset, and 140 distorted LFIs generated by compressing the sources at various bitrates using state-of-the-art compression algorithms. The dimension of each LF image is $625 \times 434 \times 13 \times 13$. The dataset contains both subjective (MOS) and objective quality scores using Peak Signal-to-Noise (PSNR) and Structural Similarity (SSIM) [11] that are available for download.
- The SMART dataset [154, 156] has 16 source LFIs and 256 distorted sequences, with both indoor and outdoor contents. The dimension of each LF image is $625 \times 434 \times 14 \times 14$. The dataset has contents with different levels of colorfulness, spatial information, and texture, but also variations in reflection, transparency, and depth of field that are specific to LFs. The degradations consist of compression distortions, obtained with 4 codecs: HEVC Intra [157, 158], JPEG, JPEG2000, and SSDC [159].
- The Win5-LID dataset [3] contains 6 real scenes, and 4 synthetic scenes. The selected contents carry abundant semantic features, such as human, plant and object. All contents are of identical angular resolution 9×9 containing both horizontal and vertical angular offsets. The spatial resolutions of real scenes and synthetic scenes are 625×434 and 512×512, respectively. The images in the Win5-LID dataset are compressed and distorted using JPEG2000 and HEVC encoders. In total, there are 220 distorted images with the corresponding MOS.
- The LFDD dataset [4] dataset contains 8 reference and 480 distorted LFIs with JPEG, JPEG2000, X264, BPG, VP9, AV1, AVC, HEVC, Gaussian, Impulse, Pincushion and Unsharp mask types of distortions at different bitrates. Each distortion has three different distortion levels. The LFIs have resolution $512 \times 512 \times 9 \times 9$.

Table 2.6.2 shows a summary of the main characteristics of these 4 light field image quality datasets, including details of the subjective experiments and their availability (e.g. site for download). Figure 2.6.1 shows sample images of 5 LF-IQA datasets.

Chapter 3

Quality Assessment of 2D Images and Videos

In this chapter [1], we discuss the proposed methods for NR quality assessment of 2D images and videos. The contributions of this work are summarized as follows:

- The image quality assessment (IQA) method, named as the Multiscale Salient Local Binary Patterns (MSLBP), is proposed, which incorporates the saliency-weighted textural features. We use these features to train a machine learning algorithm called Random Forest Regressor (RFR).
- The video quality assessment (VQA) method is proposed, which is based on a Convolution Neural Network (CNN) architecture. The method employs a spatio-temporal saliency patch-selection procedure to obtain a selected number of patches from the video frames. This procedure crops a frame into small non-overlapping blocks of images (patches), and selects the most perceptually relevant ones. The selected patches are then forwarded to the CNN for training.
- The efficiency and robustness of both IQA and VQA methods are proved through cross-dataset analysis.

3.1 The Multiscale Salient Local Binary Patterns for Image Quality Assessment

This method uses an extension of multiscale local binary pattern (MLBP) algorithm [115]. The spatial features extracted by the MLBP are weighted by the saliency information. Then,

[1] This chapter contains the research material published by ACM Multimedia Systems [160], and Journal of Electronic Imaging [161]

the weighted features are used as input to a supervised machine learning algorithm RFR that predicts final image quality score.

The MLBP is a variant of local binary pattern (LBP), and extracts features relevant to image quality. It generates several LBP maps by varying the parameters R and P and performs symmetrical sampling. In this work, we use MLBP to compute the LBP for all pixels of an image and obtain a set of LBP maps ($\mathscr{L}_R^P$). Each map $\mathscr{L}_R^P(x,y)$ corresponds to the local texture associated to the pixel $\mathscr{I}(x,y)$. Next, we use the Boolean Map saliency model (BMS) [162] to generate a saliency maps $\mathscr{W}$, with each component $\mathscr{W}(x,y)$ where pixel $\mathscr{I}(x,y)$ represent the most attracted regions in an image.

The saliency maps $\mathscr{W}$ are used to give a weight to each pixel of the LBP maps $\mathscr{L}_R^P$. This weighting process generates a feature vector based on the histogram of $\mathscr{L}_R^P$ weighted by $\mathscr{W}$. Particularly, the histogram is generated as:

$$H_R^P = \{h_R^P(0), h_R^P(1), \cdots, h_R^P(P+1)\} \tag{3.1.1}$$

where:

$$h_R^P(\phi) = \sum_{x,y} \mathscr{W}(x,y) \cdot \delta(\mathscr{L}_R^P(x,y), \phi), \tag{3.1.2}$$

and

$$\delta(v,u) = \begin{cases} 1, & \text{if } v = u, \\ 0, & \text{otherwise.} \end{cases} \tag{3.1.3}$$

The number of bins of this histogram is similar to the number of different LBP labels in $\mathscr{L}_R^P$. So, each $\mathscr{L}_R^P(i,j)$ can be represented to its weighted form, generating the map $\mathscr{S}_R^P$. We name this weighted LBP map the salient local binary patterns (SLBP). Figures 3.1.1 (a) and (b) depict the examples of the input images and their saliency maps, respectively. Figures 3.1.1 (c) to (g) depict examples of LBP maps obtained using different radius values and different numbers of neighboring points. Figures 3.1.1 (h) to (l) display the SLBP maps generated from $\mathscr{W}$ and their corresponding $\mathscr{L}_R^P$.

The Multiscale Salient Local Binary Patterns (MSLBP) algorithm generates different SLBP histograms at different scales, as illustrated in Figure 3.1.2. These histograms are concatenated to produce a feature vector for each image as:

$$\mathscr{H} = H_1^4 \oplus H_1^8 \oplus H_2^4 \oplus H_2^8 \oplus H_2^{16} \oplus \cdots \oplus H_R^N, \tag{3.1.4}$$

where $\oplus$ denotes the concatenation operator.

The computed feature vector $\mathscr{H}$ is supplied as input to random forests (RFR) regression algorithm. We chose RFR approach because in previous studies it has shown best performance [163] when compared to other machine learning algorithms (e.g. neural networks,

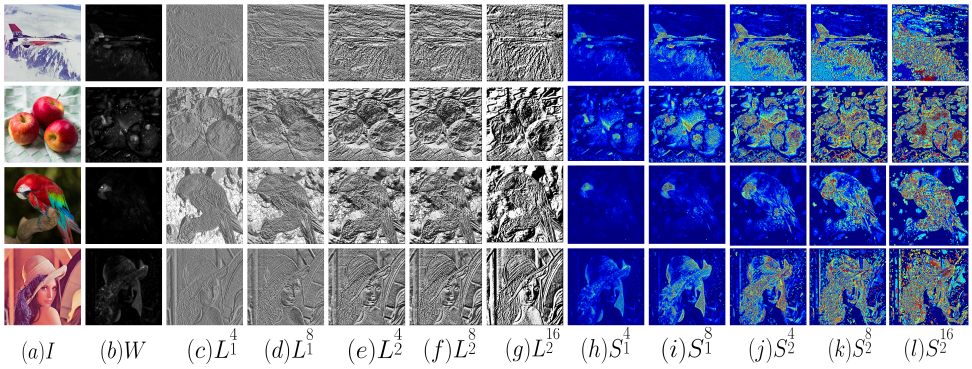


Figure 3.1.1: Example of original images (a), their saliency maps (b), LBP maps (c)-(g), and SLBP maps (h)-(l).

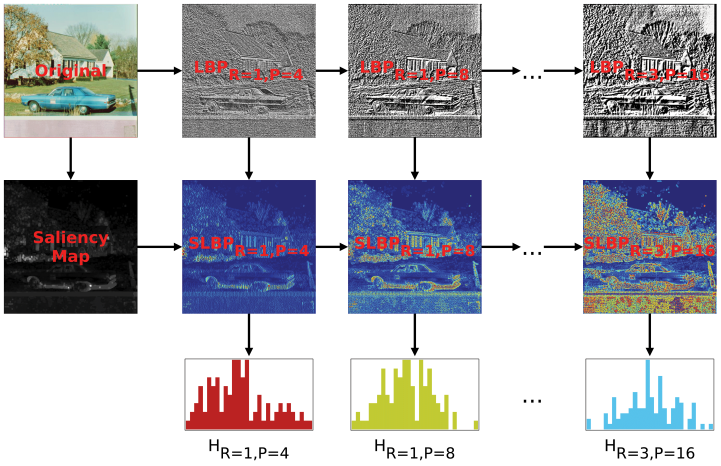


Figure 3.1.2: Multiple histogram generation from SLBP.

support vector machines, generalized linear models, etc.).

3.1.1 Experimental Setup

For implementation, we have used the following three databases:

- Laboratory for Image & Video Engineering (LIVE) Image Database version 2 [152].
- Computational and Subjective Image Quality (CSIQ) Database [151].
- Tampere Image Database 2013 (TID2013) [150].

To compare the performance, we have chosen a set of publicly available IQA methods. The chosen state-of-the-art NR-IQA methods are CORNIA [64], CQA [45], SSEQ [48], BRISQUE [112], LTP [113], and MLBP [164]. Additionally, we also compared the proposed algorithm with three well-established FR-IQA metrics, namely PSNR [165], SSIM [11], and RIQMC [21].

For the training-test procedure, we split each database into two content-independent subsets: train and test. Image contents (scenes) in the test subset are not present in the training subset, and vice-versa. Considering this, 20% of images are randomly selected for testing and the remaining 80% are used for training. This 80-20 split procedure corresponds to one simulation. We performed 1,000 simulations and the mean correlation is reported. The training and predicting steps are implemented using the Sklearn library [166]. The SVR meta-parameters are found using grid search methods provided by Sklearn's API. Likewise, Sklearn is used to implement the RF regression of the proposed method.

The simulations are performed using an Intel i7-4790 processor at 3.60GHz. The performance of tested methods is measured by comparing the predicted quality scores with the subjective quality scores. To compare the predicted and subjective scores, we computed the SROCC and PLCC, and the Root-mean-squared-error (RMSE) between these values. We generated the MSLBP features using MATLAB, and trained and tested the RFR on these features in Python, with LINUX environment.

3.1.2 Statistical Evaluation

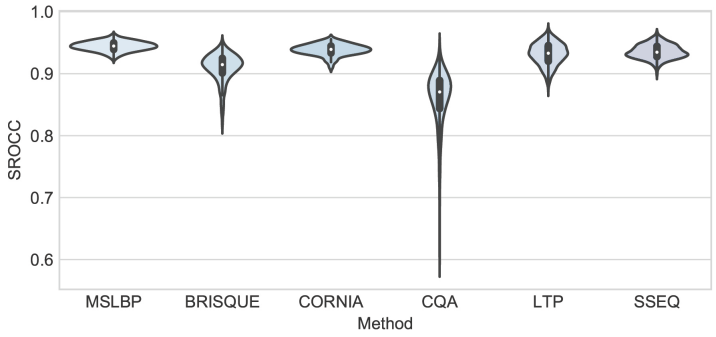
Table 3.1.1 depicts the simulations results on selected databases. In this table, numbers in italics represent the maximum correlation values among all tested methods (both NR-IQA and FR-IQA), while numbers in bold correspond to the best correlation values considering only the NR-IQA methods. From this table, we can see that, for most databases, the proposed method achieves the best performance among the NR-IQA methods

For the LIVE database, the proposed method outperforms current NR-IQA methods for WN, GB, FF, and 'ALL' distortions. In this database, the proposed method also outperforms the FR-IQA methods for GB and 'ALL' cases. Notice that for most metrics, SROCC is decreased for the distortions FF. Also, the proposed method shows little variation of the SROCC for the different distortions. For the CSIQ database, the proposed method outperforms current NR-IQA methods for WN, PN, and 'ALL' distortions, while LTP presents the best results for JPEG, JPEG2k, and CQA distortions. Notice that all NR-IQA methods have very low SROCC values for the distortion CD. Again, among the NR-IQA methods, although the proposed method also present a low correlation value for CD, in general it shows a smaller variation of SROCC. As TID2013 has higher variety of types of distortions, from Table 3.1.1, we noticed that the AGC, AGN, CA, CC, CCS, CN, IN, IS, LBD, LC, MN, and NEPN distortions have smaller SROCC values for both NR-IQA and FR-IQA methods. Notice that these distortions correspond to distortions in color, contrast distortions, and more complex types of noise. Although the proposed method also presented low SROCC values for these distortions, it has the best performance for 16 out of 25 cases and the best overall performance, followed by LTP and BRISQUE.

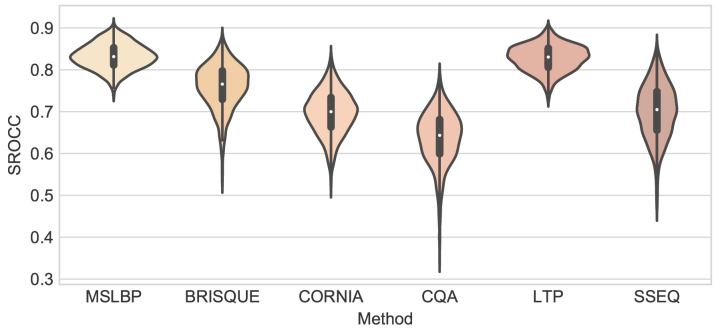
Table 3.1.1: Mean SROCC of tested FR-IQA (PSNR, SSIM, and RIQMC) and NR-IQA (BRISQUE, CORNIA, CQA, SSEQ, LTP, and MSLBP) methods, obtained from 1,000 runs on LIVE, CSIQ, and TID2013 databases.

Database	Distortion	PSNR	SSIM	RIQMC	BRISQUE	CORNIA	CQA	SSEQ	LTP	MSLBP
LIVE	JPEG	0.8515	*0.9481*	0.7794	0.8641	0.9002	0.8257	0.9122	**0.9395**	0.9165
	JPEG2k	0.8822	*0.9438*	0.5383	0.8838	0.9246	0.8366	**0.9388**	0.9372	0.9316
	WN	*0.9851*	0.9793	0.6628	0.9750	0.9500	0.9764	0.9544	0.9646	**0.9814**
	GB	0.7818	0.8889	0.8711	0.9304	0.9465	0.8377	0.9157	0.9530	***0.9553***
	FF	0.8869	*0.9335*	0.6802	0.8469	0.9132	0.8262	0.9038	0.8758	**0.9255**
	ALL	0.8013	0.8902	0.6785	0.9098	0.9386	0.8606	0.9356	0.9316	***0.9446***
CSIQ	JPEG	0.9009	*0.9309*	0.7242	0.8525	0.8319	0.6506	0.8066	**0.9292**	0.9211
	JPEG2k	*0.9309*	0.9251	0.5795	0.8458	0.8405	0.8214	0.7302	**0.8877**	0.8701
	WN	*0.9345*	0.8761	0.4678	0.6931	0.6187	0.7276	0.7876	0.6454	**0.8322**
	GB	*0.9358*	0.9089	0.8007	0.8337	0.8526	0.7486	0.7766	**0.9244**	0.8937
	PN	*0.9315*	0.8871	0.3653	0.7740	0.5341	0.5463	0.6661	0.7828	**0.8061**
	CD	0.8862	0.8128	*0.9565*	0.4255	0.4458	**0.5383**	0.4172	0.2082	0.4751
	ALL	0.8088	0.8116	0.5066	0.7597	0.6969	0.6369	0.7007	0.8280	***0.8314***
TID2013	AGC	*0.8568*	0.7912	0.3555	0.4166	0.2605	0.3964	0.3949	**0.5963**	0.4879
	AGN	*0.9337*	0.6421	0.6055	0.6416	0.5689	0.6051	0.6040	**0.6631**	0.6458
	CA	*0.7759*	0.7158	0.5726	**0.7310**	0.6844	0.4380	0.4366	0.6749	0.5694
	CC	0.4608	0.3477	*0.8044*	0.1849	0.1400	**0.2043**	0.2006	0.1886	0.1723
	CCS	0.6892	*0.7641*	0.0581	**0.2715**	0.2642	0.2461	0.2547	0.2384	0.2101
	CN	*0.8838*	0.6465	0.6262	0.2176	0.3553	0.1623	0.1642	0.3880	**0.5331**
	GB	0.8905	0.8196	0.7687	0.8063	0.8341	0.7019	0.7058	0.7465	***0.8961***
	HFN	*0.9165*	0.7962	0.4267	0.7103	0.7707	0.7104	0.7061	0.7626	**0.8507**
	ICQ	*0.9087*	0.7271	0.8691	0.7663	0.7044	0.6829	0.6834	0.7603	**0.8184**
	ID	*0.9457*	0.8327	0.8661	0.5243	0.7227	0.6711	0.6716	0.7063	**0.8011**
	IN	*0.9263*	0.8055	0.1222	**0.6848**	0.5874	0.4231	0.4272	0.6484	0.5879
	IS	*0.7647*	0.7411	0.5979	0.2224	0.2403	0.2011	0.2013	**0.3291**	0.1523
	JPEG	*0.9252*	0.8275	0.7293	0.7252	0.7815	0.6317	0.6284	0.6631	**0.8387**
	JPEGTE	*0.7874*	0.6144	0.6009	0.3581	0.5679	0.2221	0.2195	0.2314	**0.6179**
	JPEG2k	0.8934	0.7531	0.5967	0.7337	0.8089	0.7219	0.7205	0.7780	***0.9283***
	JPEG2kTE	*0.8581*	0.7067	0.7189	0.7277	0.6113	0.6529	0.6529	0.6594	**0.7308**
	LBD	0.1301	*0.6213*	0.2471	0.2833	0.2157	0.2382	0.2290	**0.3813**	0.2081
	LC	*0.9386*	0.8311	0.5346	0.5726	**0.6682**	0.4561	0.4460	0.6533	0.3153
	MGN	*0.9085*	0.7863	0.3751	0.5548	0.4393	0.4969	0.4897	0.6209	**0.6482**
	MN	*0.8385*	0.7388	0.0438	0.2650	0.2342	0.2506	0.2575	**0.4243**	0.2433
	NEPN	*0.6931*	0.5326	0.1496	0.1821	0.2855	0.1308	0.1275	0.1256	**0.3391**
	QN	0.8636	0.7428	*0.8697*	0.5383	0.4922	0.7242	0.7214	0.7361	**0.8569**
	SCN	*0.9152*	0.7934	0.7811	0.7238	0.7043	0.7121	0.7064	0.7015	**0.8173**
	SSR	*0.9241*	0.7774	0.6967	0.7101	0.8594	0.8115	0.8084	0.8457	**0.8675**
	ALL	0.6869	0.5758	0.4439	0.5416	0.6006	0.4925	0.4901	0.6078	***0.7113***

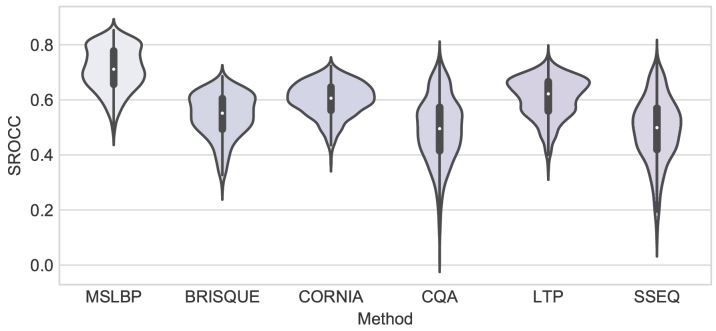
For further analysis, we used violin plots [167]. The violin plot is a combination of box plot with a rotated kernel density plot on each side. In the violin plots, the center point shows the median value, the black rectangle inside depicts the upper and lower quartiles (interquartile range), the vertical black lines corresponds to the values that occur 95% of the time, and the curves represents the distribution of data. Figure 3.1.3 depicts the violin plot of the SROCC values computed between the subjective scores (MOS) and the predicted scores obtained using the tested BIQA methods. The violin plots are generated using the distribution of SROCC values for the set containing all database distortions (corresponding



(a) LIVE



(b) CSIQ



(c) TID2013

Figure 3.1.3: Violin plot of SROCC distributions from 1000 runs of simulations on tested databases.

to "ALL" in Table 3.1.1). From Figure 3.1.3 (a), we notice that CORNIA and the proposed method present similar distributions of SROCC scores for the LIVE database. On the other hand, SROCC values vary more for CSIQ and TID2013 databases, as can be seen in Figure 3.1.3 (b) and (c).

3.2 Video Qualtiy Assessment based on Spatio-Temporal Patch-Selection Procedure

Figure 3.2.1 depicts the block diagram of the proposed NR-VQA method. As mentioned before, the proposed method is based on CNN that takes selected patches from input frame. Each input frame is labeled as custom target quality score for target predictions. Instead of choosing patches (frame is cropped into small images of size 32×32) randomly, the method selects the most perceptually relevant patches of each frame. The patch selection procedure is performed by combining spatial and temporal saliency of frames. Later in this section, each procedure is discussed in detail.

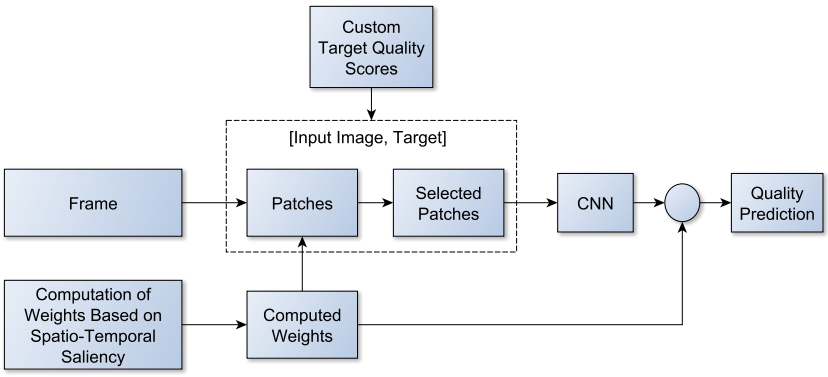


Figure 3.2.1: Block Diagram of the proposed no-reference video quality assessment method.

In this work, we use a CNN architecture named as "Visual Saliency Based Blind Image Quality Assessment via Convolutional Neural Network" [5] (VSBIQA), depicted in Figure 3.2.2, as the basis for our methodology. The VSBIQA CNN is an extension of deep neural network for no-reference and full-reference image quality assessment (deepIQA) CNN [168]. The VSBIQA CNN consists of a total of 10 layers including input and output. The first layer takes as input the selected 32×32 patches (RGB color format) of the video frames, where each patch represents different salient region. The second, fourth, and sixth layers are convolution layers with stride sizes of 5×5, 3×3, and 3×3, respectively. The third, fifth, and seventh layers are MaxPool layers of size 2×2. The eighth and ninth layers are fully connected layers of size 512. Finally, the output layer performs regression to predict quality score of each patch.

As shown in Figure 3.2.2, instead of processing the complete frame, the proposed method selects the most perceptually relevant (salient) regions from each frame, crops them into patches, and forwards the selected patches to the CNN. To determine the relevance of the regions in a frame and determine which regions are the most relevant, spatial and temporal saliency features of each frame are computed. To compute the spatial saliency (SS) of a

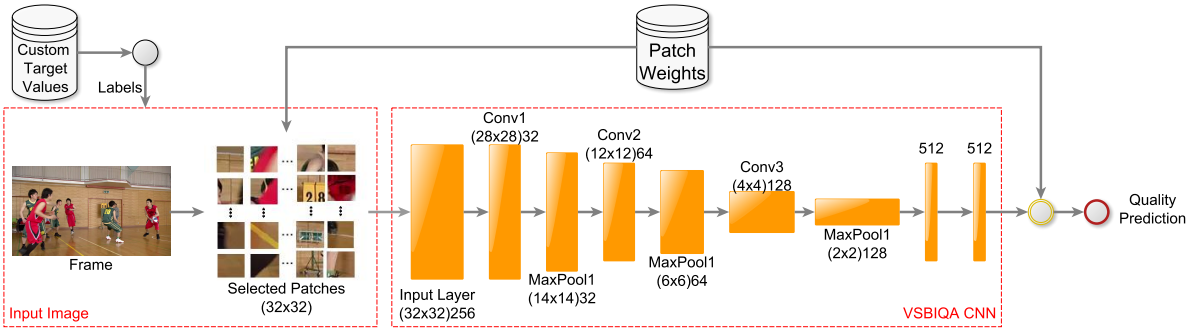
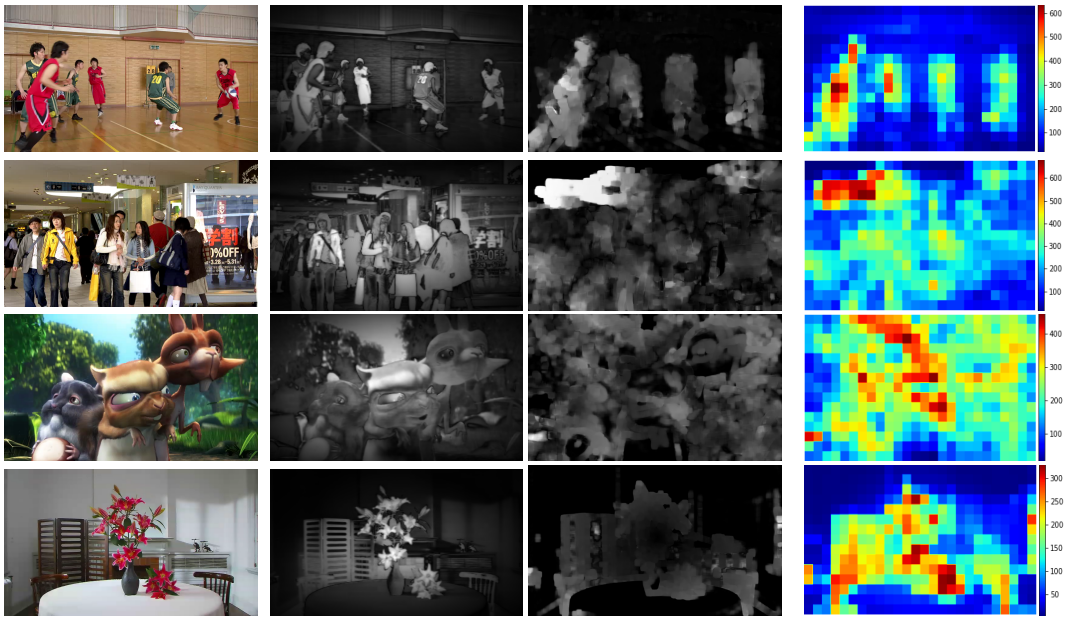


Figure 3.2.2: The image shows training process using VSBIQA CNN architecture [5]. Input frame is cropped into non-overlapping patches of size 32x32, then based on computed weights (according to equations 3.2.1 and 3.2.2), a certain number of top weighted patches are selected and supplied to CNN. For target prediction, input frame is labeled using custom target values. To compute final quality score for input frame, the predicted score is processed with computed weights of corresponding patches (according to equation 3.2.3).

frame, we use a bottom-up visual attention model named as "saliency detection method by combining simple priors" (SDSP) [169], which has a low cost of data-processing and a good performance. The SDSP algorithm has three major steps. First, it extracts features from the picture frames using a band-pass filter. Then, the frames are converted to CIE $L^*a^*b^*$ and filtered using a log-Gabor filter. Finally, all extracted features are combined to compute a saliency map.

Motion plays an important role in visual attention, with moving objects attracting the viewer's attention selective behavior in temporal domain and always leads to more attention than other locations in scene [170]. Although several aspects influence the saliency of a video signal, in this work we compute a simplified temporal saliency (TS) of the video signal using a motion estimation algorithm. More specifically, we use the optical flow algorithm (to generate motion vector maps) implemented by Farneback [171], which performs well even when there are luminance changes and the scene has a lot of edges [172, 173]. For illustration, Figure 3.2.3(a) shows sample video frames taken from the CSIQ database [148], Figure 3.2.3(b) shows the computed saliency maps that represent spatial saliency information, and Figure 3.2.3(c) shows the optical flow maps that represent temporal saliency information.

Let $SS(i,j)$ be the value of the spatial saliency map at position (i,j), while $TS(i,j)$ is value of the temporal saliency at the same position. We subdivide the temporal and spatial saliency maps into patches P_k of size 32×32, as shown in Figure 3.2.4. It is worth mentioning that the process of patch-selection is non-overlapping. The amount of spatial and temporal



(a) Distorted Frames (b) Saliency Maps (c) Optical Flow Maps (d) Weighted Maps (b)+(c)

Figure 3.2.3: (a) Example of a distorted frames taken from the CSIQ database; (b) spatial saliency by saliency maps of (a); (c) temporal saliency by optical flow maps of (a); and (d) resulting weighted maps.

saliency information in the k-th patch ($0 \leq k \leq K$) are given by:

$$\begin{aligned} SS_{P_k} &= \sum_{(i,j)\in P_k} SS(i,j), \\ TS_{P_k} &= \sum_{(i,j)\in P_k} TS(i,j). \end{aligned} \tag{3.2.1}$$

The relevance weight of the k-th patch is defined as [5]:

$$W_{P_k} = \alpha \cdot SS_{P_k} + (1-\alpha) \cdot TS_{P_k}, \tag{3.2.2}$$

where α is a constant value that balances the contributions of the spatial and temporal saliency information. The value of α ranges in [0, 1], and in this work, we use $\alpha = 0.4$ [5]. Figure 3.2.3(d) shows the weighted maps obtained by combining spatial and temporal saliency of the frames in Figure 3.2.3(a), with brighter colors corresponding to more important areas and, therefore, higher weights.

Figure 3.2.4 shows the non-overlapping patch-selection process using the spatio-temporal saliency information. The process consists of, first, sorting all K frame patches in a decreasing order of W_{P_k} and, then, choosing the top L most relevant patches. The predicted quality score corresponding to each frame (PQS_f) is obtained by computing a weighted average of

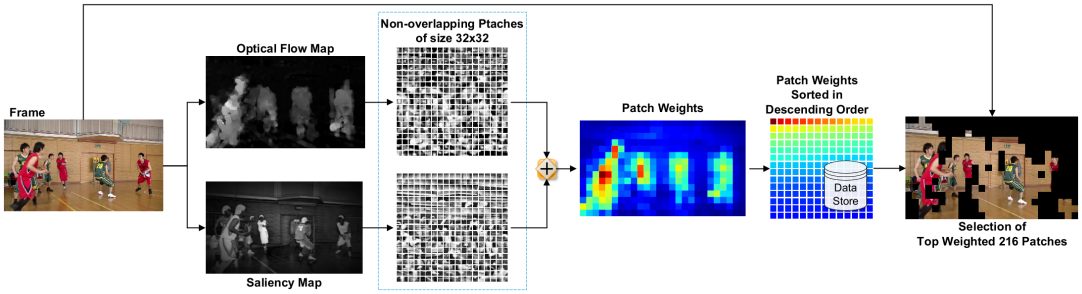


Figure 3.2.4: Optical Flow maps and saliency maps are obtained from input frame. These maps are combined (according to equations 3.2.1 and 3.2.2) to generate weighted maps. Computed weights are sorted in descending order and saved in local directory. Then, based on these weights, top weighted patches are selected and supplied to CNN. Computed weights are also used to predict final quality score for input frame (according to equation 3.2.3).

the predicted quality scores of each patch (PQS_{P_l}) [5], as given by the following equation:

$$PQS_f = \frac{\sum_{l=1}^{L} W_{P_l} \cdot PQS_{P_l}}{\sum_{l=1}^{L} W_{P_l}}, \tag{3.2.3}$$

where L is the total number of selected patches, W_{P_l} is the weight of the l^{th} patch P_l, and PQS_{P_l} is the predicted quality score of the l^{th} patch. Finally, to obtain the predicted quality score (PQS) of the complete video, we compute a simple average of the predicted quality of all video frames (PQS_f).

As mentioned above, there is a single 'ground-truth' for all the frames in a video [174]. One of the contributions of the proposed method is that, instead of using a single MOS for all video frames, we use objective quality scores as target quality scores for each video frame. To choose the most adequate IQA metric (in terms of efficiently interpreting each distorted scene, estimating distinct objective quality scores, and the network converges well on these objective scores), we performed a test, where we used a set of popular NR methods to generate quality scores for each video frame in the CSIQ database [148]. The metrics considered in this test were: no-reference image quality assessment based on spatial and spectral entropies (SSEQ) [48], unsupervised feature learning framework for no-reference image quality assessment (CORNIA) [64], no-Reference image quality assessment in the spatial domain (BRISQUE) [112], and blind image quality assessment from scene statistics (DIIVINE) [114]. Figure 3.2.5 shows the process to generate target quality scores.

For simplification, we downsampled temporally the original videos, generating videos with 2 frames per second (fps). Then, we trained the VSBIQA CNN architecture using a tuple of the input frame (considering all patches) and its corresponding objective quality score. We used 80% of the CSIQ database for training and 20% for test. Videos corresponding to the

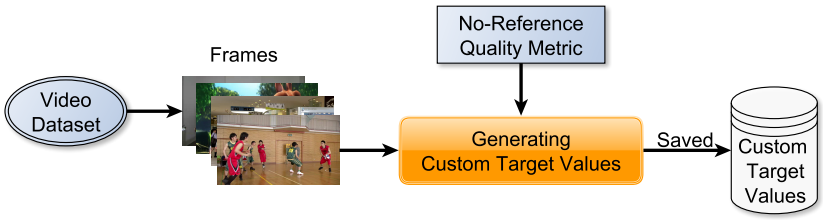


Figure 3.2.5: The process to generate custom target quality scores.

same content (the reference and its corresponding distorted versions) were not present simultaneously in the training and test sets. Table 3.2.1 shows the correlation results obtained for this test that illustrates how well the network is learning to reproduce the objective quality scores for each frame. Notice that DIIVINE has obtained the highest SROCC value, while the highest PLCC value was obtained by the SSEQ. In other words, DIIVINE is the metric whose predictions are closer to the quality as perceived by human viewers, providing the CNN more reliable target quality scores for each frame. Therefore, we chose DIIVINE as the IQA metric to compute the frame objective quality scores to perform further experiments in the proposed method.

Table 3.2.1: SROCC and PLCC values for tests performed for the CSIQ database with 2fps, using quality scores computed with SSEQ, CORNIA, BRISQUE, and DIIVINE.

Correlation	SSEQ	CORNIA	BRISQUE	DIIVINE
PLCC	**0.8760**	0.8392	0.8165	0.8615
SROCC	0.8910	0.8392	0.7688	**0.8975**

3.2.1 Experimental Setup

To train and test the proposed methodology, we have used the Computational and Subjective Image Quality (CSIQ) [148] and the Laboratory for Image and Video Engineering (LIVE) [149] video quality databases.

As performance metrics, we used the SROCC and PLCC. For statistical comparison, we used the following visual quality metrics:

- NR-IQAs: GWH-GLBP [47];
- RR-VQAs: VQM [63]
- FR-VQAs: FREITAS2018 [55];
- NR-VQAs: SACONVA [54], SINGH2019 [56], V-BLINDS [46], and SSDCT [62].

For training and testing, we divided each database into two content-independent subsets, i.e., training and testing subsets. The videos generated from one reference in the testing subset are not present in the training subset and vice-versa. Each reference video and its corresponding distorted versions belong to the same group of scenes. After grouping the videos by content (versions of the same reference), 80% of the groups are randomly selected for training and the remaining 20% are used for testing, and the correlation values for the test set is reported. The process of training was performed using mini-batches, with batches of size equal to 4, i.e., each batch has four frames, where each frame corresponds to the selected patches. We trained the network with 6,000 epochs, using the Mean Square Error (MSE) as the training loss. We used the *Adam* optimizer [175] to set the learning rate. We implemented the proposed method using Chainer[2] framework of Python. The method was trained and tested on 25GB GPU, with a LINUX environment.

3.2.2 Experimental Results

Table 3.2.2: SROCC and PLCC values for tests performed for the CSIQ database with different percentages of patches.

Perc.		AVC	PLR	HEVC	MJPEG	WC	WN	ALL
35%	PLCC	0.8908	0.8660	0.9129	0.8307	0.9239	0.9655	0.9130
	SROCC	0.9199	0.8509	0.9140	0.8579	0.9353	0.9641	0.9255
45%	PLCC	0.8960	0.8673	0.8993	0.8134	0.9244	0.9716	0.8818
	SROCC	0.9229	0.8531	0.9051	0.8351	0.9332	0.9732	0.9219
55%	PLCC	**0.9737**	**0.9738**	**0.9790**	**0.9539**	0.9772	**0.9813**	0.9640
	SROCC	**0.9872**	**0.9768**	**0.9872**	**0.9733**	**0.9732**	**0.9839**	**0.9805**
65%	PLCC	0.9723	0.9695	0.9725	0.9508	**0.9797**	0.9766	**0.9649**
	SROCC	0.9854	0.9646	0.9842	0.9710	0.9751	0.9801	0.9767
75%	PLCC	0.9584	0.9348	0.9645	0.9078	0.9592	0.9718	0.9403
	SROCC	0.9687	0.9522	0.9755	0.9394	0.9548	0.9739	0.9629
85%	PLCC	0.8876	0.8735	0.8877	0.8283	0.9404	0.9646	0.9110
	SROCC	0.9366	0.8565	0.8994	0.8557	0.9445	0.9614	0.9225
95%	PLCC	0.9440	0.9260	0.9554	0.8960	0.9470	0.9647	0.8984
	SROCC	0.9575	0.9400	0.9673	0.9341	0.9397	0.9683	0.9523
100%	PLCC	0.7868	0.8654	0.6820	0.8406	0.8825	0.9700	0.8436
	SROCC	0.8721	0.8739	0.7684	0.7659	0.8936	0.9618	0.8962

As mentioned before, the proposed architecture takes as input a selected number of patches. Our first test is to find the most adequate number of selected patches. For the CSIQ database, we considered 10 different percentages of selected patches: 15%, 25%, 35%, 35%, 55%, 65%, 75%, 85%, 95% and 100%. These percentages resulted into 10 groups of 60, 100, 138, 178, 216, 256, 294, 334, 378, and 390 patches, respectively. For the LIVE database, we also considered 10 different percentages of selected patches: 12%, 22%, 32%, 32%, 52%, 62%,

[2] `https://docs.chainer.org/en/stable/`

Table 3.2.3: SROCC and PLCC values for tests performed for the LIVE database with different percentages of patches.

Perc.		H264	IP	MPEG2	TE	ALL
12%	PLCC	0.9501	0.9124	**0.9600**	0.9115	0.9235
	SROCC	0.9064	0.8987	0.9688	0.9007	0.9376
22%	PLCC	0.9584	0.9499	0.9279	0.9608	0.9458
	SROCC	0.9625	0.9622	0.9339	0.9368	0.9461
32%	PLCC	0.9645	0.9592	0.9396	**0.9637**	**0.9520**
	SROCC	0.9604	0.9525	0.9593	0.9428	0.9548
42%	PLCC	**0.9711**	**0.9706**	0.9215	0.9514	0.9410
	SROCC	0.9784	**0.9818**	0.9597	**0.9492**	**0.9672**
52%	PLCC	0.9597	0.9600	0.8789	0.9381	0.9228
	SROCC	0.9679	0.9638	0.9200	0.9280	0.9451
62%	PLCC	0.9616	0.9627	0.8830	0.9419	0.9243
	SROCC	**0.9706**	0.9699	0.9265	0.9211	0.9463
72%	PLCC	0.9393	0.9471	0.8561	0.9269	0.9034
	SROCC	0.9653	0.9627	0.9100	0.9097	0.9377
82%	PLCC	0.9354	0.8997	0.9389	0.8967	0.9069
	SROCC	0.9070	0.9006	**0.9697**	0.8991	0.9332
92%	PLCC	0.9374	0.9458	0.8729	0.8936	0.9079
	SROCC	0.9321	0.9518	0.9076	0.9003	0.9210
100%	PLCC	0.8786	0.8786	0.8966	0.8626	0.8568
	SROCC	0.8566	0.8302	0.9008	0.8401	0.8563

Table 3.2.4: Comparison of SROCC and PLCC obtained from experiments on CSIQ, and LIVE video quality databases, using target quality scores computed by DIIVINE. For each video in quality databases, frames with 2fps are used, where top weighted patches are selected from each frame.

Database	Distortion	SINGH2019		SACONVA		FREITAS20181		GWH-GLBP		VQM		V-BLINDS		SSDCT		PROPOSED	
		PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC
CSIQ	AVC	0.9490	0.9330	0.939	0.908	0.9419	0.9166	0.8870	0.8630	0.9330	0.8990	0.9410	0.9080	0.9370	0.9120	**0.9737**	**0.9872**
	PLR	0.9210	0.8970	0.8180	0.8030	0.8122	0.7833	0.8030	0.7640	0.7900	0.7710	0.7680	0.7580	0.7800	0.7780	**0.9738**	**0.9768**
	HEVC	0.9410	0.9170	0.9140	0.8870	0.9631	0.9501	0.8410	0.8320	0.9010	0.8470	0.8710	0.8410	0.8810	0.8530	**0.9790**	**0.9872**
	MJPEG	0.9280	0.9010	0.8680	0.8450	0.9066	0.8833	0.7130	0.7010	0.8210	0.7980	0.8930	0.8830	0.8600	0.8550	**0.9539**	**0.9733**
	WC	0.9420	0.9310	0.9100	0.8870	0.9071	0.8833	0.8890	0.8630	0.8900	0.8470	0.9100	0.9120	0.9630	0.8830	**0.9772**	**0.9732**
	WN	0.9530	0.9470	0.9500	0.9320	0.9492	0.9166	0.9010	0.8940	0.8810	0.8700	0.9510	0.9350	0.9360	0.9020	**0.9813**	**0.9839**
	ALL	0.8960	0.8800	0.8620	0.8530	0.8564	0.8688	0.7400	0.7190	0.7970	0.7830	0.8228	0.8069	0.8120	0.8010	**0.9640**	**0.9805**
LIVE	H264	0.9380	0.9190	0.9330	0.9160	0.8877	0.8809	0.7890	0.7650	0.8670	0.8130	0.8840	0.8590	0.8950	0.8720	**0.9597**	**0.9679**
	IP	0.9390	0.8510	0.9270	0.8380	0.8654	0.8602	0.6570	0.6310	0.8560	0.8010	0.8770	0.7820	0.8900	0.8210	**0.9600**	**0.9638**
	MPEG2	**0.9210**	0.9070	0.9170	0.9010	0.8819	0.8809	0.7310	0.7300	**0.9210**	0.8500	0.8950	0.8770	0.9020	0.8920	0.8789	**0.9200**
	TE	0.9190	0.8720	0.9010	0.8810	0.8721	0.8285	0.7640	0.7470	0.8470	0.7920	0.9250	0.8460	0.9280	0.8120	**0.9381**	**0.9280**
	ALL	0.8690	0.8560	0.8650	0.8510	0.8367	0.8246	0.7420	0.7200	0.8020	0.7790	0.8470	0.8100	0.8580	0.8250	**0.9228**	**0.9451**

72%, 82%, 92% and 100%, yielding 38, 69, 100, 132, 163, 194, 225, 256, 288 and 312 patches, respectively. Notice that we have slightly different percentages for the two databases, which are due to the fact that videos in each database have a different spatial resolution.

Tables 3.2.2 and 3.2.3 show the SROCC and PLCC values for CSIQ and LIVE databases, respectively, with bold values representing the highest correlation values for each test case. The group '100%' corresponds to the case where no patch selection is performed and 'ALL' represents the complete set of videos in the database. Notice that, the SROCC and PLCC values do not dramatically change for the different percentages in both CSIQ and LIVE databases. For the CSIQ database the best overall performance ('ALL') is achieved for 55%, while for the LIVE database it is achieved with 42% of the patches. Interestingly, after these maximum val-

ues, the correlations decrease as the percentage of patches increases. For the CSIQ database, we did not obtain a valid correlation value for the first two groups of percentages (15% and 25%).

Table 3.2.4 shows a comparison of the PLCC and SROCC values obtained with the proposed method and the chosen quality metrics, for the CSIQ and LIVE databases. In this test we considered 52-55% of the patches of CSIQ and LIVE databases, with the small differences being due to the differences in spatial resolution. The proposed NR-VQA method obtained the highest SROCC and PLCC values for both databases. It is worth pointing out that since the code for the tested metrics are not public, the reported results corresponded to the ones published in their original works. Moreover, the reported correlations for these metrics are based on processing 100% of the video frames. Therefore, the reduction in spatial and temporal resolutions did not impact the accuracy of the proposed method. In fact, in both databases, the method clearly outperforms the other metrics for all types of distortions and for the 'ALL' case.

Table 3.2.5: PLCC and SROCC values for cross-database validation test, where the proposed model was trained on CSIQ and tested on LIVE.

Correlation	H264	IP	MPEG2	TE	ALL
PLCC	0.8961	0.9191	0.9881	0.8905	0.9122
SROCC	0.6571	0.8956	0.9515	0.9788	0.9369

To test the consistency of the proposed VQA method in the presence of different (unseen) visual content and distortions, we performed a cross-database validation test. Table 3.2.5 shows the results of this test, where our method was trained on the CSIQ database and tested on the LIVE database. In summary, these results show that the proposed NR-VQA method is robust and consistent across different contents. For example, considering the distortions IP, TE, and MPEG2 in LIVE dataset, which is a completely different scenario for a method trained on CSIQ dataset, we can see that the proposed method showed good performance. Specifically for TE and MPEG2 distortions, the method showed even higher performance for SROCC comparatively.

3.3 Conclusions

In this chapter, we have discussed our developed NR image and video quality assessment methods. The proposed NR-IQA method is based on the statistics of a new texture algorithm called the MSLBP. This algorithm extends the capabilities of a previous MLBP algorithm by incorporating both texture and saliency information. Quality is predicted after training a regression model using a random forest algorithm. Experimental results showed that, when

compared with state-of-the art NR-IQA methods, the proposed method has the best performance. Incorporation of saliency information has significantly brought enhanced prediction performances. The MSLBP is general-purpose, and has shown consistent performance for a diversity of visual content and types of distortions.

Second method proposed in this chapter is a novel NR-VQA method, which uses a single CNN model and selects the most perceptually relevant patches using spatial and temporal saliency models. The method does not require subjective quality scores to train the CNN, rather, it uses computed IQA scores as target quality scores for the video frames. Although the method has much smaller cost of data-processing because a small percentage of the total video is used, its accuracy performance is not affected. In fact, the method clearly outperforms other state-of-the-art quality assessment methods. The cross-database test has shown that our method is robust and consistent across different contents and types of distortions. In future, we intend to expand our work using other video quality datasets.

Chapter 4

LF-IQA Methods Based on Two-streams CNN

In this chapter, we discuss two proposed Light Field image quality assessment (LF-IQA) methods that are based on multi-streams CNN architectures, and they do not use reference information. In first LF-IQA method, we use Human Visual System-based multi-stream Convolutional Neural Network (HVS-CNN) which takes into account intense distortion-related characteristics in spatial and angular dimensions. Second LF-IQA method is based on a Deep Neural Network that uses Frequency domain inputs (DNNF-LFIQA). The DNNF-LFIQA method extracts and processes features from the Fourier magnitude spectrum of the LF content, represented as horizontal and vertical EPIs. Inputting the EPIs in the frequency domain allows for a better analysis of (fast) intensity changes in the spatial and angular domain, which is often difficult using the original EPIs.

The key contributions of this work are the following:

- We propose a NR LF-IQA method that uses an HVS-inspired two-stream CNN architecture (HVS-CNN) to learn intense distortion-related characteristics of the LF contents. Up to our knowledge, there is no LF-IQA method that uses a structure that extracts features independently from the SAIs and EPIs, taking into account their dependencies.
- The HVS-CNN method uses a novel approach to generate multiple epipolar-planes images, which we call *MultiEPL*. This method takes advantage of the information in the angular domain of LFIs, generating rich features for quality estimation.
- We propose another LF-IQA method that is based on two-stream CNN architecture, but it employs the horizontal and vertical EPI inputs in frequency domain.
- We analyze the performance of different variants of the proposed methods, by performing simple ablation tests and cross-database evaluation.

4.1 LF-IQA Method Based on HVS-Inspired Two-streams CNN (HVS-CNN)

To implement the proposed LF-IQA method, we chose a two-stream CNN architecture that is based on multiple layers interacting with each other. More specifically, this architecture integrates the information of the two streams by summation and subtraction of the corresponding feature maps. It is worth pointing out that Zhou *et al.* [6] used this two streams architecture to extract features from the left and right views of a 3D representation, which are later combined to identify binocular effects, such as binocular fusion and rivalry. In this work, we adapt this two-stream CNN architecture to learn relevant angular and spatial vision characteristics and dependencies of LFI contents to predict their visual quality. The first stream extracts angular features from EPIs generated from the LFIs, while the second stream extracts the spatial features from mean Canny maps generated from SAIs. The output from both streams is further processed by the interacting layers. Then, the final feature vector is sent to fully connected layers for regression operation that produces a scalar value as quality score.

Figure 4.1.1 depicts the block diagram of the proposed NR LF-IQA method. The method works as follows:

1. Prepare input1: We generate EPIs by the proposed method, named *MultiEPL*, and are converted into Canny maps.
2. Prepare input2: The input2 is computed using mean Canny maps that are generated from SAI Canny maps.
3. Training: Input1 and input2 are fed to the two-stream CNN for training.
4. Regression: After training, a regression is performed on the output feature vectors.
5. Output: Regression operation generates a scalar output that represents Estimated quality score the quality estimate for the corresponding input pairs.

4.1.1 Two Stream Network

The CNN architecture [6] used in this work is an end-to-end interactive CNN, with a multi-layer architecture inspired by the HVS hierarchical structure [176]. This architecture was originally designed to process stereoscopic images and, because of this, it is named StereoQA-Net [1]. It has two streams, namely *stream1* and *stream2*, with each stream having an identical number of convolutional layers, as shown in Figure 4.1.2. The concatenation

[1] `https://github.com/weizhou-geek/Stereoscopic-Image-Quality-Assessment-Network`

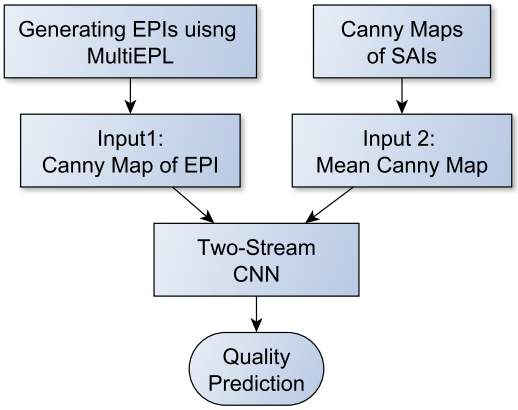


Figure 4.1.1: Block Diagram of the proposed no-reference light field image quality assessment method.

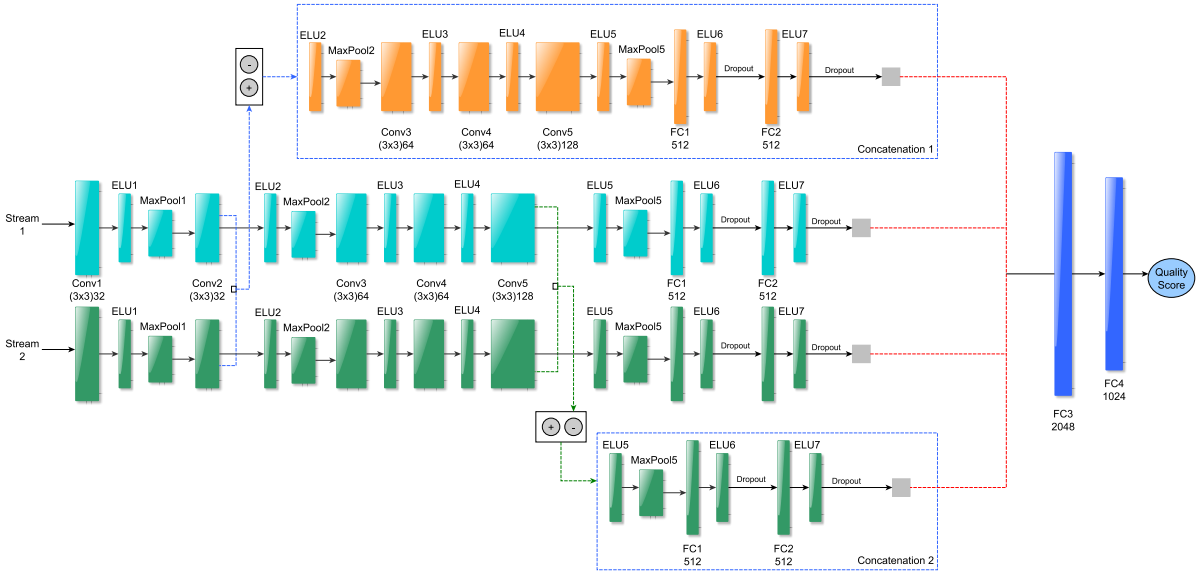


Figure 4.1.2: The architecture of StereoQA-Net model [6].

of these two streams occurs on layers *Conv2*, *Conv5*, and *FC2*. The sum (+) and subtraction (-) symbols in Figure 4.1.2 correspond to the fusion and difference stages, which map the corresponding feature maps coming from *stream1* and *stream2*. Then, the *FC2* layers are concatenated and passed to fully connected structures (*FC3* and *FC4*), which produce the quality estimation of the content.

In this work, we have adapted the StereoQA-Net architecture to predict the quality of LFI contents, using both SAI and EPI formats. Particularly, *stream1* processes spatial information, while *stream2* processes the angular information. Regarding the spatial information, for each LFI, the corresponding SAIs carry spatial information relevant to the LF overall quality. To emphasize the spatial information represented in SAIs, we compute the edge map

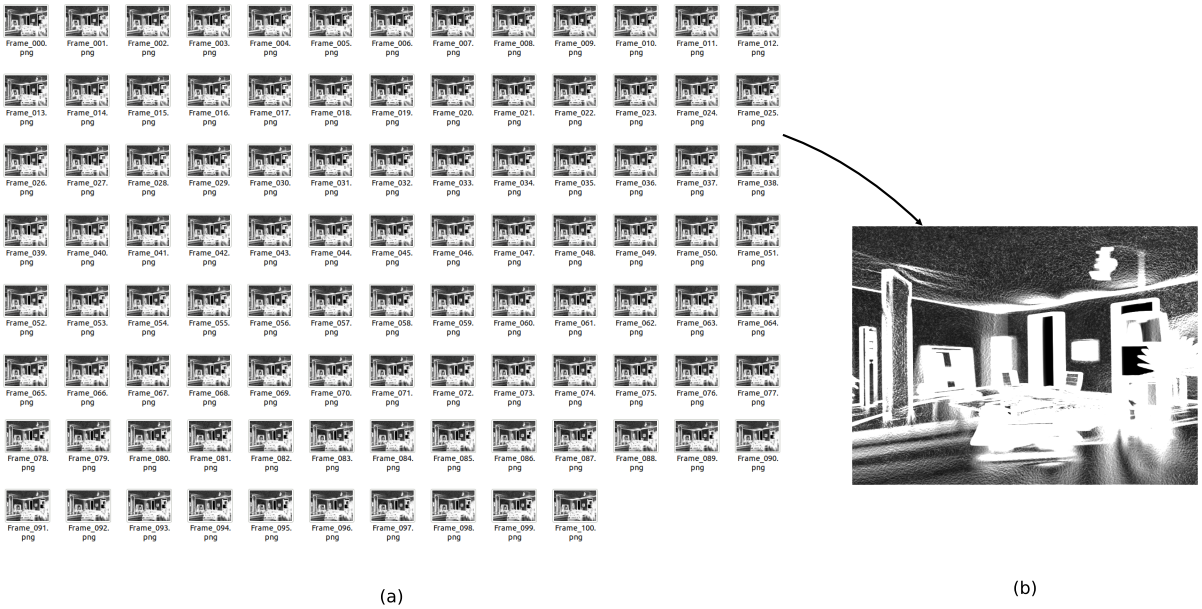


Figure 4.1.3: An example of a mean Canny map of a light field image (ArtGallery2) taken from the MPI dataset [7]: (a) grid of 10 × 10 Canny maps of sub-aperture images and (b) mean Canny map generated from (a).

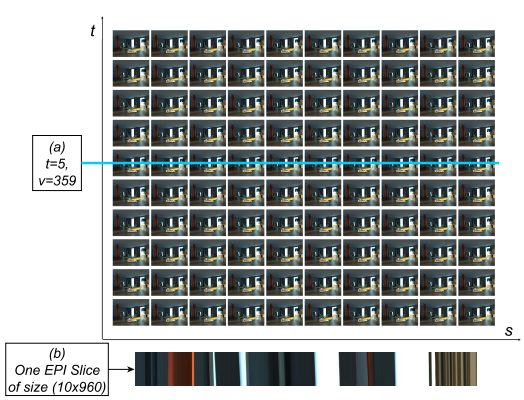
of each SAI using the Canny[2] operator [177], and compute the mean of all SAI Canny edge maps. Figure 4.1.3 shows an example of the mean Canny edge maps that are used as input to *stream1*.

4.1.2 MultiEPL Approach

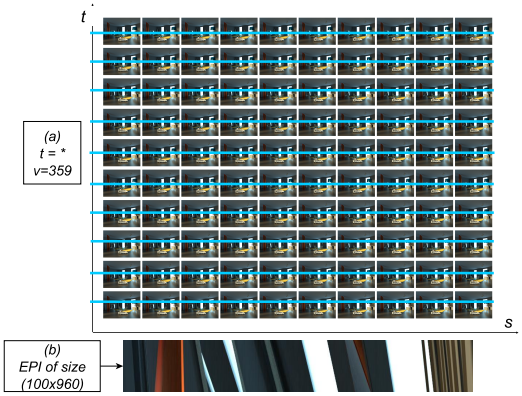
As mentioned before, EPIs can be generated from LFIs. EPIs contain the angular information of LF contents, which are generally able to show more clearly the angular distortions that may affect the overall quality. For a single LFI, we can extract vertical and horizontal (directions) EPIs. As mentioned before, the vertical EPI is obtained by fixing s and u, while the horizontal EPI is obtained by fixing t and v. We can also get a single epipolar line[3] (*SingleEPL*) image by fixing only one row of t [179]. For example, the LFIs available in the MPI dataset [7] have 100 sub-aperture images with a spatial resolution of 960 × 720 pixels, where these dimensions represent the u and v plane of the plenoptic space. By converting 100 SAIs into a 10 × 10 grid and setting $t = 5$ and $v = 359$ (mean = 960), we can get one EPI from one epipolar line, i.e. a *SingleEPL*, as illustrated in Figure 4.4(a). This way, we are able to get 720 EPI slices where every slice has resolution of 10 × 960 pixels.

[2]Canny is ad edge detection algorithm, and it is named after developer of this algorithm i.e., John Canny.

[3]In epipolar geometry, an epipolar line represents a straight line of intersection of the epipolar plane with the image plane [178]. In this work, by epipolar line, we refer to the row of parallax views of the light field image.



(a) Generating *SingleEPL* from a light field image (ArtGallery2 from MPI dataset [7]): (a) 10×10 grid of SAIs, with the coordinates $t = 5$ and $v = 359$ highlighted in blue and (b) EPI slice of resolution 10×960 generated from extracting the highlighted area in (a).



(b) Generating a *MultiEPL* image from a light field image (ArtGallery2 - from MPI dataset [7]): (a) 10×10 grid of SAIs, with coordinates $t = *$ and $v = 359$ highlighted in blue (b) EPI of resolution 100×960 generated highlighted areas in (a).

Figure 4.1.4: Illustration of traditional *SingleEPL* and the proposed *MultiEPL* method to generate EPIs.

In this work, instead of using a single row as an epipolar plane or line, we process all rows of the grid. We name this approach multi-epipolar line (*MultiEPL*). Specifically, we fix $v = 359$ and process all SAI rows ($t = *$). This way, we get EPIs from each row, and then, after horizontally stacking all EPIs, we get final EPI of size 100×960 ($10(10) \times 960$), as illustrated in Figure 4.1.4. Figure 4.1.5 depicts the EPIs and their corresponding Canny edge maps of LFI from the MPI dataset [7]. Next, we obtain the Canny edge maps of these EPIs that will be processed by *stream2*. The outputs of the *stream1* and *stream2* sub-networks are processed at the convolutional layers by performing summation and subtraction operations of the corresponding feature maps. Finally, the extracted features are concatenated at the fully connected layers and a regression is performed to predict the quality of the corresponding input pairs. The code of the proposed LF-IQA is available for download on GitHub[4] under a general public license.

4.1.3 Experimental Setup

To train and test the proposed LF-IQA method, we have used 4 light field image quality datasets MPI, VALID, SMART, and Win5-LID. We have chosen these datasets due to the diversity of their content, the types of LF distortions, and the availability of subjective quality scores. We used the following performance evaluation metrics: SROCC and PLCC. We compared the proposed NR LF-IQA method with the following state-of-the-art LF-IQA methods:

[4] `https://bit.ly/3Da8fB6`

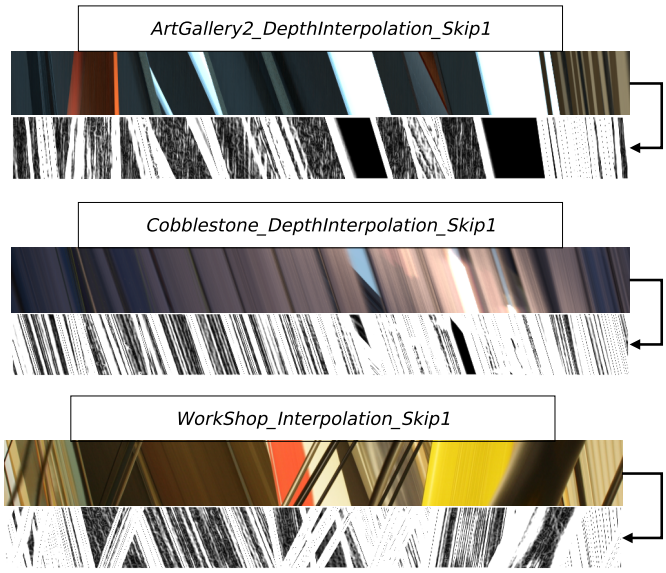


Figure 4.1.5: Example EPIs and their corresponding Canny edge maps for an LFI from the MPI dataset [7].

MDFM [180], LFIQM [75], Fang *et al.* [102], SDFM [107], Meng *et at.* [76], LGF-LFC [181], NR-LFQA [77], LF-QMLI [78], Jiang *et al.* [104], BELIF [182], Tensor-NLFQ [79], Ak *et al.* [80], Shan *et al.* [109], VBLIF [183], ALAS-DADS [83] and Lamichhane *et al.* [84]. We also compared the proposed method with the following 2D image/video quality assessment methods: PSNR-YUV [165], IW-PSNR [184], FI-PSNR [185], MW-PSNR [68], SSIM [11], IW-SSIM [184], UQI [186], VIF [25], MJ3DFR [187], GMSD [53], NIQE [188] and STMAD [189].

4.1.4 Parameter Setup

For training and testing, we divided each dataset into two content-independent training and testing subsets, i.e. distorted images generated from one reference in the test subset are not present in the training subset and vice versa. We define a group of scenes as a set containing the reference LFI and its corresponding distorted versions. Then, 80% of the groups were randomly selected for training and the remaining 20% were used for testing. We only report the correlation values for the test group. It is worth mentioning that we trained the CNN architecture from scratch (instead of using a pre-trained model) with the following parameters: mini-batches of size 128, 6,000 epochs, and Mean Square Error (MSE) as the training loss. Furthermore, we used the SGD [118] optimizer to minimize the loss function with a learning rate of 0.0001. We implemented the proposed method using the Keras [190] library of Python. The method was trained and tested on 25GB GPU, in a LINUX environment. Table 4.1.1 shows a summary of parameters used for training and testing the CNN architecture.

Table 4.1.1: CNN Parameter Setup.

Parameter	Value
Training Set	80%
Test Set	20%
Batch Size	128
Epochs	6,000
Training Loss	MSE
Optimizer	SGD
Learning Rate	0.0001

4.1.5 Experimental Results

To determine which representation approach, *SingleEPL* or *MultiEPL*, performs best for the quality assessment of LFI, we first conduct a simple test on a single MPI data set. To conduct this test, we prepared two formats of input1 (input1 and input2 are shown in Figure 4.1.1). The first format of input1 is generated using the *MultiEPL*, while the second format is obtained using the standard approach *SingleEPL*. The input2 is generated as described before. Then, we trained the CNN architecture for these two formats of input1, using 80% of the MPI dataset for training and 20% for testing. Then, we compute the correlations between the generated predicted quality scores and the subjective scores of the test set.

Table 4.1.2: SROCC and PLCC values obtained for the MPI dataset, using *MultiEPL* and *SingleEPL* approaches.

Dataset	Distortion	MultiEPL		*SingleEPL*	
		SROCC	PLCC	SROCC	PLCC
MPI	QD	**0.8956**	**0.9157**	0.0181	0.0901
	Gaussian	0.9393	0.9832	**0.9999**	**1.0000**
	HEVC	**0.9571**	**0.9199**	0.4285	0.5426
	OPT	**0.9767**	**0.9079**	0.5058	0.5188
	Linear	**0.9999**	**0.9952**	0.9181	0.8904
	NN	**0.9821**	**0.9704**	0.7676	0.6768
	ALL	**0.9411**	**0.9404**	0.6103	0.5835

The SROCC and PLCC values for the MPI dataset are reported in Table 4.1.2, where the rows correspond to the different distortion types in this dataset and the row marked as 'All' corresponds to the results obtained for the complete dataset. The bold values represent the highest correlations for each row (distortion). Notice that for the overall ('ALL') case, the *MultiEPL* approach has obtained the highest correlation values. Looking closely at each distortion type, the *MultiEPL* performed best for most distortions, with the exception of the 'Gaussian' distortion. Henceforth, in this work we use the *MultiEPL* approach to generate the input images for *stream2* in our LF-IQA method.

Next, we performed tests on the VALID, SMART, and Win-LID LFI quality datasets, and the results are shown in Table 4.1.3. Again, the rows in this table show the results for each

Table 4.1.3: The SROCC and PLCC values for VALID, SMART, MPI, and Win5-LID datasets.

Dataset	Distortion	PROPOSED	
		SROCC	PLCC
MPI	QD	0.8956	0.9157
	Gaussian	0.9393	0.9832
	HEVC	0.9571	0.9199
	OPT	0.9767	0.9079
	Linear	0.9999	0.9952
	NN	0.9821	0.9704
	ALL	0.9411	**0.9404**
VALID	10bit_HEVC	0.9680	0.9607
	10bit_P3	0.9751	0.8112
	10bit_P5	0.9066	0.8304
	10bit_VP9	0.9461	0.9529
	8bit_HEVC	0.9043	0.9659
	8bit_VP9	0.9450	0.9106
	ALL	0.9410	0.9388
SMART	HEVC	0.9166	0.9857
	JPEG	0.9428	0.9338
	JPEG2000	0.9758	0.9834
	SSDC	0.9181	0.8968
	ALL	0.9364	0.9294
Win-5LID	HEVC	0.9571	0.9054
	JPEG2000	0.9351	0.9560
	LN	0.9868	0.8938
	NN	0.9423	0.9355
	EPICNN	0.9965	0.945
	ALL	**0.9469**	0.9361

dataset and for each distortion, with the 'All' row corresponding to the results obtained for the complete dataset. Notice that the proposed method performs very well in all datasets with SROCC values over 0.93 and PLCC values over 0.92. Across the different distortions, the proposed method also performed very well, with only a few distortions showing slightly lower values (e.g. 10bitP5 of VALID, SSDC of SMART).

Figure 4.1.6 shows the scatter plots of the subjective quality scores versus predicted quality scores obtained for the MPI, SMART, VALID and Win5-LID LFI quality datasets. It is worth mentioning that the MOS ranges for each dataset may be different since different experimental methodologies were used to collect the quality scores. We decided not to normalize the MOS values since a previous study demonstrated that normalizing subjective scores into standard values does not significantly improve the quality predictions [74]. Even though no normalization was performed, the lines in the graphs show good fitting results, indicating that the proposed LF-IQA method is able to predict the quality of LF contents accurately.

Table 4.1.4 illustrates the comparison of the results with other state-of-the-art LF-IQA methods. In this table, the NR and FR LF-IQA methods are classified into three categories, taking into consideration the models used to map the pooled features into quality estimates.

Table 4.1.4: SROCC and PLCC values obtained for state-of-the-art LF-IQA methods tested on VALID, SMART, MPI, and Win5-LID datasets.

Category	Type	Methods	Year	MPI		VALID		SMART		Win5-LID	
				SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Based on Pre-defined Functions	FR	UQI	2002	0.7400	0.8460	0.9310	0.9550	0.6480	0.7980	0.8252	0.8764
	FR	SSIM	2004	0.9120	0.9320	0.9500	0.9640	0.7550	0.8010	0.6812	0.7880
	FR	VIF	2006	0.8600	0.8960	0.9620	0.9790	0.7260	0.8370	0.9347	0.9555
	FR	NIQE	2009	0.5821	0.5122	0.6211	0.6544	0.5214	0.5426	0.4892	0.5002
	FR	STMAD	2011	0.8650	0.8940	0.7940	0.8020	0.6640	0.8010	0.8489	0.9074
	FR	IW-SSIM	2011	0.9320	0.9440	0.9650	0.9780	0.8060	0.8850	0.8212	0.8736
	FR	IW-PSNR	2011	0.9300	0.9160	0.9470	0.9670	0.7840	0.8520	0.8842	0.9022
	FR	MJ3DFR	2013	0.8720	0.9300	0.9560	0.9700	0.8160	0.8480	0.8836	0.8998
	FR	GMSD	2014	0.7358	0.7410	0.6821	0.6948	0.7264	0.8000	0.4352	0.5041
	FR	FI-PSNR	2014	0.8740	0.8510	0.7060	0.7060	0.7730	0.8320	0.6951	0.7419
	FR	PSNR-YUV	2014	0.9342	0.9452	0.9230	0.9310	0.9102	0.9211	0.9007	0.9215
	FR	MW-PSNR	2016	0.7251	0.7698	0.6869	0.6904	0.5281	0.5869	0.7582	0.7758
	FR	MDFM	2018	0.8346	0.8123	0.712	0.7198	0.7535	0.7683	0.8157	0.8591
	FR	Fang *et al.*	2018	0.8065	0.7942	-	-	-	-	-	-
	RR	LFIQM	2019	0.6815	0.7013	0.3934	0.5001	0.4503	0.4763	0.4503	0.4763
	FR	SDFM	2020	0.8435	0.8423	0.824	0.8542	0.7514	0.7941	0.6742	0.7142
	FR	Meng *et at.*	2020			**0.9579**	**0.9762**	-		-	-
	FR	LGF-LFC	2020	0.8543	0.8476	-	-	0.8246	0.8276	-	-
SVR-based	FR	Jiang *et al.*	2018	-	0.8954	-	-	-	-	-	-
	NR	BELIF	2019	0.8854	0.9096	0.8863	0.895	0.8367	0.8833	0.8719	0.8910
	NR	NR-LFQA	2019	0.9119	0.9155	0.9233	0.9316	0.9033	0.9231	0.9032	0.9206
	NR	LF-QMLI	2019	-	-	0.9286	0.9683	-	-	0.8802	0.9038
	NR	Shan *et al.*	2019	-	-	-	-	0.8917	0.9106	-	-
	NR	Tensor-NLFQ	2019	0.9101	0.9225	0.8702	0.9028	0.8702	0.9028	0.9101	0.9217
	NR	Ak *et al.*	2020	0.8942	0.9005	-	-	-	-	-	-
	NR	VBLIF	2020	0.9015	0.9158	-	-	-	-	0.9009	0.9232
CNN-based	FR	Lamichhane *et al.*	2021	-	-	-	-	0.8900	0.9300	-	-
	NR	ALAS-DADS	2021	-	-	-	-	0.8540	**0.9344**	-	-
	NR	**Proposed**	2021	**0.9411**	**0.9404**	0.941	0.9388	**0.9364**	0.9294	**0.9469**	**0.9361**

The categories of the mapping models are: (1) pre-defined functions, (2) SVR algorithms, or (3) CNN approaches. Notice that, for simplicity, only the overall performance ('ALL') correlation values are reported for each dataset. Also, since the authors of these LF-IQA methods did not publish their results for all four datasets, the results matrix is incomplete. Notice that the proposed method has the highest correlation values among all LF-IQA methods for three out of the four datasets. For the dataset SMART, the proposed method obtained the highest SROCC, while the method ALAS-DADS obtained the highest PLCC value. The method proposed by Meng *et at.* [76] (a full-reference IQA method based on a pre-defined mapping function) has the best results for the VALID dataset, while the proposed method is the second best performing method.

To test the consistency of the proposed LF-IQA method in the presence of different (unseen) visual content and distortions, we performed a cross-database test. In this test, the proposed method is trained on the MPI dataset and tested on the Win5-LID dataset. Table 4.1.5 shows the SROCC and PLCC results for this test. Notice that, although this test is more challenging to the method, the correlation values obtained are higher than the correlation values obtained by other LF-IQA methods in Table 4.1.4. This results shows the robustness of the method, considering that the other methods were trained and tested on Win5-LID.

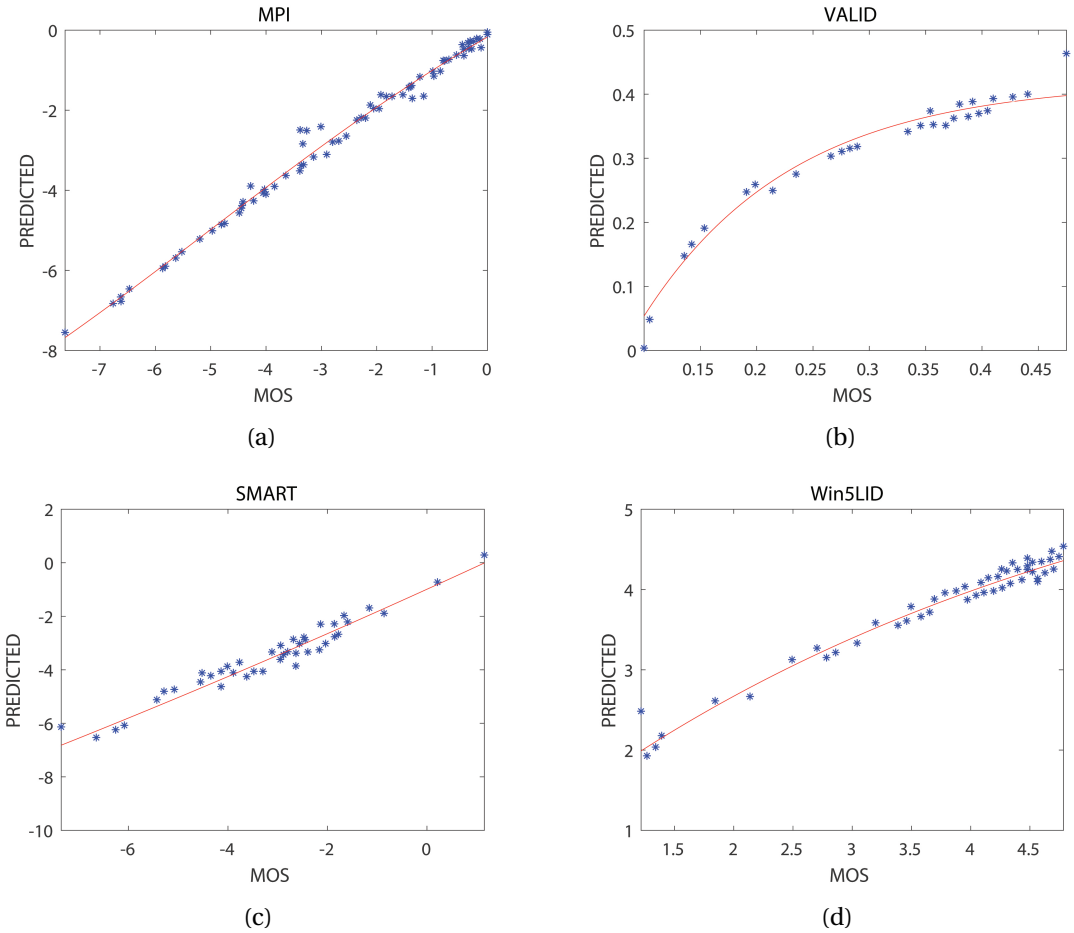


Figure 4.1.6: Scatter plots of subjective quality scores versus predicted quality scores. (a) MPI, (b) VALID, (c) SMART, and (d) Win5-LID.

Table 4.1.5: PLCC and SROCC values for the cross-database test, where the proposed model is trained on MPI and tested on Win5-LID dataset.

Dataset	Distortion	PROPOSED	
		SROCC	PLCC
Win5-LID	EPICNN	0.9	0.9101
	HEVC	0.9333	0.9365
	JPEG2000	0.9552	0.8257
	LN	0.9049	0.9333
	NN	0.9746	0.8412
	ALL	0.9490	0.9347

To demonstrate the effectiveness of the proposed method, we also performed an ablation test. In this test, we split the CNN architecture into two independent streams, where all the concatenation layers are frozen. In other words, we implemented the proposed LF-IQA method using only the *stream1* channel and another one using only the *stream2* channel. The input samples for each stream are kept unchanged, i.e., *stream1* processes the mean

Canny edge map of the SAIs, while *stream2* processes the Canny edge maps of the EPIs. Table 4.1.6 shows the SROCC and PLCC values obtained for these methods based on the individual channels of the CNN architecture. Notice that correlation values obtained with only spatial information (*stream2*) are higher than what was obtained with only angular information (*stream1*). For the overall case 'ALL', both streams achieved much lower correlations, when compared with results obtained for the complete architecture shown in Table 4.1.4. Therefore, the information provided by each channel complements each other to provide accurate LF quality prediction.

Table 4.1.6: Ablation Test Results: SROCC and PLCC values for MPI dataset, separated according the distortion types, using *stream1* and *stream2* of StereoQA-Net without concatenation layers.

Dataset	Distortion	*stream1*		*stream2*	
		SROCC	PLCC	SROCC	PLCC
MPI	QD	0.0934	0.0043	0.3241	0.5099
	Gaussian	0.8545	0.8524	0.9272	0.9562
	HEVC	0.8285	0.8220	0.9428	0.9169
	OPT	0.2694	0.4144	0.7929	0.7869
	Linear	0.8181	0.8035	0.9636	0.9606
	NN	0.4892	0.5176	0.8571	0.8321
	ALL	0.5284	0.5284	0.7190	0.7372

4.1.6 Findings and Practical Application

Table 4.1.7: Findings, advantages and disadvantages of the proposed method.

Property	Advantage	Disadvantage
Pre-processing	Useful Edge Information	Dependent on third-party algorithm for edge detection.
MultiEPL Method	Provides valuable angular information	Dependent on both horizontal and vertical sub-views.
Interactive Multi-streams	Inspired by HVS to process multi-view stimuli.	Dependent on multi-views.
	Able to extract rich distortion-related features from spatial and angular information.	
	Requires no reference information.	
	Require small number of input samples	

In summary, the proposed LF-IQA method is able to capture the most relevant features of LF content, predicting quality with accuracy and robustness. Also, the method requires a small number of input samples, which reduces the computational complexity of the method. However, the proposed method has the following limitations. First, our statistical analysis of

the performance of the *MultiEPL* method is limited, mainly because there are few LF datasets that contain both horizontal and vertical views. Second, we have not evaluated the performance of the proposed LF-IQA method for a large variety of compression distortions, such as JPEG Pleno [191]. This is due to the fact that there is lack of LFI quality datasets that have a large variety of compression distortions. In our future work, we plan to create an LFI quality dataset, which has both horizontal and vertical views and a diversity of distortions generated by more advanced compression techniques, including JPEG Pleno. We also plan to expand the CNN architecture, so that it takes into account the salient areas of LF images. Additionally, the proposed method is dependent on a third-party algorithm for generating edge maps, which might hinder good performance in case of non-availability of the library. Table 4.1.7 describes a summary of outcomes of the proposed method.

The proposed method has exclusive practical applications. Due to the fact that existing LF-IQA datasets do not contain a large amount of distorted input samples, CNN-based quality assessment methods might not perform well for quality prediction. But the proposed method has the ability to perform good, and predict the quality in accordance with human judgments, even with a small number of input samples. Moreover, the existing datasets have subjective quality scores obtained from different types of subjective quality experiments, and therefore, contain non-similar subjective quality scores. The proposed method can perform well without normalizing the subjective scores for all datasets providing a useful application for quality assessment. Last but not least, the method is reference-free. The proposed method can perform well even in the absence of reference information, making the application more feasible.

4.2 LF-IQA Method Using Frequency Domain Inputs (DNNF-LFIQA)

Figure 4.2.1 shows the block-diagram of the DNNF-LFIQA method. Notice that the method has two streams, each composed of identical blocks of convolutional neural networks (CNN). The outputs of the first (stream1) and second (stream2) streams are combined using two fusion blocks that generate a single feature vector. Finally, this fused feature vector is fed to a regression block that produces a quality estimate. Next, we describe each of the stages in Figure 4.2.1.

Figure 4.2.2 shows a description of the CNN blocks of the proposed method, which contains 7 high-level feature extraction stages. Stages 1, 2, and 5 are identical, containing a 2D convolution layer with 32 output filters and a 3×3 kernel, an ELU activation [117] layer, and a 2D max-pooling layer with a 2×2 pool and a 2×2 stride. Stages 3 and 4 are also identical, containing a 2D convolution with 64 output filters and a 3×3 kernel, and an ELU activation layer.

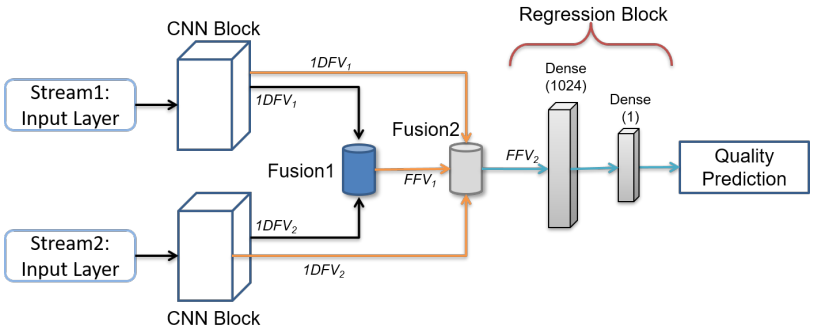


Figure 4.2.1: Block Diagram of the proposed DNNF-LFIQA method.

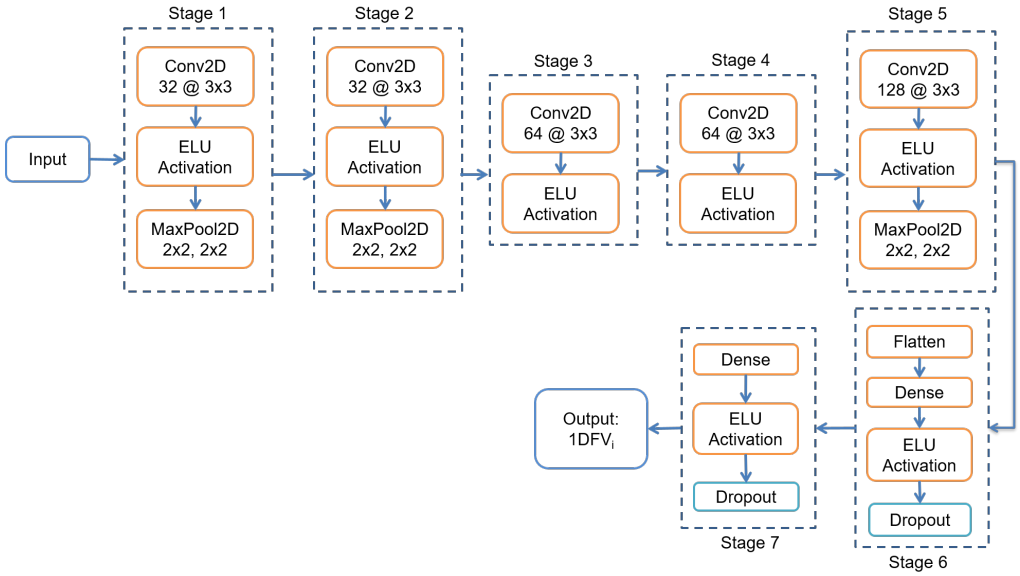


Figure 4.2.2: Illustration on CNN Block in the proposed DNNF-LFIQA method.

Stages 6 and 7 represent fully connected layers. Stage 6 has a Flatten layer, a Dense layer, and a Dropout layer, while stage 7 has a Dense layer, an ELU layer, and a Dropout layer. Notice that, the Dropout layers are added to prevent overfitting. The output of the CNN block is a one-dimensional (1D) feature vector ($1DFV_i$). In this notation, the i index corresponds to the corresponding stream ($i = 1,2$) in the proposed method (see Figure 4.2.1).

To concatenate the outputs of stream1 and stream2 in the proposed DNNF-LFIQA method, we have created two fusion blocks (Fusion1 and Fusion2). Figure 4.2.3 illustrates the architecture of the proposed fusion blocks. As shown in this figure, Fusion1 block is composed of 3 ELU activation layers, 1 Flatten layer, 2 Dense layers with 512 output features, and 2 Dropout layers. Fusion1 block generates as output a 1D feature vector FFV_1, which is fed to the Fusion2 block. The Fusion2 block concatenates the outputs FFV_1 (obtained from

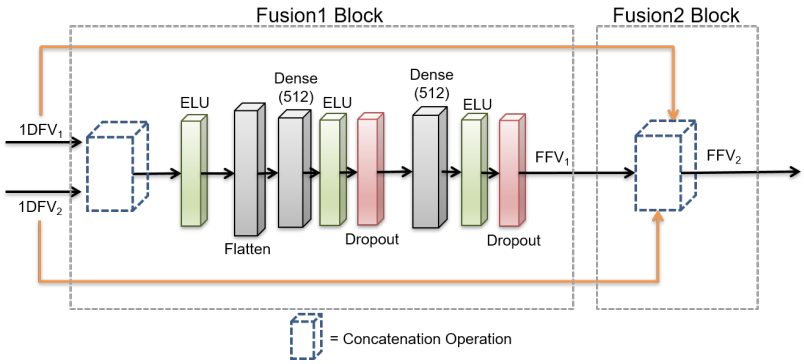


Figure 4.2.3: Illustration on Fusion Blocks in DNNF-LFIQA Method.

Fusion1 block), $1DFV_1$ and $1DFV_2$ (obtained from stream1 and stream2 respectively), producing a final feature vector FFV_2 at the end.

As shown in Figure 4.2.1, the output FFV_2 (obtained from the Fusion2 block) is fed to a regression block, which is composed of 2 Dense layers. The first Dense layer generates a feature vector of size 1024, while the last Dense layer produces a scalar number that corresponds to the estimated perceptual quality score of the test LFI, input as horizontal and vertical EPIs in the frequency domain.

As mentioned before, stream1 takes as input the horizontal EPI, while stream2 takes as input the vertical EPI. But, instead of inputting the EPIs in the spatial domain, we input the EPIs in the frequency domain. More specifically, we first convert the EPIs to the RGB format and obtain their grayscale representation. Then, we compute Fourier transform of these grayscale EPIs and compute the magnitude of the Fourier spectrum. In summary, the Fourier Magnitude spectrum of the horizontal and vertical EPIs are used as inputs to stream1 and stream2, respectively, of the proposed model depicted in Figure 4.2.1.

4.2.1 Experimental Setup

To train and test the proposed DNNF-LFIQA method, we have used 3 light field image quality datasets LFDD, VALID, Win5-LID. We used SROCC and PLCC as performance evaluation methods. We compared the proposed NR LF-IQA method with the following state-of-art LF-IQA methods: SDFM [107], LFIQM [75], Tensor-NLFQ [79], GELFIQE [86], DELFIQE [85], and ALAS-DADS [83]. We also compared the method with the following 2D-FR IQA methods [72, 74]: UQI, VIF, GMSD, NIQE, SSIM, IW-SSIM, IW-PSNR, FI-PSNR, MW-PSNR, MJ3DFR, PSNR-YUV and STMAD.

For training and testing, we divided each dataset into three content-independent training, validation, and test subsets. In this division, test (possibly distorted) images generated from one reference can only be in one of the subsets, i.e., if images corresponding to a spe-

cific reference content are in the test subset, they are not present in the training and validation subsets and vice-versa. More specifically, we define a group of scenes as a group containing the reference LFI and its corresponding distorted versions. Then, 80% of the groups are randomly selected for training, 10% for validation, and the remaining 10% are used for testing. We report the correlation values only for the test subset. We train the DNNF-LFIQA method using mini-batches of size 128, 2,000 epochs, and Mean Square Error (MSE) as the training loss. Also, we used the Stochastic Gradient Descent (SGD) optimizer [118] with a learning rate 0.0001 to minimize the loss function. We implemented the proposed method using Keras [190] library of Python. The method was trained and tested on 25GB GPU, with a LINUX environment. The code of the proposed LF-IQA method is available for download on GitHub [5], under the general public license.

4.2.2 Experimental Results

Figure 4.2.4 displays the train and validation loss curves obtained for the proposed DNNF-LFIQA method, when trained and tested on 3 different LF-IQA test datasets. For the dataset Win5-LID, the training loss decreases continuously until the last epoch, but we see some fluctuation in the validation loss. For dataset VALID, training and validation losses continuously decrease up to the last epoch, without any fluctuation. For datasets VALID and Win5-LID, the proposed method achieved training and validation loss values smaller than one. On the other hand, the dataset LFDD shows a different behaviour, with training loss values smaller than one, but with higher validation loss values.

Table 4.2.1 shows the correlation values obtained for the VALID, Win5-LID, and LFDD LFI quality datasets. The rows in this table show the results for each dataset and for each distortion, with the 'All' row corresponding to the results obtained for the complete datasets. Notice that the proposed method performs very well for the (complete) VALID dataset with a SROCC value of 0.9783 and a PLCC of 0.9883. For the (complete) win5-LID dataset, the proposed method achieves an SROCC value of 0.9357 and a PLCC value of 0.9640. Finally, for the (complete) LFDD dataset, the method obtained lower correlation values, with a SROCC value of 0.7810 and a PLCC value of 0.7332. In this dataset, 7 distortions (Pincushion, Unsharp Mask, AV1, JPEG2000, VP9, x264 and x265) show lower correlation values, while the other 5 distortions have high correlation values. The dataset LFDD has diverse and complex contents (foreground and background have the same range of pixel values), which is probably the reason why the proposed method does not perform well in terms of correlation with the subjective quality scores provided with this dataset.

[5] https://bit.ly/36G6HDy

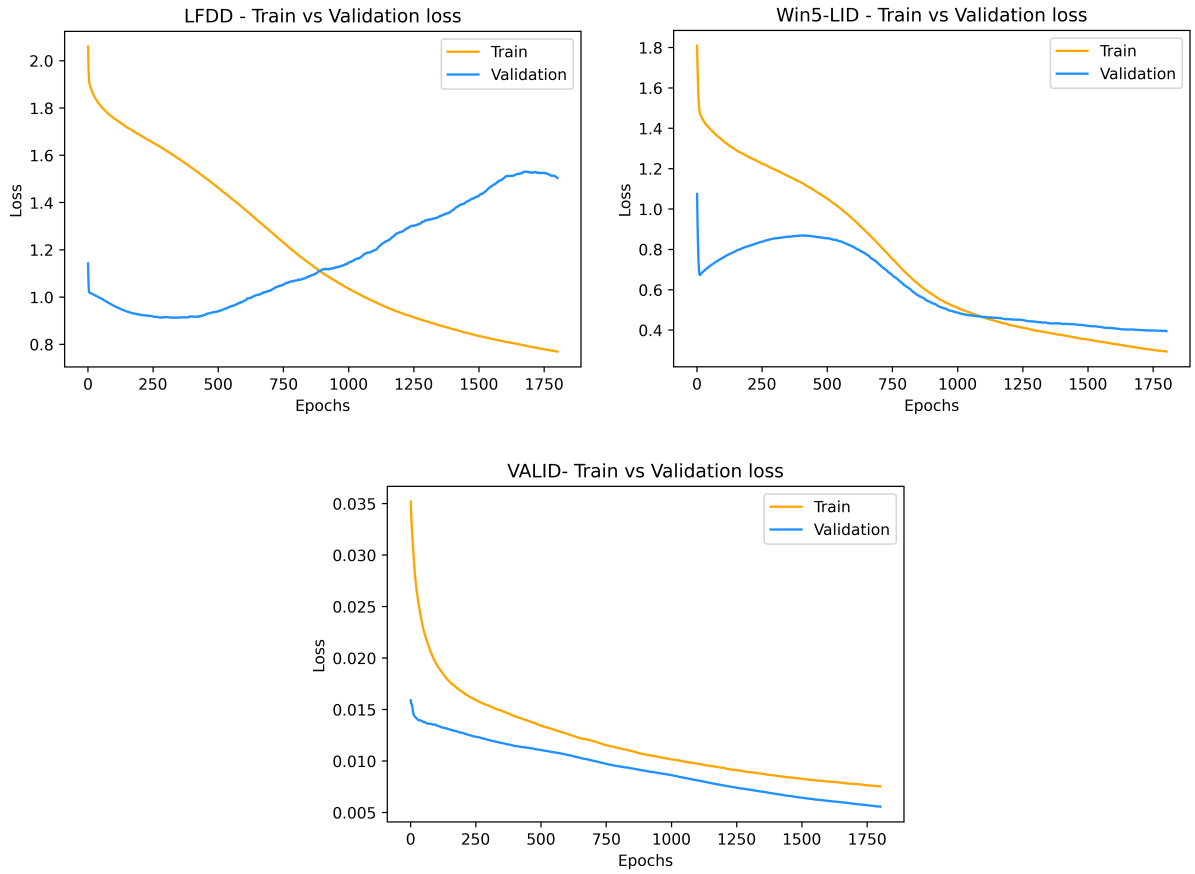


Figure 4.2.4: Train vs Validation Loss of the proposed DNNF-LFIQA method on 3 LF-IQA test datasets.

Table 4.2.1: The SROCC and PLCC values for VALID, LFDD, and Win5-LID datasets.

LFDD			**VALID**			**Win-5LID**		
Distortion	SROCC	PLCC	Distortion	SROCC	PLCC	Distortion	SROCC	PLCC
AV1	0.7437	0.7363	10bit_HEVC	0.9716	0.9888	EPICNN	0.9200	0.9800
BAR	0.8074	0.8601	10bit_P3	0.9716	0.9757	HEVC	0.9383	0.9793
BPG	0.8292	0.8397	10bit_P5	0.9884	0.9796	JPEG2000	0.9291	0.9773
Gaussian	0.9710	0.8322	10bit_VP9	0.9816	0.9784	LN	0.8764	0.9286
JPEG2000	0.7527	0.7113	8bit_HEVC	0.9217	0.9884	NN	0.8309	0.9223
JPEG	0.9274	0.8150	8bit_HEVC	0.9716	0.9790	QD	0.9200	0.9800
Pincushion	0.7500	0.6014	8bit_VP9	0.9800	0.9778	-	-	-
Impulse	0.9165	0.7911	-	-	-	-	-	-
Unsharp Mask	0.6219	0.7445	-	-	-	-	-	-
VP9	0.6764	0.7118	-	-	-	-	-	-
x264	0.6636	0.6108	-	-	-	-	-	-
x265	0.7618	0.5447	-	-	-	-	-	-
ALL	0.7810	0.7332	ALL	0.9783	0.9883	ALL	0.9357	0.9640

Table 4.2.2 depicts the comparison results, containing the correlation values obtained with the proposed method and with state-of-the-art LFI-IQA methods. In this table, we grouped the NR and FR LF-IQA methods into three categories, taking into consideration the

models used to map the pooled features into quality estimates. The categories include methods that use (1) a pre-defined function, (2) an SVR algorithm, or (3) a CNN-based approach. For simplicity, only the overall performance ('ALL') correlation values are reported for each dataset. Also, since the authors of some of the LF-IQA methods did not publish their results for all 3 datasets, our comparison matrix is incomplete. Notice that, for all datasets, the proposed method achieves the highest correlation values among all LF-IQA methods. For the LFDD dataset, although the proposed method obtained lower correlation values than what was obtained for the other datasets, they are much higher than the values obtained by other methods.

Table 4.2.2: SROCC and PLCC values obtained for state-of-the-art LF-IQA methods tested on LFDD, VALID, and Win5-LID datasets.

Category	Type	Methods	Year	LFDD		VALID		Win5-LID	
				SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Pre-defined Functions	FR	UQI	2002	0.4673	0.3486	0.9310	0.9550	0.8252	0.8764
	FR	SSIM	2004	0.4488	0.2457	0.9500	0.9640	0.6812	0.7880
	FR	VIF	2006	0.4588	0.4026	0.9620	0.9790	0.9347	0.9555
	FR	NIQE	2009	0.5235	0.4138	0.6211	0.6544	0.4892	0.5002
	FR	STMAD	2011	0.2054	0.2005	0.7940	0.8020	0.8489	0.9074
	FR	IW-SSIM	2011	0.4432	0.2594	0.9650	0.9780	0.8212	0.8736
	FR	IW-PSNR	2011	0.4184	0.3060	0.9470	0.9670	0.8842	0.9022
	FR	MJ3DFR	2013	0.4235	0.3182	0.9560	0.9700	0.8836	0.8998
	FR	GMSD	2014	0.4384	0.4000	0.6821	0.6948	0.4352	0.5041
	FR	FI-PSNR	2014	0.2415	0.1645	0.7060	0.7060	0.6951	0.7419
	FR	PSNR-YUV	2014	0.4325	0.4124	0.9230	0.9310	0.9007	0.9215
	FR	MW-PSNR	2016	0.4021	0.3842	0.6869	0.6904	0.7582	0.7758
	RR	LFIQM [75]	2019	0.1245	0.1041	0.3934	0.5001	0.4503	0.4763
	FR	SDFM [107]	2020	-	-	0.8240	0.8542	0.6742	0.7142
SVR-based	NR	Tensor-NLFQ [79]	2019	0.5134	0.4124	0.8702	0.9028	0.9101	0.9217
CNN-based	NR	GELFIQE [86]	2021	-	-	0.9442	0.9753	-	-
	NR	DELFIQE [86]	2021	-	-	0.9260	0.9749	-	-
	NR	ALAS-DADS [83]	2021	-	-	-	-	0.9260	0.9257
	NR	**Proposed**	2022	**0.7810**	**0.7332**	**0.9783**	**0.9883**	**0.9357**	**0.9640**

To test the robustness of the proposed NR LF-IQA method in the presence of unseen contents and distortions, we performed a cross-database evaluation. To perform this test, we trained the proposed model on one dataset and tested on a different one. Given the good results obtained for the VALID dataset, we used this dataset for training and used the LFDD dataset for testing. As mentioned earlier, LFDD is a challenging dataset, with the results obtained for this dataset for all tested metrics being much lower than what was obtained for the other 2 datasets (see Table 4.2.2). Table 4.2.3 shows the SROCC and PLCC values for this cross-database evaluation. These results show that the proposed method is robust and consistent across different contents.

The dataset VALID has less complex contents, with a wide range of compression-related distortions. The subjective quality scores for this dataset were obtained using an interactive approach, where the subjects were allowed to change focus point in test content. Previ-

Table 4.2.3: Summary of cross-database evaluation results (SROCC and PLCC) for different train–test dataset combination.

Training Dataset	Testing Dataset	SROCC	PLCC
VALID	LFDD	0.8154	0.7954

ous study [87] shows that the interactive approach can provide better quality of experience. Therefore, the trained model of VALID dataset achieved better correlation values on LFDD dataset.

To demonstrate the effectiveness of the proposed approach, we also performed a simplified ablation test. In this test, we performed 4 experiments (Exp1, Exp2, Exp3 and Exp4) using the Win5-LID dataset. In Exp1, we removed stream2, Fusion1, and Fusion2 blocks and trained the network only with stream1 and the regression block. Exp2 is similar to Exp1, except that we removed stream1 and trained the network with only stream2. In Exp3, we removed the stages 2, 4, and 7 of the CNN blocks of both streams and kept the rest of the network. In Exp4, we used as inputs the horizontal and vertical EPIs in the spatial domain (instead of the frequency domain) and kept the model of the method unchanged. Table 4.2.4 shows the SROCC and PLCC values obtained for these 4 experiments and for the complete proposed model. Note that the correlation values for Exp1, Exp2, Exp3, and Exp4 are significantly lower than the values obtained for the complete model with frequency-domain inputs. In other words, the complete model provides a better performance, in terms of correlation values, than the other variants of the model.

Table 4.2.4: Comparison of proposed model (combination) with 4 variants of the model. Training/Test is performed on the Win5-LID dataset.

Dataset	Experiment	SROCC	PLCC
Win5-LID	Exp1: Stream1 + Regression Block of Figure 4.2.1	0.5180	0.5733
	Exp2: Stream2 + Regression Block of Figure 4.2.1	0.5116	0.5028
	Exp3: CNN Blocks = 1, 3, 5, and 6 stages	0.7861	0.7990
	Exp4: Model in Figure 4.2.2 with EPI inputs (spatial and angular domain)	0.6103	0.5835
	DNNF-LFIQA: Model in Figure 4.2.2 with Frequency domain EPIs	**0.9357**	**0.9640**

Table 4.2.5 presents the time required to train and test/run the proposed DNNF-LFIQA method, using the Win5-LID dataset. We compared the time consumption with the results of Exp4 (see Table 4.2.4), which corresponds to using the inputs in the spatial domain. Notice that Exp4 requires 3 seconds for pre-processing (loading images in RGB format), 4.8 hours for training, and 20 seconds for testing/running the model. The proposed method requires a

slightly longer time for data pre-processing, since it needs to compute the Fourier transform of the EPIs. But, the amount of time required for training and testing is significantly lower.

Table 4.2.5: The time consumption of DNNF-LFIQA method on Win5-LID dataset.

Method	Pre-Processing (seconds)	Training (hours)	Testing (seconds)
Exp4	3	4.8	20
DNNF-LFIQA	6	0.72	10

4.3 Conclusions

In this chapter [6], we presented two no-reference LF objective quality assessment method, HVS-CNN and DNNF-LFIQA. The HVS-CNN method is based on a two-stream CNN architecture, which is inspired by the human visual system. The two-stream network is able to extract rich distortion-related spatial and angular LF characteristics and predict the LF quality. The first stream of the architecture processes the angular information from Canny maps of EPIs generated from the corresponding LF contents, while the second stream processes the spatial information from mean Canny maps generated from Canny maps of SAIs. We also proposed a novel approach to generate multiple epipolar-plane images - the *MultiEPL*. The *MultiEPL* approach produced accurate results in comparison with a standard *SingleEPL*-based approach. We trained the HVS-CNN method using four LFI quality datasets. Results show that the HVS-CNN method outperforms other state-of-the-art LF-IQA methods. Results from a cross-database test and an ablation study show that the HVS-CNN method is robust and consistent.

The DNNF-LFIQA method is based on a deep neural network that uses as inputs frequency domain EPI representations. The method is composed of two processing streams, with the first stream taking as input the horizontal EPI and the second stream taking as input the vertical EPI. Both streams have identical CNN blocks that generate 1D feature vectors as outputs, which are later concatenated using two fusion blocks. Finally, the fused vector is fed to the regression block for quality estimation. We tested the proposed method on 3 different datasets, obtaining high correlation values when compared to other state-of-the art methods. We also performed a cross-database test and a simplified ablation test. In summary, these quantitative tests showed that the DNNF-LFIQA method is robust and accurate. The good performance and low complexity of DNNF-LFIQA makes it a good candidate for quality estimation in real-time or complexity-constrained applications. In future, we plan to investigate the performance of the DNNF-LFIQA method for different parameters.

[6] This chapter contains the research material published by Multimedia Tools and Applications (MTAP) [192]

Chapter 5

LF-IQA Methods Based on Long Short-Term Memory Network

In this chapter, we present two novel Long Short-Term Memory (LSTM) based Deep Neural Networks for NR LF image quality assessment, that predict visual quality by extracting long-term dependent distortion-related features. Our methods also incorporates bottleneck features to increase the number of input samples for training. The second method is an extended version of the first method in this chapter, and it incorporates diverse parameters in training. The key contributions of our work can be summarized as follows:

- We propose two novel and efficient blind LF-IQA methods that are based on a Long Short-Term Memory deep neural network. First method incorporates bottleneck features from a pre-trained neural network (LSTM-DNN), while the second method takes diverse parameters in terms of bottleneck features extracted from three pre-trained neural networks (LSTM-DP).
- We demonstrate the efficiency of using bottleneck features extracted from the pre-trained neural networks for quality estimation.
- We analyze the performance of different variants of the proposed architectures to show the robustness of the methods.

5.1 LF-IQA Method Based on a Long-Short Term Memory Neural Network (LSTM-DNN)

Figure 5.1.1 shows the block diagram of the proposed LSTM-DNN network, which has two streams. *Stream1* extracts primary features from the LF content and learns the long-term dependencies among them. *Stream2* processes bottleneck features generated from a pre-

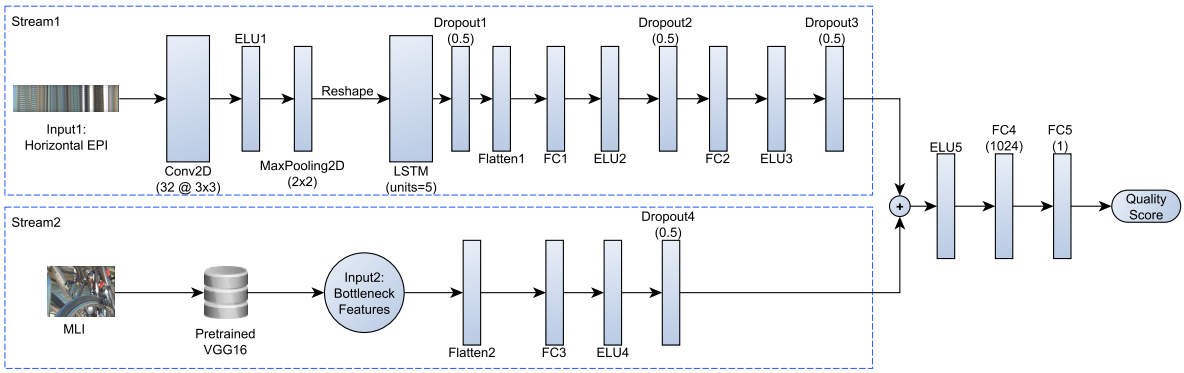


Figure 5.1.1: Block diagram of the proposed NR LSTM-DNN method.

trained neural network. The output of both streams is fused using a summation operation. The result is fed to a set of fully connected layers, which produce the quality prediction. Next, we describe each of the stages in Figure 5.1.1.

5.1.1 Stream1

In *stream1*, the first layer is an input layer that takes horizontal EPIs (in RGB color format) as input. The second layer is a CNN block, which acts as the primary-feature extractor for the Long Short-Term Memory (LSTM) layer. The CNN block includes Conv2D, ELU, and MaxPooling2D layers. The Conv2D layer filters the input image using a 3×3 kernel, where padding is applied to maintain the input size unchanged. This layer generates an output of 32 feature maps that are activated using an Exponential Linear Unit (ELU) activation function [117] in the ELU layer. The ELU layer is followed by a MaxPooling2D layer that uses stride and pool of size 2×2. The CNN output is reshaped to a 2D feature vector and fed to the third layer - the LSTM layer.

The LSTM layer uses 5 recursive LSTM memory units to extract long-term dependent distortion-related features from the primary features extracted with the CNN block. Then, the output from LSTM layer is fed to the flattening block (Flatten1), which forwards the output to two fully connected layers (FC1 and FC2). The FC1 and FC2 layers output feature vectors of size 512. Finally, Dropout layers are added after the LSTM, ELU2 and ELU3 layers to prevent overfitting.

5.1.2 Stream2

In *stream2*, the first layer is an input layer that takes bottleneck features of the LF content in a MLI format as input. The second layer includes a flattening block (Flatten2), which

forwards its output to a fully connected layer (FC3). The FC3 layer is activated by an ELU activation function. Finally, a Dropout layer is added to prevent overfitting.

We generate bottleneck features from the MLIs using a transfer learning approach. Specifically, we use a Very Deep Convolutional Networks for Large-Scale Image Recognition (VGG16) [193] pre-trained using ImageNet [194], which contains 1.2 million color images and 1,000 classes. In VGG16 model, layers 1 to 19 are part of the feature extraction, while layers 20 to 23 are part of the classification stage. We freeze layers 20 to 23 and extract bottleneck features from the 19th layer using the pre-trained ImageNet weights. To reduce memory requirements, we downsampled the LF MLIs to a spatial resolution of $256 \times 256 \times 3$ [82]. This way, for every input with dimension $256 \times 256 \times 3$, we obtain $512 \times 8 \times 8$ bottleneck features.

5.1.3 Quality Prediction

Table 5.1.1: LSTM-DNN network configuration parameters.

Layer	Output	Number of Parameters
Stream1 Input	(654, 81, 3)	0
CNN Block	(654, 81, 32)	896
MaxPooling2D	(327, 40, 32)	0
Reshape	(10464, 40)	0
LSTM	(10464, 5)	920
Flatten1	(52320)	26788352
FC1	(512)	0
FC2	(512)	0
Stream2 Input	(512, 8, 8)	0
Flatten2	(32768)	262656
FC3	(512)	16777728
Add	(512)	0
FC4	(1024)	525312
FC5	(1)	1025
Total Parameters		44,356,889

As shown in Figure 5.1.1, the outputs from *stream1* and *stream2* are fed to the Add layer, which performs a summation operation and generates a fused feature vector as output. The fused feature vector is then supplied to fully connected layers FC4 and FC5. Layers FC4 and FC5 perform a regression operation, in which the layer FC4 generates 1,024 output features and the layer FC5 generates a scalar output that corresponds to the estimated perceptual quality score. Table 5.1.1 lists the parameters used for the LSTM-DNN network. In this table, the CNN Block represents a Conv2D layer and an ELU layer. The LSTM block represents an

LSTM layer and a Dropout layer. Each FC block represents an FC layer, an ELU layer, and a Dropout layer. Finally, the Add block represents an Add layer and an ELU layer.

5.1.4 Experimental Setup

To train and test the proposed LSTM-DNN method, we have used 3 light field image quality datasets: MPI [7], Win5-LID [3], and LFDD [4]. We used SROCC and PLCC as performance evaluation methods. We compared the proposed NR LF-IQA method with the following state-of-art LF-IQA methods: SDFM [107], MDFM [180], LFIQM [75], Tensor-NLFQ [79], NR-LFQA [77], LF-QMLI [78], VBLIF [183], Ak *et al.* [80], BELIF [182], DeLFIQE [85], Guo *et al.* [82], LF-ASC [83]. We also compared the method with the following 2D-FR IQA methods [72, 74]: UQI, VIF, GMSD, NIQE, SSIM, IW-SSIM, IW-PSNR, FI-PSNR, MW-PSNR, MJ3DFR, PSNR-YUV and STMAD.

For training and testing, we divided each dataset into three content-independent training, validation, and test subsets. In this division, test (possibly distorted) images generated from one reference can only be in one of the subsets, i.e., if images corresponding to a specific reference content are in the test subset, they are not in the training and validation subsets and vice-versa. More specifically, we define a group of scenes as a group containing the reference LFI and its corresponding distorted versions. Then, 60% of the groups were randomly selected for training, 20% for validation, and the remaining 20% were used for testing. We report the correlation values only for the test subset. We trained the no-reference LSTM-DNN using mini-batches of size 128, 6,000 epochs, and Mean Square Error (MSE) as the training loss. Also, we used the Stochastic Gradient Descent (SGD) optimizer with a learning rate 0.0001 to minimize the loss function. We implemented the proposed method using Keras [190] library of Python. The method was trained and tested on 25GB GPU, with a LINUX environment. The code of the proposed LF-IQA method is available for download on GitHub[1] under a general public license.

5.1.5 Experimental Results

Figure 5.1.2 illustrates graphs of loss versus epoch for both training and validation using the three LF-IQA datasets. For datasets MPI and Win5-LID, the LSTM-DNN converges well after epoch 1500, achieving very small loss values afterwards. But, for dataset LFDD, the proposed method has shown different a different behavior, with higher training and validation losses until around 2500 epochs, after which, the LSTM-DNN converges to a small loss value, with very close and parallel training and validation loss curves.

[1] `https://bit.ly/2YT3FIz`

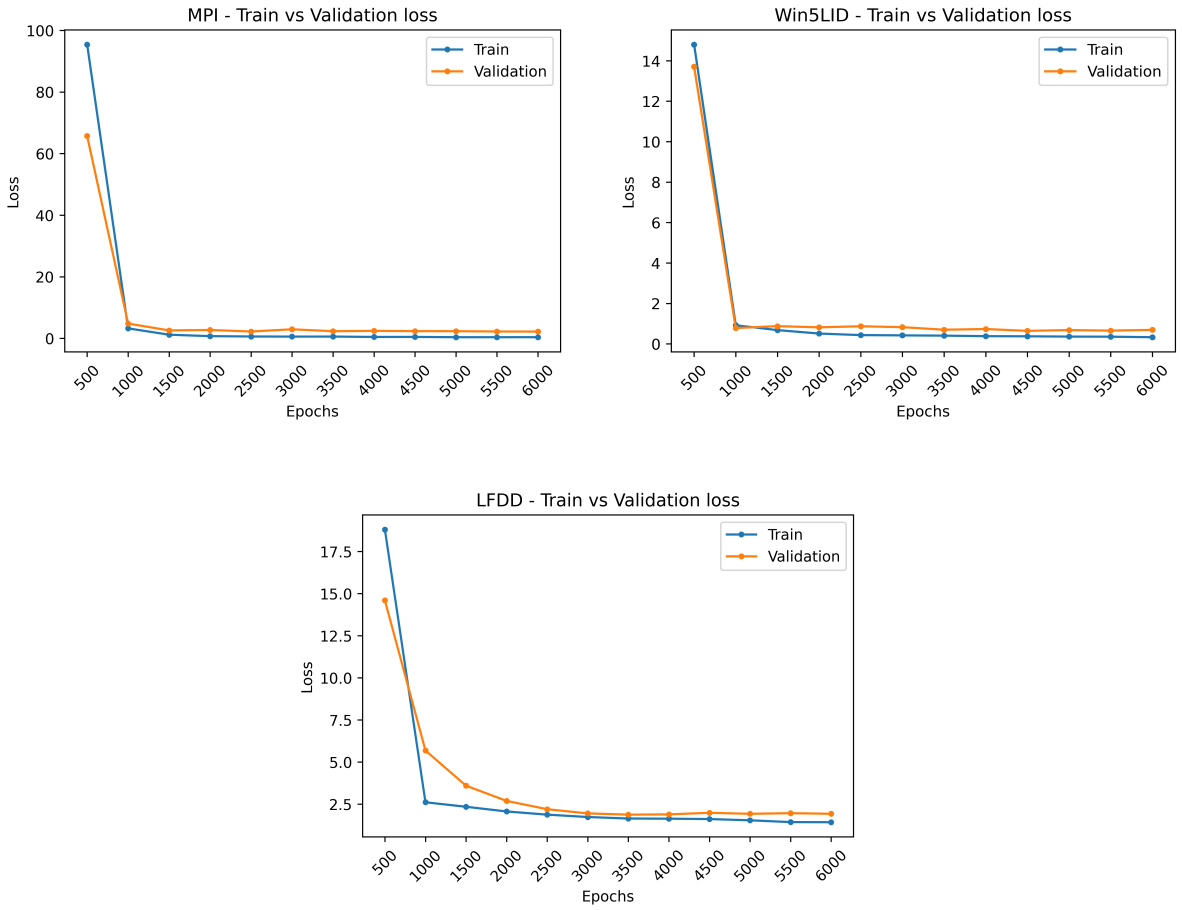


Figure 5.1.2: Train vs Validation Loss of the proposed LSTM-DNN method on 3 LF-IQA test datasets.

Table 5.1.2 shows the correlation values obtained for the MPI, Win5-LID, and LFDD LFI quality datasets. The rows in this table show the results for each dataset and for each distortion, with the 'All' row corresponding to the results obtained for the complete datasets. The proposed method performs very well for the complete MPI dataset ('All' row), obtaining a SROCC of 0.94 and a PLCC of 0.97. For the complete Win5-LID dataset ('All' row), the method achieves a SROCC of 0.95 and a PLCC of 0.96. In these two datasets, there was not a large variation of correlation among distortion, with the largest difference being for the NN distortion in Win-5LID (SROCC of 0.89). For the LFDD dataset, the method obtained smaller correlation values, with a SROCC of 0.80 and PLCC of 0.74. In this dataset, 4 distortions (Pincushion, BAR, JPEG2000, Impulse and Gaussian) show lower correlation values, while the other 8 distortions have high correlation values. The dataset LFDD has diverse and complex contents (foreground and background have the same range of pixel values), which is probably the reason why the proposed method does not perform well in terms of correlation with

Table 5.1.2: The SROCC and PLCC values for LFDD, Win-5LID, and MPI datasets.

Dataset	Distortion	PROPOSED	
		SROCC	PLCC
LFDD	AV1	0.8999	0.4979
	BAR	0.3999	0.6631
	BPG	0.9999	0.9671
	Gaussian	0.7000	0.7348
	JPEG2000	0.6000	0.7547
	JPEG	0.9999	0.9221
	Pincushion	0.3000	0.2313
	Impulse	0.8999	0.6770
	Unsharp Mask	0.9999	0.9499
	VP9	0.9999	0.8449
	x264	0.8999	0.9342
	x265	0.9999	0.9493
	ALL	0.8083	0.7432
Win-5LID	HEVC	0.9899	0.9714
	JPEG2000	0.9581	0.9582
	LN	0.9300	0.9824
	NN	0.9300	0.9450
	ALL	0.9515	0.9680
MPI	QD	0.9271	0.9618
	HEVC	0.9505	0.9569
	OPT	0.9142	0.9743
	Linear	0.9428	0.9778
	NN	0.8966	0.9598
	ALL	0.9484	0.9700

the subjective quality scores provided with this dataset [195].

Table 5.1.3 illustrates the results of the comparison of the proposed method with other state-of-the-art LFI-IQA methods. Since the results for some LF-IQA methods are not available for all 3 datasets, the table is incomplete. For the MPI dataset, the proposed method has obtained the highest correlation values. For the Win5-LID dataset, the proposed method is the second best performing method in terms of PLCC, but the method proposed by Guo *et al.* [82] has a slightly better performance in terms of SROCC (a 0.0083 difference). For the LFDD dataset, although the proposed method obtained lower SROCC and PLCC values than for other datasets, it outperformed all other LFI-IQA methods. More specifically, it obtained an SROCC of 0.80 and a PLCC of 0.74, while the other methods achieved a maximum SROCC of around 0.52 and a maximum PLCC of around 0.41.

Table 5.1.4 presents the time required to train and run the proposed method. The proposed method was implemented using the Keras library and run on a 25GB GPU using a Linux environment. For comparison, we present the results reported by DeLFIQE [85]. It is worth mentioning that their code is in MATLAB and it was run on a 4GB GPU.

Finally, we conducted a simplified ablation test of the proposed NR LSTM-DNN method using only the Win5-LID database. In this test, we performed two experiments. In first ex-

Table 5.1.3: SROCC and PLCC values of state-of-the-art LF-IQA methods, tested on MPI, Win5-LID, and LFDD datasets.

Category	**Type**	**methods**	**Year**	**MPI**		**Win5-LID**		**LFDD**	
				SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Pre-defined Functions	FR	UQI	2002	0.7400	0.8460	0.8252	0.8764	0.4673	0.3486
	FR	SSIM	2004	0.9120	0.9320	0.6812	0.7880	0.4488	0.2457
	FR	VIF	2006	0.8600	0.8960	0.9347	0.9555	0.4588	0.4026
	FR	NIQE	2009	0.5821	0.5122	0.4892	0.5002	0.5235	0.4138
	FR	STMAD	2011	0.8650	0.8940	0.8489	0.9074	0.2054	0.2005
	FR	IW-SSIM	2011	0.9320	0.9440	0.8212	0.8736	0.4432	0.2594
	FR	IW-PSNR	2011	0.9300	0.9160	0.8842	0.9022	0.4184	0.3060
	FR	MJ3DFR	2013	0.8720	0.9300	0.8836	0.8998	0.4235	0.3182
	FR	GMSD	2014	0.7358	0.7410	0.4352	0.5041	0.4384	0.4000
	FR	FI-PSNR	2014	0.8740	0.8510	0.6951	0.7419	0.2415	0.1645
	FR	PSNR-YUV	2014	0.9342	0.9452	0.9007	0.9215	0.4325	0.4124
	FR	MW-PSNR	2016	0.7251	0.7698	0.7582	0.7758	0.4021	0.3842
	FR	MDFM [180]	2018	0.8346	0.8123	0.8157	0.8591	-	-
	RR	LFIQM [75]	2019	0.6815	0.7013	0.4503	0.4763	0.1245	0.1041
	FR	SDFM [107]	2020	0.8435	0.8423	0.6742	0.7142	-	-
	NR	BELIF [182]	2019	0.8854	0.9096	0.8719	0.8910	-	-
	NR	NR-LFQA [77]	2019	0.9119	0.9155	0.9032	0.9206	0.4188	0.4033
	NR	LF-QMLI [78]	2019	-	-	0.8802	0.9038	-	-
SVR	NR	Tensor-NLFQ [79]	2019	0.9101	0.9225	0.9101	0.9217	0.5134	0.4124
	NR	Ak *et al.* [80]	2020	0.8942	0.9005	-	-	-	-
	NR	VBLIF [183]	2020	0.9015	0.9158	0.9009	0.9232	-	-
CNN	NR	LF-ASC [83]	2021	-	-	0.9260	0.9257	-	-
	NR	Guo *et al.* [82]	2021	-	-	**0.9598**	0.9535	-	-
	NR	DeLFIQE [85]	2022	0.9515	0.9520	-	-	-	-
	NR	**Proposed**	2021	**0.9484**	**0.9700**	0.9515	**0.9680**	**0.8083**	**0.7432**

Table 5.1.4: The time consumption of LSTM-DNN method on MPI dataset, with the best performance results in bold.

Method	Pre-Processing (minute)	Training (hour)	Testing (minute)
DeLFIQE [85]	195	5	0.3
LSTM-DNN	**4**	**2.3**	**0.06**

Table 5.1.5: Ablation test on Win5-LID Dataset, with the best performance results in bold.

Dataset	Experiment	SROCC	PLCC
Win5-LID	Exp1: Without LSTM Layer and Bottleneck Features	0.5116	0.5028
	Exp2: Without Bottleneck Features	0.8467	0.8650
	Proposed method	**0.9515**	**0.9680**

periment (Exp1), we split the network into one stream, that included input layer, CNN block, and the last two fully connected layers. Specifically, in this experiment, we did not include

the LSTM layer and the *stream2*. In a second experiment (Exp2), we kept the *stream1* as it is, but we did not include the bottleneck features. Using these modifications, we trained and tested the method only on the Win5-LID dataset. Table 5.1.5 depicts the SROCC and PLCC values for these two experiments. Notice that, for Exp1, there is a decrease in performance, in terms of SROCC and PLCC, when compared to the (full) proposed method. More specifically, for Exp1 there is a SROCC difference of 0.4399 and a PLCC difference of 0.4652. In Exp2, the performance is slightly better when compared to Exp1, but the proposed method still performs better with a SROCC difference of 0.1048 and a PLCC difference of 0.1030. In summary, this study has demonstrated the positive impact of using LSTM and bottleneck features in the proposed architecture.

5.2 LF-IQA Method Based on Long Short-Term Memory Network with Diverse Parameters (LSTM-DP)

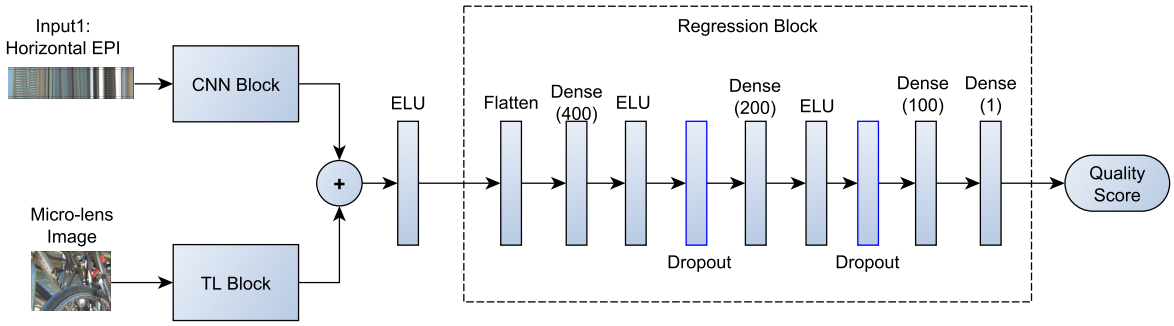


Figure 5.2.1: Block diagram of the proposed no-reference LSTM-DP method.

Figure 5.2.1 shows the block diagram of the proposed LSTM-DP network, which has two streams. The main stream consists of a CNN block, which extracts high level features from the LF content and learns the long-term dependencies among them. Second stream processes the bottleneck features generated from three pre-trained neural networks using transfer learning (TL) approach. The output of both streams is fused using a concatenation operation, which later is activated by an Exponential Linear Unit (ELU) activation [117] function. Then, the final result is fed to a regression block, which contains a set of fully connected layers, and generates the quality prediction. Next, we describe each of the stages in Figure 5.2.1.

5.2.1 CNN Block

As shown in Figure 5.2.1, the input layer of CNN block processes the horizontal EPI in RGB color format. Overall, the CNN block consists of five 2D convolutional layers, where each layer is followed by one ELU layer, and one 2D MaxPooling layer with pool and stride of sizes 2×2. First and second Conv2D layers generate 32 output features with a 3×3 kernel. Third and fourth Conv2D layers generate 64 output features with a 3×3 kernel. The last Conv2D layer generates 128 output features with same size of the kernel that is 3×3. The output of last Conv2D is reshaped into a 2D feature vector, which is then supplied to an LSTM layer.

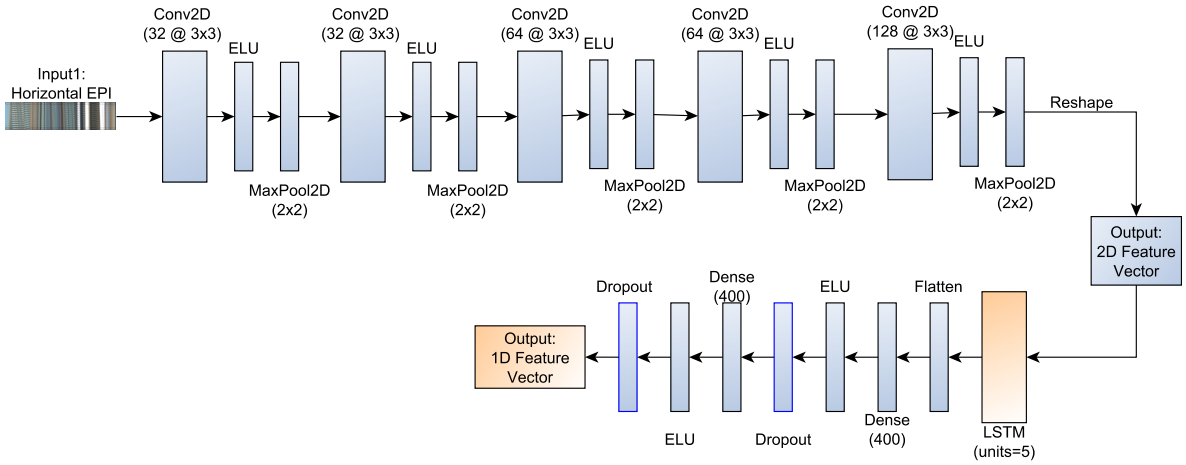


Figure 5.2.2: Block diagram of CNN block in the proposed no-reference LSTM-DP method.

The LSTM layer uses 5 recursive memory units to extract long-term dependent distortion-related features. Then, the output from LSTM layer is fed to the flattening block (Flatten), which forwards the output towards two fully connected Dense layers. The two Dense layers generate one-dimensional (1D) feature vectors of size 400, and each Dense layer is followed by one ELU layer and one Dropout layer. The Dropout layers help to prevent overfitting in training process.

5.2.2 Transfer Learning Block

As shown in Figure 5.2.3, the first layer in the transfer learning (TL) block is an input layer that takes an MLI in RGB color format. The input is supplied to three consecutive TL streams that extract bottleneck features from thee pre-trained neural networks. The rest of the layers in three streams are identical, i.e., one Flatten layer, one Dense layer with 512 output features, and one ELU layer. The outputs of TL streams are fused by a concatenation operation,

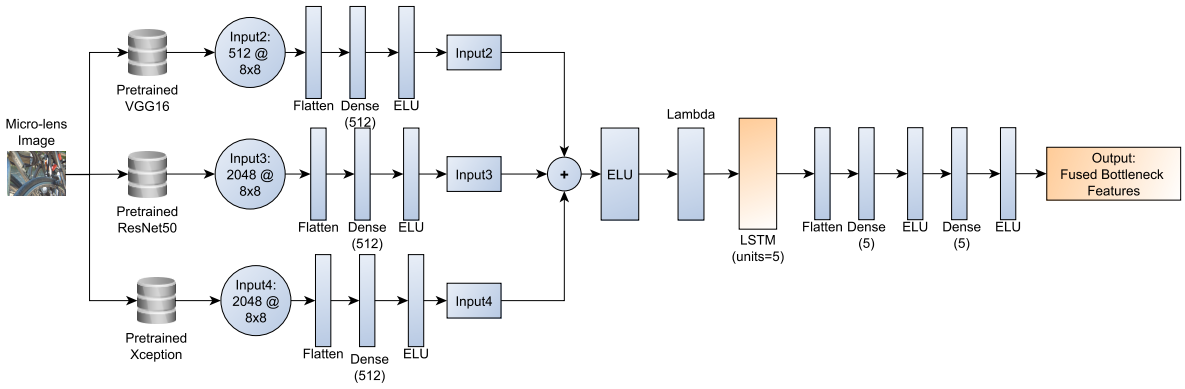


Figure 5.2.3: Block diagram of the transfer learning block in the proposed no-reference LSTM-DP method.

and after applying ELU activation, we get a 1D fused feature vector (FFV). After converting 1D FFV into 2D FFV, we pass this vector to an LSTM layer. LSTM layer further process 2D FFV, and learns fine distortion-related features. The output of LSTM layer is forwarded to fully connected operation that is performed by one Flatten layer, two Dense layers with 5 output features and two ELU activation layers. At the end of TL block, we obtain a vector of fused bottleneck features (FBF).

We generate bottleneck features from the MLIs using a transfer learning approach. Specifically, we use the VGG16 [193], ResNet50 [196], and Exception [197] networks that are pre-trained on ImageNet [194] dataset with 1.2 million color images and 1,000 classes. In VGG16, ResNet50 and Xception models, the last fully connected layers are used for classification predictions. In this work, we freeze the last layers in three models, and extract bottleneck features from the last convolutional or pooling layers directly using the pre-trained ImageNet weights. To reduce memory requirements, we downsampled the LF MLIs to a spatial resolution of $256 \times 256 \times 3$ [82]. This way, for every input with dimension $256 \times 256 \times 3$, we obtain $512 \times 8 \times 8$, $2048 \times 8 \times 8$, and $2048 \times 8 \times 8$ bottleneck features from VGG16, ResNet50 and Xception networks, respectively.

5.2.3 Regression Block

As shown in Figure 5.2.1, we add the outputs of CNN and TL blocks by performing a concatenation operation, and then apply an ELU activation to the fused features. The fused feature vector is then supplied to a regression block, that contains four Dense layers, two ELU activation layers, and two Dropout layers. First, second and third Dense layers output 400, 200 and 100 features, respectively. The last Dense layer generates a scalar number that corresponds to the estimated perceptual quality score. Table 5.2.1 lists the parameters used

Table 5.2.1: The configuration parameters of the proposed LSTM-DP network .

CNN Block		**TL Block**	
Layer	**Output**	**Layer**	**Output**
input_1	((100, 960, 3)	input_2	(512, 8, 8)
Conv2D	(100, 960, 32)	Flatten	(32768)
MaxPooling2D	(50, 480, 32)	Dense2	(512)
Conv2D	(50, 480, 32)	input_3	(2048, 8, 8)
MaxPooling2D	(25, 240, 32)	Flatten	(131072)
Conv2D	(25, 240, 64)	Dense3	(512)
MaxPooling2D	(12, 120, 64)	input_4	(2048, 8, 8)
Conv2D	(12, 120, 64)	Flatten	(131072)
MaxPooling2D	(6, 60, 64)	Dense4	(512)
Conv2D	(6, 60, 128)	Concatenate (2,3,4)	(1536)
MaxPooling2D	(3, 30, 128)	-	-
Reshape	(90, 128)	Lambda	(1, 1536)
LSTM	(90, 5)	LSTM	(1, 5)
Flatten	(450)	Flatten	(5)
Dense	(400)	Dense	(5)
Dense*	(400)	Dense**	(5)
Fusion layer			
Concatenate (*, **)		(405)	
Regression Block			
Dense		(400)	
Dense		(200)	
Dense		(100)	
Dense		(1)	

for the LSTM-DP network. In this table, for simplicity, we have not included the ELU and Dropout layers.

5.2.4 Experimental Setup

To train and test the proposed LSTM-DP method, we have used 3 light field image quality datasets MPI, VALID, and LFDD. We have chosen these datasets because of the diversity of their visual contents, types of distortions and the availability of the corresponding subjective quality scores. We used SROCC and PLCC as performance evaluation methods. We compared the proposed NR LF-IQA method with the following state-of-art LF-IQA methods: SDFM [107], MDFM [180], LFIQM [75], Tensor-NLFQ [79], NR-LFQA [77], BELIF [182], and DeLFIQE [86]. We also compared the method with the following 2D-FR IQA methods [72,74]: UQI, SSIM, GMSD, NIQE, FI-PSNR, MW-PSNR, PSNR-YUV and STMAD.

For training and testing, we divided each dataset into three content-independent training, validation, and test subsets. In this division, test (possibly distorted) images generated from one reference can only be in one of the subsets, i.e., if images corresponding to a spe-

cific reference content are in the test subset, they are not in the training and validation subsets and vice-versa. More specifically, we define a group of scenes as a group containing the reference LFI and its corresponding distorted versions. Then, 80% of the groups were randomly selected for training, 10% for validation, and the remaining 10% were used for testing. We report the correlation values only for the test subset. We trained the NR LSTM-DP using mini-batches of size 128, 6,000 epochs, and Mean Square Error (MSE) as the training loss. Also, we used the Stochastic Gradient Descent (SGD) optimizer [118] with a learning rate 0.0001 to minimize the loss function. We implemented the proposed method using Keras [190] library of Python. The method was trained and tested on 25GB GPU, with a LINUX environment. The code of the proposed LF-IQA method is available for download on GitHub[2] under a general public license.

5.2.5 Experimental Results

Figure 5.2.4 displays the train and validation loss curves obtained for the proposed LSTM-DP method, when trained and tested on 3 different LF-IQA test datasets. For the dataset MPI, the training and validation losses decrease continuously until 1200 epoch, and we see best fitting curves until the last epoch. For dataset VALID, training and validation losses continuously decrease up to the last epoch, without any fluctuation. For datasets VALID and MPI, the proposed method achieved training and validation loss values smaller than one. On the other hand, the dataset LFDD, that contains complex LF images, shows a different behaviour, with training loss values smaller than one, but with higher validation loss values.

Table 5.2.2 shows the correlation values obtained for the MPI, VALID, and LFDD LFI quality datasets. The rows in this table show the results for each dataset and for each distortion, with the 'All' row corresponding to the results obtained for the complete datasets. The proposed method performs very well for the complete MPI dataset ('All' row), obtaining an SROCC of 0.96 and a PLCC of 0.97. For the complete VALID dataset ('All' row), the method achieves an SROCC of 0.94 and a PLCC of 0.96. In these two datasets, there was not a large variation of correlation among distortion. For the LFDD dataset, the method obtained smaller correlation values, with an SROCC of 0.77 and a PLCC of 0.82. In this dataset, 4 distortions (x265, x264, VP9, and Gaussian) show lower correlation values, while the other 7 distortions have high correlation values. The dataset SMART has obtained high correlation values, because the opinion scores for the 2D views in SMART dataset have been obtained by the Pairwise Comparison method, which has high discriminatory power when several test items are nearly equal in quality [198] helping the quality metrics perform better.

[2] `https://bit.ly/3veXeLP`

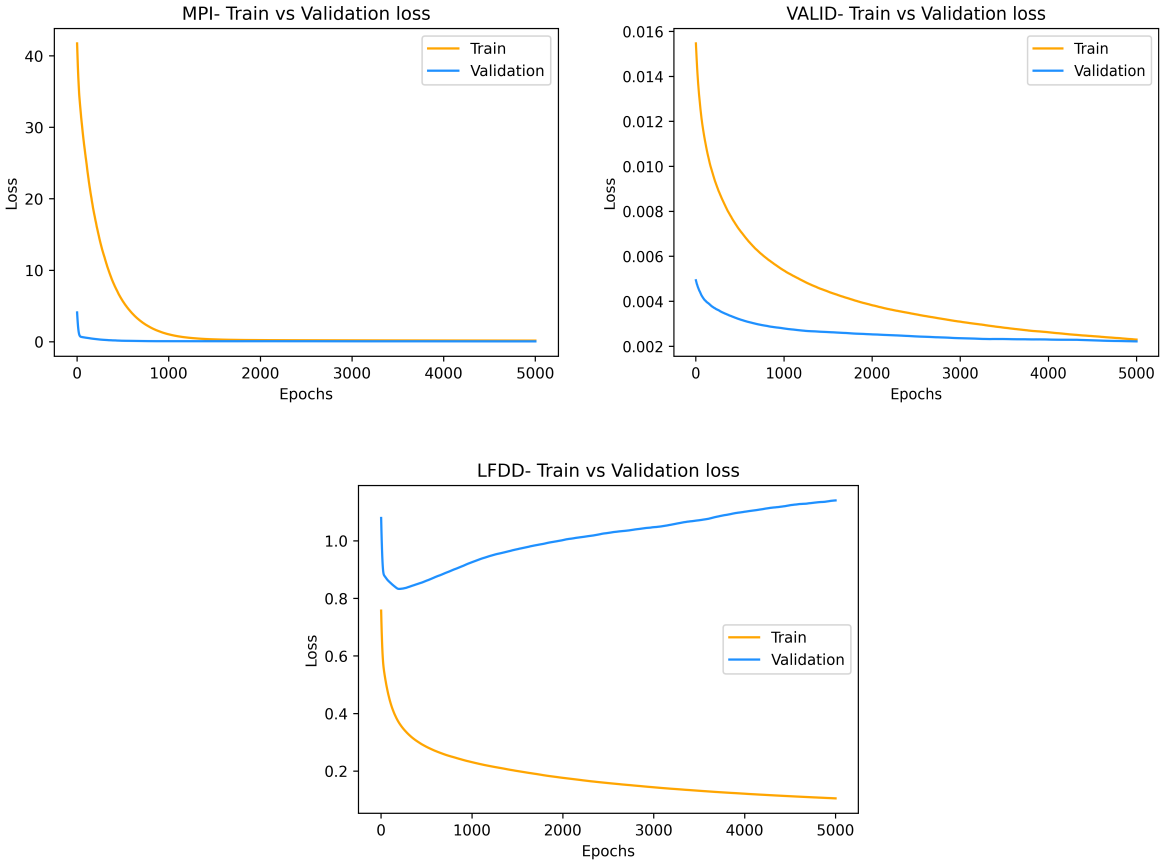


Figure 5.2.4: Train vs Validation Loss of the proposed LSTM-DP method on 3 LF-IQA test datasets.

Table 5.2.3 illustrates the results of the comparison of the proposed method with other state-of-the-art LFI-IQA methods. For the MPI and VALID datasets, the proposed method is the best performing method in terms of PLCC and SROCC. For the LFDD dataset, although the proposed method obtained lower SROCC and PLCC values than for other datasets, it outperformed all other LFI-IQA methods. More specifically, it obtained an SROCC of 0.77 and a PLCC of 0.82, while the other methods achieved a maximum SROCC of around 0.52 and a maximum PLCC of around 0.41.

Table 5.2.4 presents the time required to train and run the proposed method. The proposed method was implemented using the Keras library and run on a 25GB GPU using a Linux environment. For comparison, we present the results reported by DeLFIQE [86], in which the code is written in MATLAB and it was run on a 4GB GPU. Notice that the DeLFIQE requires 195 minutes for pre-processing / feature extraction, 5 hours for training, and 0.3 minutes for testing the model. On the contrary, the amount of time required by the proposed LSTM-DP method for pre-processing and testing, is significantly lower. But, for training, the

Table 5.2.2: The SROCC and PLCC values for MPI, VALID and LFDD datasets.

Dataset	Distortion	PROPOSED	
		SROCC	PLCC
LFDD	BAR	0.8572	0.8176
	BPG	0.8783	0.9813
	Gaussian	0.6655	0.8446
	JPEG2000	0.8193	0.8404
	JPEG	0.7608	0.8483
	Pincushion	0.8324	0.7172
	Impulse	0.8204	0.8077
	Unsharp Mask	0.8783	0.8929
	VP9	0.6833	0.9186
	x264	0.6884	0.7139
	x265	0.6743	0.7288
	ALL	0.7708	0.8293
VALID	10bit_HEVC	0.9783	0.9829
	10bit_VP9	0.9527	0.9628
	8bit_HEVC	0.9486	0.9661
	8bit_VP9	0.9056	0.9674
	ALL	0.9460	0.9696
MPI	QD	0.9559	0.9608
	Gaussian	0.9819	0.9853
	HEVC	0.9536	0.9792
	OPT	0.9683	0.9816
	Linear	0.9528	0.9740
	NN	0.9520	0.9790
	ALL	0.9608	0.9766

Table 5.2.3: SROCC and PLCC values of state-of-the-art LF-IQA methods, tested on MPI, VALID and LFDD datasets.

Category	**Type**	**methods**	**Year**	**MPI**		**VALID**		**LFDD**	
				SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Pre-defined Functions	FR	UQI	2002	0.7400	0.8460	0.9310	0.9550	0.4673	0.3486
	FR	SSIM	2004	0.9120	0.9320	0.9500	0.9640	0.4488	0.2457
	FR	NIQE	2009	0.5821	0.5122	0.6211	0.6544	0.5235	0.4138
	FR	STMAD	2011	0.8650	0.8940	0.7940	0.8020	0.2054	0.2005
	FR	GMSD	2014	0.7358	0.7410	0.6821	0.6948	0.4384	0.4000
	FR	FI-PSNR	2014	0.8740	0.8510	0.7060	0.7060	0.2415	0.1645
	FR	PSNR-YUV	2014	0.9342	0.9452	0.9230	0.9310	0.4325	0.4124
	FR	MW-PSNR	2016	0.7251	0.7698	0.6869	0.6904	0.4021	0.3842
	FR	MDFM [180]	2018	0.8346	0.8123	0.7120	0.7198	-	-
	RR	LFIQM [75]	2019	0.6815	0.7013	0.3934	0.5001	0.1245	0.1041
	FR	SDFM [107]	2020	0.8435	0.8423	0.8240	0.8542	-	-
SVR	NR	BELIF [182]	2019	0.8854	0.9096	0.8863	0.8950	-	-
	NR	NR-LFQA [77]	2019	0.9119	0.9155	0.9233	0.9316	0.4188	0.4033
	NR	Tensor-NLFQ [79]	2019	0.9101	0.9225	0.8702	0.9028	0.5134	0.4124
CNN	NR	DeLFIQE [86]	2022	0.9515	0.9520	-	-	-	-
	NR	**Proposed**	2022	**0.9608**	**0.9766**	**0.9460**	**0.9696**	**0.7708**	**0.8293**

method requires an equal amount of time which is 5 hours.

Finally, we conducted a simplified ablation test of the proposed NR LSTM-DP method using only the MPI dataset. In this test, we performed four experiments. In first experiment

Table 5.2.4: The time consumption of LSTM-DP method on MPI dataset, with the best performance results in bold.

Method	Pre-Processing (minute)	Training (hour)	Testing (minute)
DeLFIQE [85]	195	5	0.3
LSTM-DP	**4.6**	5	**0.7**

(Exp1), we split the network so that it contains only the CNN block, and regression block. In second experiment (Exp2), we split the network such that it contains only TL block, and regression block. In third experiment (Exp3), we used the model shown in Figure5.2.1, but we removed LSTM layers from CNN block and TL block. In fourth experiment (Exp4), we used the model shown in Figure5.2.1, but we created three separate training models such that each model has only one of three pre-trained networks, i.e., model1 with VGG16, model2 with ResNet50, and model3 with Xception in TL block.

Table 5.2.5: Ablation test on MPI Dataset, with the best performance results in bold.

Dataset	Experiment	SROCC	PLCC
MPI	Exp1: Without TL Block	0.8501	0.8551
	Exp2: Without CNN Block	0.3086	0.3497
	Exp3: Model in Figure 5.2.1, but without LSTM layers in CNN and TL blocks.	0.5198	0.6583
	Exp4: Model1 in Figure 5.2.1, but with only VGG16 in TL block.	0.6583	0.6764
	Exp4: Model2 in Figure 5.2.1, but with only ResNet50 in TL block.	0.4886	0.5332
	Exp4: Model3 in Figure 5.2.1, but with only Xception in TL block.	0.5514	0.6199
	Proposed method	**0.9515**	**0.9680**

Table 5.2.5 shows the SROCC and PLCC values for these four experiments. Notice that, for Exp1, there is a decrease in performance, in terms of SROCC and PLCC, when compared to the (full) proposed method. More specifically, for Exp1 there is an SROCC difference of 0.1014 and a PLCC difference of 0.1129. In Exp2, the performance deteriorates mor,e when compared to Exp1, and the proposed method still performs better with an SROCC difference of 0.3535 and a PLCC difference of 0.6183. Even for Exp4, none of the models (model1, model2 and model3) with single pre-trained network (VGG16, ResNet50 and Xception) performed well with respect to the correlation values for MPI dataset. Specifically, by closely looking at the results, we find that, the model2 with only ResNet50 pre-trained network, has

worst performance in terms of lowest SROCC of 0.4886 and PLCC of 0.5332 value, when compared to model1 (SROCC of 0.6583 and PLCC of 0.6764) and model3 (SROCC of 0.5514 and PLCC of 0.6619). In summary, this study has demonstrated the positive impact of using LSTM layers and bottleneck features extracted from three pre-trained networks VGG16, ResNet50 and Xception, in the proposed architecture.

5.3 Conclusion

In this chapter, we have presented two Long Short-Term Memory based Deep Neural Networks for NR LF image quality assessment method, called LSTM-DNN and LSTM-DP. The LSTM-DNN method is composed of two processing streams. The first stream consists of a CNN block that extracts primary features from horizontal EPIs. We fed these features to an LSTM network that extracts long-term dependent distortion-related features. The second stream processes bottleneck features of MLIs that are generated from a pre-trained VGG16 network.

The LSTM-DP method not only extracts the long-term dependent distortion-related features from LFIs with the help of LSTM, but also incorporates diverse parameters to increase the number of input samples for training. Specifically, the LSTM-DP method consists of two processing streams. The first stream consists of a CNN block that extracts the high level features from horizontal EPIs in RGB color format, and an LSTM layer to learn distortion-related features. The second stream of the LSTM-DP method is based on the transfer learning (TL) block, which extracts bottleneck features of MLI using three pre-trained networks VGG16, ResNet50, Xception, and fuse these features to generate a bottleneck feature vector. We used ImageNet pre-trained weights of the selected networks to extract bottleneck features. Then, the outputs of CNN and TL blocks are fused by performing concatenation operation, and forwarded to a regression block for quality prediction.

For both LSTM-DNN and LSTM-DP, results show that these methods are robust and accurate, outperforming several state-of-the-art LF-IQA methods. Specifically, we noticed that the LSTM-DP method has performed better than the LSTM-DNN method with respect to the correlations. Ablation tests have shown the importance of the LSTM and bottleneck features in LSTM-DNN and LSTM-DP methods.

Chapter 6

LF-IQA Methods with Dense Atrous Convolutions

In this chapter, our focus is to explore the incorporation of spatial and angular features extracted from both the horizontal and vertical EPIs into the design of a CNN-based NR LF-IQA method. We use a CNN architecture with Atrous Convolution Layers (ACL), which are also known as dilation convolution layers. In this architecture, the insertion of zeros between the filter elements increases the size of receptive field of kernel, which allows the CNN to cover more relevant information. Specifically, in this chapter, we discuss two proposed methods to assess the quality of simple and complex LF contents. First method is based on Atrous Convolution Layers, while the second method is based on ACL, and Long Short-Term Memory (LSTM) layers. The incorporation of LSTM layers make the proposed method more accurate by learning long-term dependent distortion-related features.

In summary, the main contributions of this work are as follows:

- The design of a deep neural network (CNN-ACL) architecture that is able to extract dense spatial and angular information from EPIs.
- The design of a diverse neural network (ACL-LSTM) that is able to learn long-term dependent distortion-related features from spatial and angular information of EPIs.
- A detailed comparison analysis on different variants of the proposed methods showing that our network outperforms state-of-the-art LF-IQA methods on existing LFI datasets.

6.1 LF-IQA Method with Dense Atrous Convolutions (CNN-ACL)

Figure 6.1.1 shows the block-diagram of the proposed CNN-ACL method of NR LF-IQA. Notice that the method has two streams, each composed of an independent CNN-ACL architecture that takes as input either the horizontal or vertical EPIs. Both streams first extract primary features of inputs using CNNs and, then, extract high-level features using the block of ACLs. The horizontal EPI is fed to the 1st stream, while the vertical EPI is fed to the 2nd stream. Finally, the output from both streams is concatenated and fed to the regression block that estimates the LFI quality score. Next, we describe each of the stages in Figure 6.1.1.

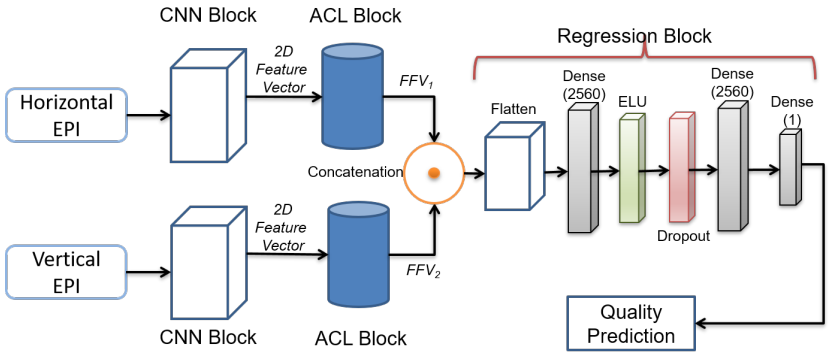


Figure 6.1.1: Block Diagram of the proposed CNN-ACL method.

6.1.1 CNN Block

Figure 6.1.2 shows the CNN block, which contains six stages that extract primary features from an input RGB image. Stages 1, 2, and 5 are identical, with the first layer being a 2D convolution layer with 32 output filters and a 3×3 kernel, the second layer an ELU activation layer, and the third layer a 2D max-pooling layer with a 2×2 pool and a 2×2 stride. Stages 3 and 4 are identical, with the first layer being a 2D convolution with 64 output filters and a 3×3 kernel and the second layer being an ELU activation layer. Stage 6 contains a reshape layer that takes a 3D input and converts to a 2D feature vector. To keep the output of the CNN block consistent for both streams, we have used a fixed size of 128.

6.1.2 ACL Block

As mentioned earlier, ACLs can be used to effectively enlarge the network's receptive field and capture richer spatial and angular features, without increasing the number of parameters. More specifically, ACL is a dilated convolution where zeroes are added between the

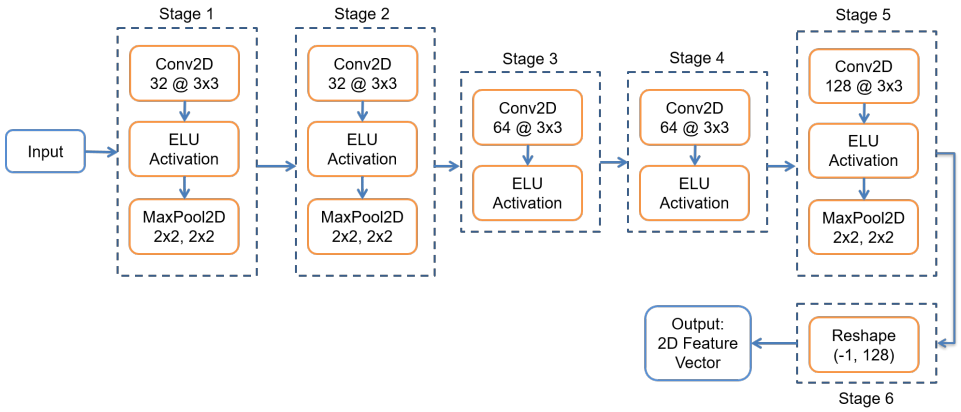


Figure 6.1.2: Illustration on CNN Block in CNN-ACL Method.

weights of the convolution kernel. In the simpler case of a 1D Atrous convolution, the output of the signal is defined as follows [199]:

$$y[i] = \sum_{k=1}^{K} x[i + rK] \cdot \omega[k] \tag{6.1.1}$$

where r is the dilation rate (or Atrous rate), $\omega[k]$ is the filter of length K, $x[i]$ is the input, and $y[i]$ is the output of a pixel.

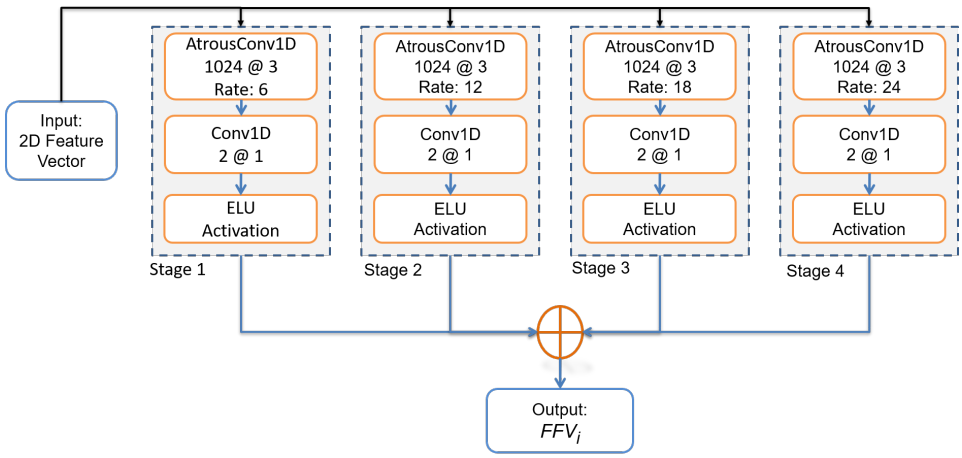


Figure 6.1.3: Illustration on ACL Block in CNN-ACL Method.

Figure 6.1.3 illustrates the architecture of the proposed ACL block that consists of 4 stages. Each stage is composed of a 1D Atrous Convolution layer with 1024 output filters and a kernel length of 3, a 1D convolution layer with 2 output filters and a kernel length of 1, and an ELU activation layer. To keep the ACLs distinct in every stage, we use use four dilation rates: 6, 12, 18 and 24 [119]. From the ACL block, we obtain a fused feature vector by the

summation of the outputs y_i from each i-th stage, as follows:

$$FFV_i = y_1 + y_2 + y_3 + y_4, \tag{6.1.2}$$

where the i index corresponds to the stream ($i = 1,2$) of the proposed method (see Figure 6.1.1).

6.1.3 Regression Block

As shown in Figure 6.1.1, the outputs of both CNN-ACL streams are concatenated as follows:

$$FFV = FFV_1 \oplus FFV_2, \tag{6.1.3}$$

where FFV represents the concatenated feature vector and '$\oplus$' represents the concatenation operation. The concatenated feature vector is fed to the regression block, which is composed by one flatten layer and three dense layers. The first dense layer has 2560 features and is followed by the ELU and Dropout layers. The second dense layer has 2560 features and is followed by the last dense layer. The outputs of this last dense layer is a one-dimensional scalar number that corresponds to the estimated perceptual quality score. Table 6.1.1 lists the concrete network configuration of CNN-ACL, in which, every layer with * is followed by one ELU activation layer.

6.1.4 Experimental Setup

To train and test the proposed CNN-ACL method, we have used three light field image quality datasets: VALID [153], Win5-LID [3], and LFDD [4]. We used SROCC and PLCC as performance evaluation methods. We compared the proposed NR LF-IQA method with the following state-of-art LF-IQA methods: SDFM [107], LFIQM [75], Tensor-NLFQ [79], GE-LFIQE [86], and ALAS-DADS [83]. We also compared the method with the following 2D-FR IQA methods [72, 74]: UQI, VIF, GMSD, NIQE, SSIM, IW-SSIM, IW-PSNR, FI-PSNR, MW-PSNR, MJ3DFR, PSNR-YUV and STMAD.

The horizontal EPI is fed to the 1st stream, while the vertical EPI is fed to the 2nd stream. Both inputs are used in RGB format. To train and test the model, we used data augmentation techniques, using horizontal and vertical flips operations. For training and testing, we divided each dataset into three content-independent sets: 60% for training, 20% for validation, and 20% for testing. To avoid biases, each set contains the reference LFI and all its corresponding distorted versions. For training, we used mini-batches of size 128, the Mean Square Error (MSE) as the training loss, and the Stochastic Gradient Descent (SGD) optimizer [118] with a learning rate of 0.0001. In total, the method was trained for 6,000 epochs

Table 6.1.1: The CNN-ACL network configuration.

Stream 1		**Stream 2**	
CNN Block			
Layer	Output	Layer	Output
input_1	(169, 626, 3)	input_2	(169, 626, 3)
Conv2D*	(169, 626, 32)	Conv2D	(169, 626, 32)
MaxPooling2D	(84, 313, 32)	MaxPooling2D	(84, 313, 32)
Conv2D*	(84, 313, 32)	Conv2D	(84, 313, 32)
MaxPooling2D	(42, 156, 32)	MaxPooling2D	(42, 156, 32)
Conv2D*	(42, 156, 64)	Conv2D	(42, 156, 64)
Conv2D*	(42, 156, 64)	Conv2D	(42, 156, 64)
Conv2D*	(42, 156, 128)	Conv2D	(42, 156, 128)
MaxPooling2D	(21, 78, 128)	MaxPooling2D	(21, 78, 128)
Reshape	(1638, 128)	Reshape	(1638, 128)
ACL Block			
AtrousConvolution1D	(1638, 1024)	AtrousConvolution1D	(1638, 1024)
Convolution1D*	(1638, 2)	Convolution1D	(1638, 2)
AtrousConvolution1D	(1638, 1024)	AtrousConvolution1D	(1638, 1024)
Convolution1D*	(1638, 2)	Convolution1D	(1638, 2)
AtrousConvolution1D	(1638, 1024)	AtrousConvolution1D	(1638, 1024)
Convolution1D*	(1638, 2)	Convolution1D	(1638, 2)
AtrousConvolution1D	(1638, 1024)	AtrousConvolution1D	(1638, 1024)
Convolution1D*	(1638, 2)	Convolution1D	(1638, 2)
Add	(1638, 2)	Add	(1638, 2)
Concatenation Layer			
Concatenate		(1638, 4)	
Regression Block			
Flatten		(6552)	
Dense*		(2560)	
Dropout Layer			
Dense		(2560)	
Dense		(1)	

and the model with minimum validation loss was reported. We implemented the proposed method using Keras [190] library of Python. The method was trained and tested on 25GB GPU, with a LINUX environment. The code of the proposed LF-IQA method is available for download on GitHub[1].

6.1.5 Experimental Results

Table 6.1.2 show the results of tests performed. The rows in this table show the results for each distortion of a dataset (groups of columns), with the 'All' row corresponding to the results obtained for the complete datasets. Notice that the proposed method performs very well in all datasets, reaching SROCC values over 0.96 and PLCC values over 0.97 for the VALID

[1] `https://bit.ly/3B67bOo`

Table 6.1.2: The SROCC and PLCC values for VALID, LFDD, and Win5-LID datasets.

LFDD			**VALID**			**Win-5LID**		
Distortion	SROCC	PLCC	Distortion	SROCC	PLCC	Distortion	SROCC	PLCC
AV1	0.8074	0.9305	10bit_HEVC	0.9716	0.9796	EPICNN	0.9600	0.9600
BAR	0.7092	0.8502	10bit_P3	0.9784	0.9695	HEVC	0.9637	0.9896
BPG	0.9492	0.9692	10bit_P5	0.9690	0.9867	JPEG2000	0.9623	0.9897
Gaussian	0.9819	0.8824	10bit_VP9	0.9684	0.9893	LN	0.9764	0.9880
JPEG2000	0.4800	0.6570	8bit_HEVC	0.97176	0.9738	NN	0.9818	0.9861
JPEG	0.9819	0.9610	8bit_HEVC	0.9716	0.9872	QD	0.9600	0.9600
Pincushion	0.3491	0.4773	8bit_VP9	0.9716	0.9795	-	-	-
Impulse	0.9819	0.7913	-	-	-	-	-	-
Unsharp Mask	0.6873	0.7923	-	-	-	-	-	-
VP9	0.9492	0.8973	-	-	-	-	-	-
x264	0.8619	0.8561	-	-	-	-	-	-
x265	0.7746	0.6277	-	-	-	-	-	-
ALL	0.7928	0.8077	ALL	0.9733	0.9808	ALL	0.9688	0.9789

Table 6.1.3: SROCC and PLCC values obtained for state-of-the-art LF-IQA methods tested on LFDD, VALID, and Win5-LID datasets.

Category	Type	Methods	Year	LFDD		VALID		Win5-LID	
				SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Pre-defined Functions	FR	UQI	2002	0.4673	0.3486	0.9310	0.9550	0.8252	0.8764
	FR	SSIM	2004	0.4488	0.2457	0.9500	0.9640	0.6812	0.7880
	FR	VIF	2006	0.4588	0.4026	0.9620	0.9790	0.9347	0.9555
	FR	NIQE	2009	0.5235	0.4138	0.6211	0.6544	0.4892	0.5002
	FR	STMAD	2011	0.2054	0.2005	0.7940	0.8020	0.8489	0.9074
	FR	IW-SSIM	2011	0.4432	0.2594	0.9650	0.9780	0.8212	0.8736
	FR	IW-PSNR	2011	0.4184	0.3060	0.9470	0.9670	0.8842	0.9022
	FR	MJ3DFR	2013	0.4235	0.3182	0.9560	0.9700	0.8836	0.8998
	FR	GMSD	2014	0.4384	0.4000	0.6821	0.6948	0.4352	0.5041
	FR	FI-PSNR	2014	0.2415	0.1645	0.7060	0.7060	0.6951	0.7419
	FR	PSNR-YUV	2014	0.4325	0.4124	0.9230	0.9310	0.9007	0.9215
	FR	MW-PSNR	2016	0.4021	0.3842	0.6869	0.6904	0.7582	0.7758
	RR	LFIQM [75]	2019	0.1245	0.1041	0.3934	0.5001	0.4503	0.4763
	FR	SDFM [107]	2020	-	-	0.8240	0.8542	0.6742	0.7142
SVR-based	NR	Tensor-NLFQ [79]	2019	0.5134	0.4124	0.8702	0.9028	0.9101	0.9217
CNN-based	NR	GELFIQE [86]	2021	-	-	0.9442	0.9753	-	-
	NR	DELFIQE [86]	2021	-	-	0.9260	0.9749	-	-
	NR	ALAS-DADS [83]	2021	-	-	-	-	0.9260	0.9257
	NR	**Proposed**	2022	**0.7928**	**0.8077**	**0.9733**	**0.9808**	**0.9688**	**0.9789**

and Win-5LID datasets. For these 2 datasets, the correlation values across different distortions are all above 0.96. On the other hand, notice that the SROCC and PLCC values obtained for the LFDD dataset are lower (0.79 and 0.80, respectively, for the 'All' case). It is worth mentioning that the sub-aperture images in this dataset have the lowest resolution, which might explain this difference in performance. In fact, for 3 out of the 12 distortions in the LFDD dataset the correlation values are below 0.5, while for 5 out of the 12 distortions they are below 0.7.

Figure 6.1.4 shows the scatter plots of the subjective quality scores versus predicted quality scores obtained for the LFDD, VALID and Win5-LID LFI quality datasets. It is worth men-

tioning that the MOS ranges for each dataset may be different since different experimental methodologies were used to collect the quality scores. We decided not to normalize the MOS values since a previous study demonstrated that normalizing subjective scores into standard values does not significantly improve the quality predictions [74]. Even though no normalization was performed, points in the graphs in Figure 6.1.4 show very good fitting results for Win5-LID and VALID datasets, indicating that the proposed LF-IQA method is able to predict the quality of LF contents accurately with a small variation in the subjective scores. But for LFDD, we observe different curve showing a large variation across the distribution of data points.

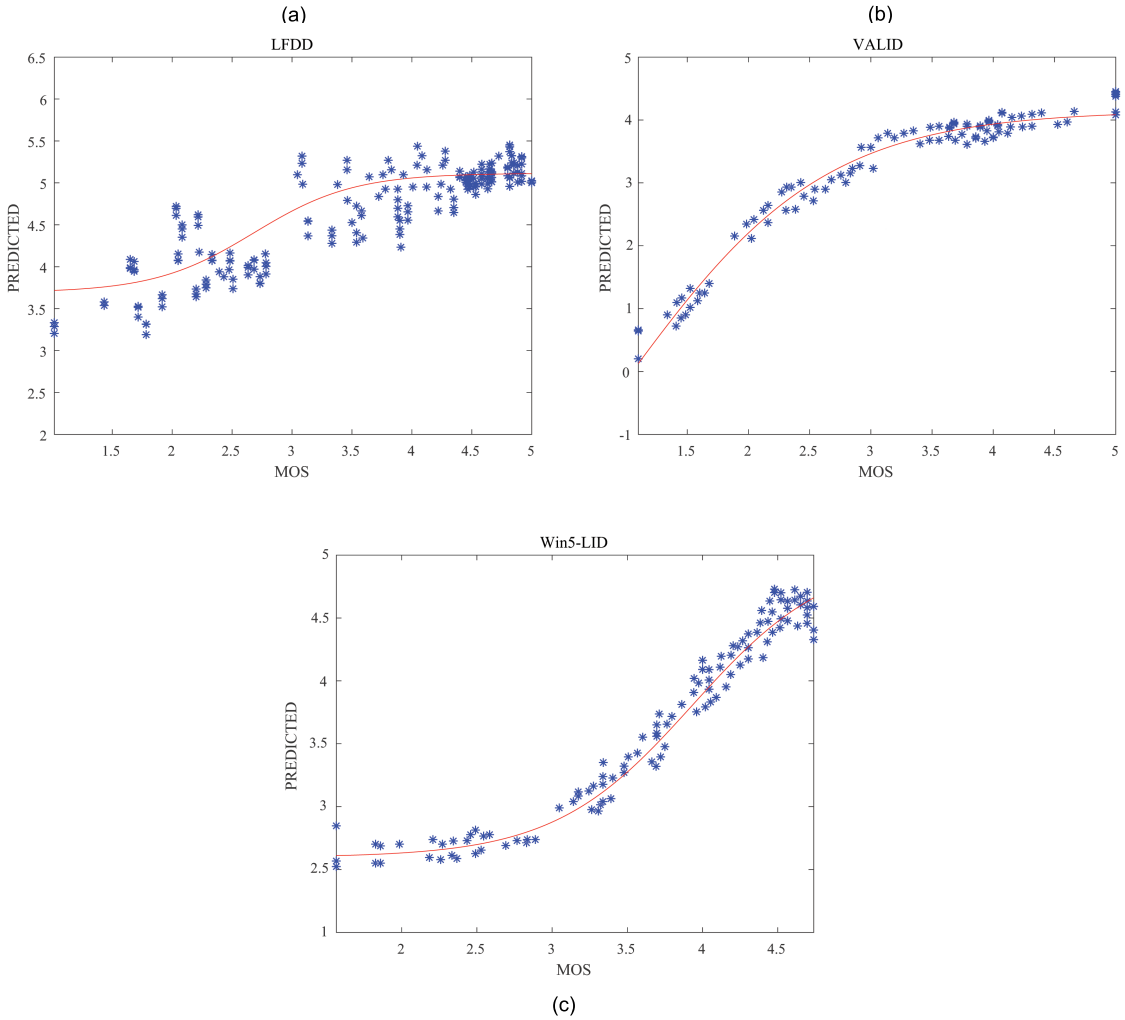


Figure 6.1.4: Scatter plots of subjective quality scores versus predicted quality scores. (a) LFDD, (b) VALID, and (c) Win5-LID.

Table 6.1.3 shows a comparison of the results with other state-of-the-art LF-IQA methods. In this table, results are separated by the type of IQA method (No-Reference - NR, Reduced-Reference - RR, and Full-Reference - FR) and the type of models used to pool fea-

tures into quality estimates (pre-defined functions, ML based algorithms - LR or SVR, or CNN approaches). For simplicity, only the overall performance ('ALL') correlation values are reported for each dataset. The line for SDFM has two blank cells because the authors do not provide results for the LFDD dataset. Notice that the proposed method has the highest SROCC and PLCC values among all tested metrics for all three LF-IQA datasets. As observed earlier, correlation values for the LFDD dataset are lower, not only for the proposed method but for all tested metrics.

To test the robustness of the proposed NR LF-IQA method in the presence of unseen contents and distortions, we performed a cross-database evaluation. To perform this test, we trained the proposed model on one dataset and tested on a different one. Given the lower results achieved for the LFDD dataset, in this test we only used the VALID and Win5-LID datasets for training, while the three datasets were used for test. Table 6.1.4 shows the SROCC and PLCC values for this evaluation. Results show that the proposed NR LF-IQA method is robust and consistent across different contents. In fact, when the method is trained on the Win-5LID and tested on the LFDD dataset, SROCC and PLCC values for the LFDD dataset are significantly higher than the values shown in Table 6.1.3.

Table 6.1.4: Summary of cross-database evaluation results (SROCC and PLCC) for different train–test dataset combinations.

Training Dataset	Testing Dataset	SROCC	PLCC
VALID	Win5-LID	0.9546	0.9663
	LFDD	0.7774	0.7919
Win5-LID	VALID	0.9698	0.9722
	LFDD	0.8383	0.8399

Table 6.1.5: Comparison of proposed model (combination) with 4 CNN-ACL models with only one Atrous rates (6, 12, 18, 2). Training/Test performed on the Win5-LID dataset.

Dataset	Atrous Rate	SROCC	PLCC
Win5-LID	1	0.3154	0.4889
	6	0.4121	0.4283
	12	0.3903	0.3977
	18	0.4904	0.5092
	24	0.3546	0.3552
	Proposed: 6+12+18+24	**0.9688**	**0.9789**

Finally, we analyzed if a combination of CNN-ACLs with different Atrous rates performed better than using one CNN-ACL with a single Atrous rate. For this, we split the ACL block of the original model so that each training model has only one CNN-ACL stage. Then, we tested the four individual models: model 1 with Atrous rate equal to 6, model 2 with Atrous rate equal to 12, model 3 with Atrous rate equal to 18, and model 4 with Atrous rate equal

to 24. We also performed a second test in which we used the proposed method with the entire ACL block, but the Atrous rate is set to 1 in every stage. We performed these tests using only the Win5-LID dataset. Table 6.1.5 shows the SROCC and PLCC values obtained that are significantly lower than the results shown in Table 6.1.3. In other words, using a model with a combination of four Atrous rates provides a better performance than using a single model with a specific Atrous rate.

6.2 Diverse Neural Network for Quality Assessment of Complex LF Images (ACL-LSTM)

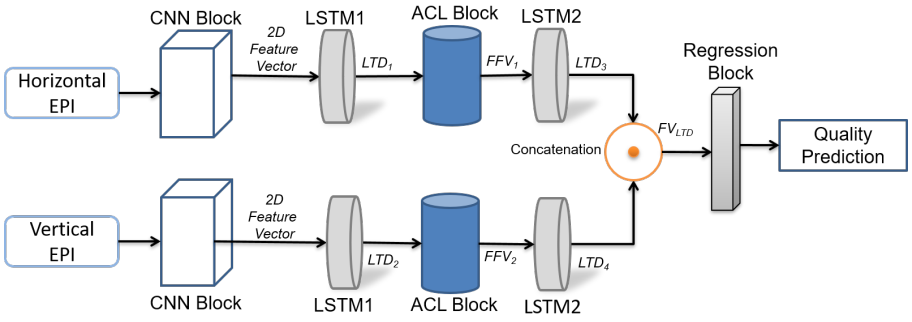


Figure 6.2.1: Block Diagram of the proposed ACL-LSTM method.

Figure 6.2.1 shows the block-diagram of the proposed ACL-LSTM method of NR LF-IQA. Notice that the method has two streams, each composed of an independent CNN, ACL, and LSTM layers, taking as input either the horizontal or vertical EPIs. Both streams first extract primary features of inputs using CNNs and, then, extract high-level features using the block of ACLs. LSTM layers are used to extract long-term dependent distortion related features from the ACL output features. The horizontal EPI is fed to the 1st stream, while the vertical EPI is fed to the 2nd stream. Finally, the output from both streams is concatenated and fed to the regression block that estimates the LFI quality score.

6.2.1 CNN Block

Figure 6.2.2 shows the CNN block, which contains six stages that extract primary features from an input RGB image. Stages 1, 2, and 5 are identical, with the first layer being a 2D convolution layer with 32 output filters and a 3×3 kernel, the second layer an ELU activation layer, and the third layer a 2D max-pooling layer with a 2×2 pool and a 2×2 stride. Stages 3 and 4 are identical, with the first layer being a 2D convolution with 64 output filters and a 3×3 kernel and the second layer being an ELU activation layer. Stage 6 contains a reshape

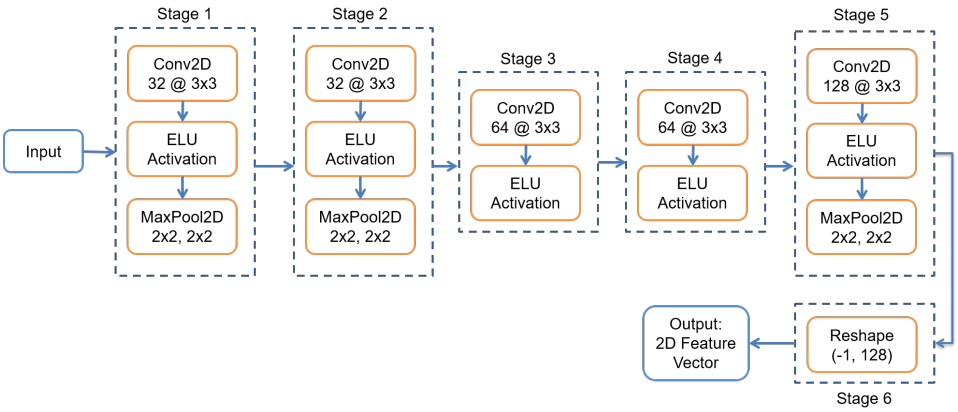


Figure 6.2.2: Illustration of CNN Block in ACL-LSTM Method.

layer that takes a 3D input and converts it into a 2D feature vector. To keep the output of the CNN block consistent for both streams, we have used a fixed size of 128.

6.2.2 ACL Block

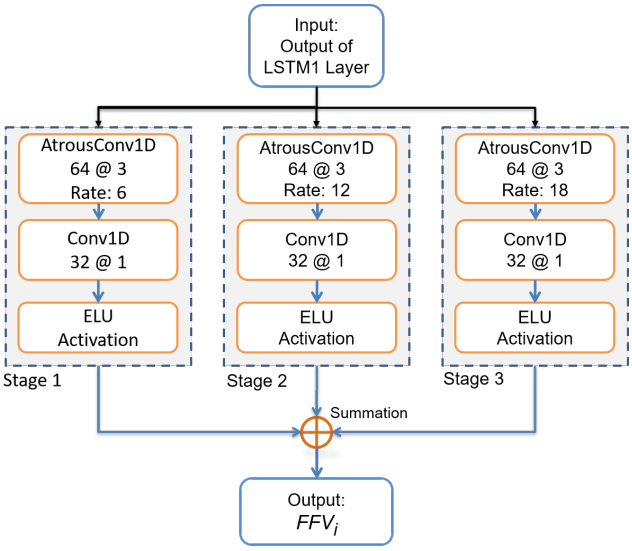


Figure 6.2.3: Illustration of ACL Block in ACL-LSTM Method.

As mentioned earlier, ACLs can be used to effectively enlarge the receptive field of the kernels and capture dense spatial and angular features, without increasing the number of parameters. More specifically, ACL is a dilated convolution where zeroes are added between the weights of the convolution kernel. We used same 1D Atrous convolution layer as computed using equation 6.1.1.

Figure 6.2.3 illustrates the architecture of the proposed ACL block that consists of 3 stages. Each stage is composed of a 1D Atrous Convolution layer with 64 output filters and a kernel of length 3, a 1D convolution layer with 32 output filters and a kernel of length 1, and an ELU activation layer. To keep the ACLs distinct in every stage, we use three dilation rates: 6, 12 and 18 [119]. From the ACL block, we obtain a fused feature vector by the summation of the outputs y_i from each i-th stage, as follows:

$$FFV_i = y_1 + y_2 + y_3 + y_4, \tag{6.2.1}$$

where the i index corresponds to the stream ($i = 1,2$) of the proposed method (see Figure 6.2.1).

6.2.3 LSTM Block

In ACL-LSTM method, LSTM block consists of two layers of LSTM network (LSTM1 and LSTM2), taking advantage from its memory cells that extracts long-term dependencies and relationship among distortion-related (LTD) features. First LSTM layer occurs after the CNN block, while second layer occurs after the ACL block (see Figure 6.2.1). In this work, we have used 5 recursive memory units in both layers of LSTM network.

6.2.4 Regression Block

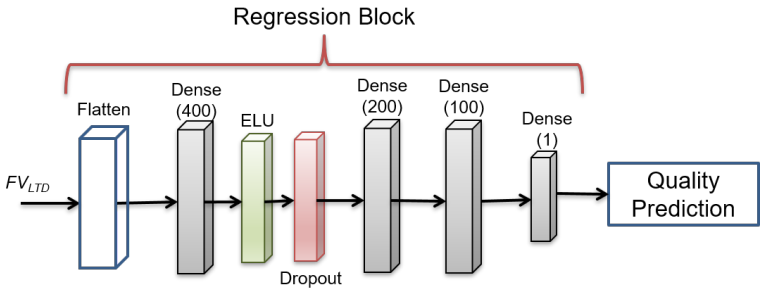


Figure 6.2.4: Illustration of Regression Block in ACL-LSTM Method.

As shown in Figure 6.2.1, the output of LSTM2 layers of both streams (LTD_3 and LTD_4) is concatenated as follows:

$$FV_{LTD} = LTD_3 \oplus LTD_4, \tag{6.2.2}$$

where FV_{LTD} represents the concatenated feature vector and '$\oplus$' represents the concatenation operation. The concatenated feature vector is fed to the regression block, which is composed by one Flatten layer and four Dense layers, as shown in Figure 6.2.4. The first Dense

layer has 400 features and is followed by the ELU and Dropout layers. The second Dense layer has 200 features and is followed by third Dense layer with 100 features. The last Dense layer generates a one-dimensional scalar number that corresponds to the estimated perceptual quality score. Table 6.2.1 lists the concrete network configuration of ACL-LSTM, in which, layers marked with a * symbol are followed by one ELU activation layer. Next, we describe each of the stages in Figure 6.2.1.

Table 6.2.1: The ACL-LSTM network configuration.

Stream 1		**Stream 2**	
CNN Block			
Layer	Output	Layer	Output
input_1	(81, 512, 3)	input_2	(81, 512, 3)
Conv2D*	(81, 512, 32)	Conv2D	(81, 512, 32)
MaxPooling2D	(40, 256, 32)	MaxPooling2D	(40, 256, 32)
Conv2D*	(40, 256, 32)	Conv2D	(40, 256, 32)
MaxPooling2D	(20, 128, 32)	MaxPooling2D	(20, 128, 32)
Conv2D*	(20, 128, 64)	Conv2D	(20, 128, 64)
Conv2D*	(20, 128, 64)	Conv2D	(20, 128, 64)
Conv2D*	(20, 128, 128	Conv2D	(20, 128, 128)
MaxPooling2D	(10, 64, 128)	MaxPooling2D	(10, 64, 128)
Reshape	(640, 128)	Reshape	(640, 128)
LSTM	(640, 5)	LSTM	(640, 5)
ACL Block			
AtrousConvolution1D	(640, 64)	AtrousConvolution1D	(640, 64)
Convolution1D*	(640, 32)	Convolution1D	(640, 32)
AtrousConvolution1D	(640, 64)	AtrousConvolution1D	(640, 64)
Convolution1D*	(640, 32)	Convolution1D	(640, 32)
AtrousConvolution1D	(640, 64)	AtrousConvolution1D	(640, 64)
Convolution1D*	(640, 32)	Convolution1D	(640, 32)
Add	(640, 32)	Add	(640, 32)
LSTM	(640, 5)	LSTM	(640, 5)
Concatenation Layer			
Concatenate		(640, 10)	
Regression Block			
Flatten		(6400)	
Dense*		(400)	
Dropout Layer			
Dense		(200)	
Dropout Layer			
Dense		(100)	
Dense		(1)	

6.2.5 Experimental Setup

To train and test the proposed ACL-LSTM method, we have used two LFI datasets LFDD [4], and SMART [156] which contain complex LF images. We used SROCC and PLCC as per-

formance evaluation methods. We compared the proposed NR LF-IQA method with the following state-of-art LF-IQA methods: SDFM [107], LFIQM [75], Tensor-NLFQ [79], GE-LFIQE [86], DE-LFIQE [85], and ALAS-DADS [83]. We also compared the method with the following 2D-FR IQA methods [72, 74]: UQI, VIF, GMSD, NIQE, SSIM, IW-SSIM, IW-PSNR, FI-PSNR, MW-PSNR, MJ3DFR, PSNR-YUV and STMAD.

The horizontal EPI is fed to the 1st stream, while the vertical EPI is fed to the 2nd stream. Both inputs are used in RGB format. To train and test the model, we used data augmentation techniques, using horizontal and vertical flips operations. For training and testing, we divided each dataset into three content-independent sets: 80% for training, 10% for validation, and 10% for testing. To avoid biases, each set contains the reference LFI and all its corresponding distorted versions. For training, we used mini-batches of size 128, the Mean Square Error (MSE) as the training loss, and the Stochastic Gradient Descent (SGD) optimizer [118] with a learning rate of 0.0001. In total, the method was trained for 6,000 epochs and the model with minimum validation loss was reported. We implemented the proposed method using Keras [190] library of Python. The method was trained and tested on 25GB GPU, with a LINUX environment. The code of the proposed LF-IQA method is available for download on GitHub[2].

6.2.6 Experimental Results

Table 6.2.2 show the results of tests performed. The rows in this table show the results for each distortion of a dataset (groups of columns), with the 'All' row corresponding to the results obtained for the complete datasets. Notice that the SROCC and PLCC values obtained for the LFDD dataset (complex LF images) are 0.80 and 0.84 respectively. For this dataset, very few distortions, such as JPEG and JPEG2000, have shown lower correlation values. On the other hand, the SROCC and PLCC values obtained for the SMART dataset (less complex LF images) are 0.90 and 0.93 respectively, for the 'All' case. For this dataset, the distortion JPEG2000 has obtained 0.90 and 0.92 SROCC and PLCC values, but the distortion HEVC has obtained 0.95 PLCC value, with lower SROCC value that is 0.89.

Figure 6.2.5 shows scatter plots of subjective quality scores versus predicted quality scores obtained for LFDD, and SMART LFI quality datasets. It is worth mentioning that the MOS ranges for each dataset may be different since different experimental methodologies were used to collect the quality scores. We decided not to normalize the MOS values since a previous study demonstrated that normalizing subjective scores into standard values does not significantly improve the quality predictions [74]. Even though no normalization was performed, points in the graphs in Figure 6.2.5 show good fitting results for LFDD and SMART

[2] https://bit.ly/3wNPliU

Table 6.2.2: The SROCC and PLCC values for LFDD, and SMART datasets.

Dataset	Distortion	PROPOSED	
		SROCC	PLCC
LFDD	AV1	0.8113	0.8963
	BAR	0.8418	0.8654
	BPG	0.9492	0.8130
	Gaussian	0.8054	0.8087
	JPEG2000	0.8783	0.6570
	JPEG	0.6831	0.5967
	Pincushion	0.7419	0.7870
	Impulse	0.7773	0.9876
	Unsharp Mask	0.8420	0.8628
	VP9	0.7557	0.7983
	x264	0.8783	0.9110
	x265	0.8783	0.9659
	ALL	0.8085	0.8448
SMART	HEVC	0.8946	0.9519
	JPEG2000	0.9061	0.9234
	ALL	0.9004	0.9375

datasets. Specifically for LFDD dataset, when compared with previous CNN-ACL method, the method ACL-LSTM has obtained better predictions in accordance with subjective scores.

Table 6.2.3 shows a comparison of the results with other state-of-the-art LF-IQA methods. For simplicity, only the overall performance ('ALL') correlation values are reported for each dataset. The lines for SDFM, ALAD-DADS, GELFIQA, and DELFIQE has two blank cells because these methods have not used LFDD dataset with complex LF images. Notice that the correlation values for the LFDD dataset are lower for all comparison methods, but the proposed has obtained the stand-out performance. For SMART dataset, our method has obtained the highest PLCC value, while other LF-IQA method which is Tensor-NLFQ, has shown the highest SROCC value.

To test the robustness of the proposed NR LF-IQA method in the presence of unseen contents and distortions, we performed a cross-database evaluation. To perform this test, we trained the proposed model on one dataset and tested on a different one. Since our focus is to provide better quality evaluation of complex LF images, in this test, we used the SMART dataset for training because it has obtained higher correlation values, while the LFDD dataset was used for test. Table 6.2.4 shows the SROCC and PLCC values for this evaluation. Results show that the proposed NR LF-IQA method is robust and consistent across different contents. The dataset SMART has less complex contents, with compression-related distortions only. From the results of this experiment, it can be concurred that our model trained on simple LF images and their subjective scores, become useful application for complex LF images to provide better quality of experience.

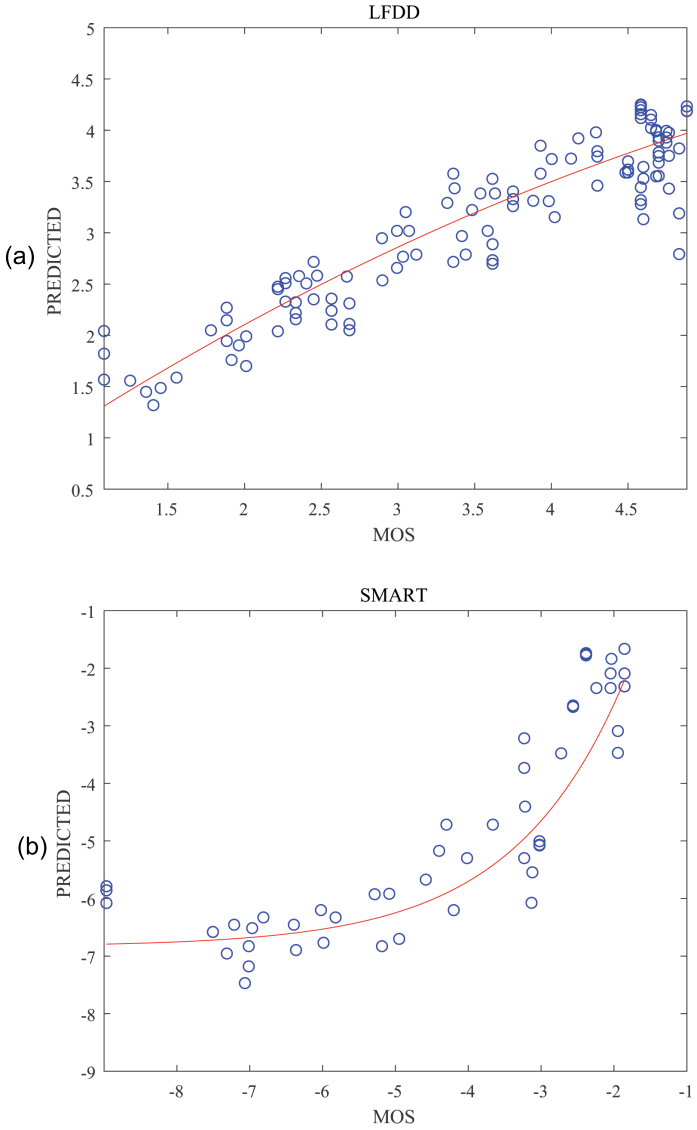


Figure 6.2.5: Scatter plots of subjective quality scores versus predicted quality scores. (a) LFDD, and (b) SMART.

Finally, we analyzed if a combination of ACLs with different Atrous rates, and LSTMs performed better than using one ACL with a single Atrous rate, without LSTM layers. For this, we split the ACL block of the original model so that each training model has only one ACL stage. Then, we tested three individual models: model 1 with Atrous rate equal to 6, model 2 with Atrous rate equal to 12, and model 3 with Atrous rate equal to 18. We performed a second test in which we used the proposed method without LSTM1 and LSTM2 layers in both streams. We also performed a third test in which we used the proposed method with the entire ACL block, but the Atrous rate is set to 1 in every stage. We performed these tests using only the LFDD dataset. Table 6.2.5 shows the SROCC and PLCC values obtained that are significantly lower than the results shown in Table 6.2.3. In other words, using a model,

Table 6.2.3: SROCC and PLCC values obtained for state-of-the-art LF-IQA methods tested on LFDD, and SMART datasets.

Category	Type	Methods	Year	LFDD		SMART	
				SROCC	PLCC	SROCC	PLCC
Based on Pre-defined Functions	FR	UQI	2002	0.4673	0.3486	0.6480	0.7980
	FR	SSIM	2004	0.4488	0.2457	0.7550	0.8010
	FR	VIF	2006	0.4588	0.4026	0.7260	0.8370
	FR	STMAD	2011	0.2054	0.2005	0.6604	0.8010
	FR	IW-SSIM	2011	0.4432	0.2594	0.8060	0.8850
	FR	IW-PSNR	2011	0.4184	0.3060	0.7840	0.8520
	FR	MJ3DFR	2013	0.4235	0.3182	0.8160	0.8480
	FR	GMSD	2014	0.4384	0.4000	0.8520	0.8700
	FR	FI-PSNR	2014	0.2415	0.1645	0.7730	0.8320
	FR	PSNR-YUV	2014	0.4325	0.4124	0.9102	0.9211
	FR	MW-PSNR	2016	0.4021	0.3842	0.6893	0.7241
	RR	LFIQM [75]	2019	0.1245	0.1041	0.4503	0.4763
	FR	SDFM [107]	2020	-	-	0.7514	0.7941
ML-based	NR	Tensor-NLFQ [79]	2019	0.5134	0.4124	**0.9124**	0.8988
CNN-based	NR	ALAS-DADS [83]	2021	-	-	0.8540	0.9344
	NR	GELFIQE [86]	2021	-	-	0.8851	0.9156
	NR	DELFIQE [85]	2021	-	-	0.8812	0.9149
	NR	**Proposed**	2021	**0.8085**	**0.8448**	0.9004	**0.9375**

Table 6.2.4: Summary of experimental results (SROCC and PLCC) for train–test combinations of different pair of legacy LF-IQA datasets.

Training Dataset	Testing Dataset	SROCC	PLCC
SMART	LFDD	0.8165	0.8249

Table 6.2.5: Ablation Test Results: SROCC and PLCC values for LFDD dataset.

Dataset	Test Type	SROCC	PLCC
LFDD	Model1: Atrous Rate = 6	0.1091	0.0581
	Model2: Atrous Rate = 12	0.4909	0.5824
	Model3: Atrous Rate = 18	0.4146	0.4219
	Method in Figure 6.2.1, but without LSTM layers	0.7861	0.7990
	Method in Figure 6.2.1, but with Atrous Rate = 1 for all stages in ACL Block	0.0840	0.1044
	Proposed Method	**0.9004**	**0.9375**

with a combination of three Atrous rates, and LSTM layers, provides a better performance than using a single model with a specific Atrous rate or without LSTM layers.

6.3 Conclusion

To explore dense features of the EPIs, we have proposed two novel NR LF-IQA methods, in which, the first method uses CNN with Atrous Convolution layers (CNN-ACL), while the second method use Atrous Convolution Layers with LSTM layers (ACL-LSTM). In CNN-ACL method, the architecture is composed of two streams, each containing CNN-ACL layers that extract spatial and angular features from the horizontal and vertical EPIs. The feature vectors obtained from each CNN-ACL stream, with different Atrous rates, are concatenated and fed to the regression block for quality prediction.

In ACL-LSTM method, we aimed to achieve better performance in terms of quality prediction for complex LF image datasets, such as LFDD and SMART, because the method CNN-ACL did not perform well on such complex datasets. The ACL-LSTM method is based on a diverse neural network that includes CNN, ACL with three variants of Atrous rates, and LSTM layers. More specifically, the model architecture is composed of two streams, each containing CNN, ACL, and LSTM layers that extract spatial and angular features from the horizontal and vertical EPIs. The feature vectors obtained from each stream, are concatenated and fed to the regression block for quality prediction.

For both CNN-ACL and ACL-LSTM methods, results show that these methods outperformed state-of-the-art methods on popular LFI datasets. We noticed that the ACL-LSTM method obtained better correlations for LFDD than the ACL-CNN method. We also performed a cross-database evaluation, with results showing that the method is robust for different scenarios. Finally, we compared the proposed method with methods using a single ACL layer with a specific Atrous rate. Results show that the combination of 4 ACL layers with different Atrous rates performs better.

Chapter 7

LF-IQA Method Based on Deep Graph Convolutional Neural Network

In this chapter, we propose a no-reference LF-IQA method that predicts the quality of compressed LF images using a Deep Graph Convolutional Neural Network (GCNN-LFIQA). The GCNN-LFIQA method is based on a deep single-stream network architecture which takes horizontal EPI as input assuming that the data is unordered and irregular. This way, the method identifies and learns from unstructured data for quality prediction.

7.1 Methodology

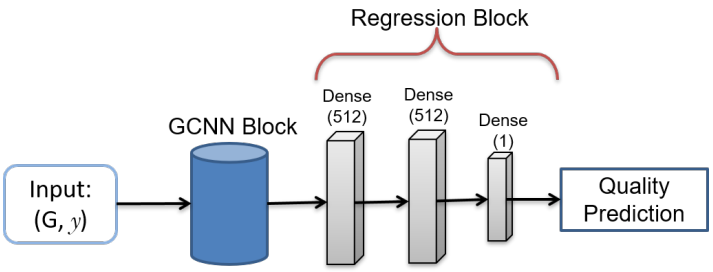


Figure 7.1.1: Block Diagram of the proposed Graph Convolutional Neural Network-based Light Field image quality assessment (GCNN-LFIQA)

Figure 7.1.1 shows the block-diagram of the proposed method, in which, first layer is the input layer (G, y), where G represents graphs (nodes and edges information) and y represents the subjective quality score as target labels. The input is passed to a Graph Convolutional Neural Network (GCNN) block and then to the regression block. The regression block includes three Dense layers, in which first two Dense layers produce output feature vector of

size 512 and the last Dense layer produces a scalar number that represents predicted quality score of the corresponding input LF content. Next, we describe input preparation, and architecture of the GCNN block.

7.1.1 Input Preparation

To prepare the graphs, we have used horizontal EPIs. EPI format consumes low computational resources and, most importantly, they allow the incorporation of comprehensive and distortion-related features of a 4D LFI. To prepare the graphs, we follow the steps described next:

1. First, we generate Canny [200] edge map from RGB format of horizontal EPIs;
2. Second, we convert edge map to a binary image;
3. Third, we create skeleton image in boolean format from a binary image using a fast parallel algorithm for thinning digital patterns [201];
4. Finally, we compute the graphs based on every pixel value in skeleton image. Every true pixel becomes node, while false pixels are discarded.

It is worth pointing out that, our purpose of converting Canny edge map into skeleton image is to reduce the number of nodes and edges, keeping only the important data. The skeleton image helps finding the connected components in a boolean image, where the foreground has 1s as pixel values and the background has 0s as pixel values. We compute the neighbors of every pixel, which are connected with each other through 8-pixels connectivity. This way, we obtain a direct graph G with nodes and edges information. Figure 7.1.2 shows an illustration of input preparation. Specifically, Figure 7.1.2(a) shows a horizontal EPI distorted by JPEG distortions, while Figure 7.1.2(b) shows a horizontal EPI distorted by HEVC distortions. Both EPIs are extracted from the Swans LFI taken from the Win5-LID [3] LF-IQA dataset. In this figure, we can see a clear graph difference for the same LF content distorted with different types of artifacts.

It is important to extract important node features / attributes, that can be helpful for a graph convolutional neural network to learn and make predictions. In this work, we consider the artificial node feature known as Betweenness Centrality (CB) [202]. The Betweenness centrality of a node n is the sum of the fraction of all-pairs shortest paths that pass through n:

$$CB = \sum_{c,b \in N} \left[\frac{\sigma(c,b|N)}{\sigma(c,b)} \right], \tag{7.1.1}$$

where N is the set of nodes, $\sigma(c,b)$ is the number of shortest (c,b) paths, and $\sigma(c,b|N)$ is the number of those paths passing through some node $n \in N$ other than c,b. After preparing

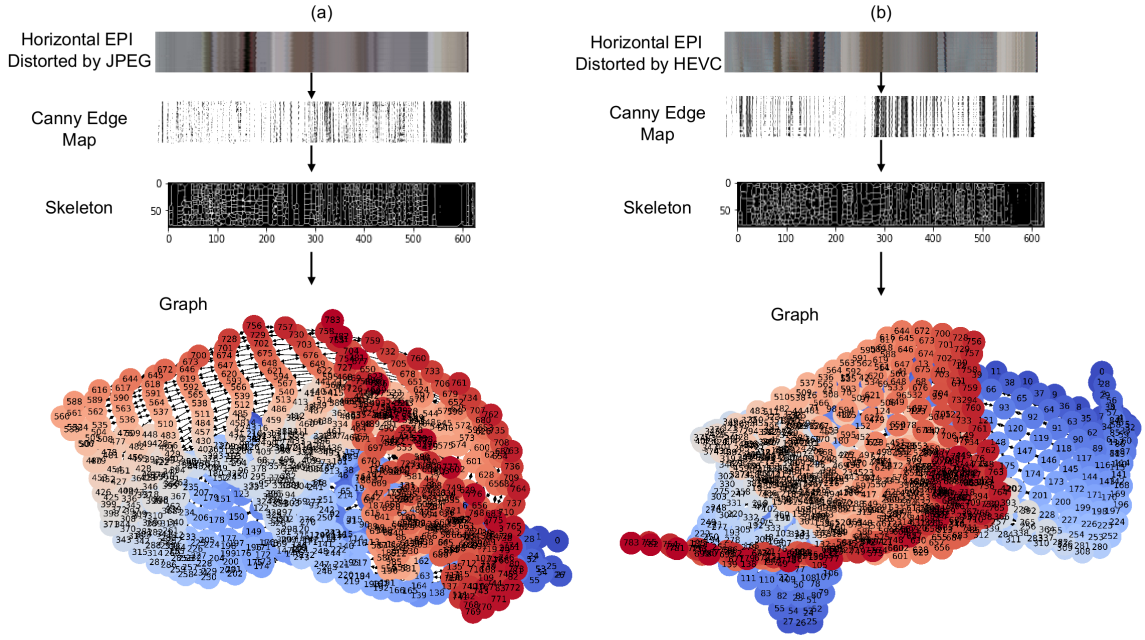


Figure 7.1.2: Illustration of input preparation which includes transformation of horizontal EPI from Canny edge map, to skeleton, and then its graph representation. EPIs are generated from two distorted LFIs taken from Win5-LID [3] LF-IQA dataset: (a) RGB format of horizontal EPI distorted by JPEG distortion, Canny edge map, skeleton, and then its graph representation, and (b) RGB format of horizontal EPI distorted by HEVC distortion, Canny edge map, skeleton, and then its graph representation.

these features, we obtain a feature vector of size 1×1 for every node in a graph. Algorithm 1 illustrates step-by-step instructions for preparing graphs from skeleton image.

7.1.2 Graph Convolutional Neural Network Block

Since every image generates a graph with different number of nodes and edges, traditional CNN layers cannot process multi-dimensional inputs. To resolve this problem, in this work, we have adapted a Graph Convolutional Neural Network (GCNN) proposed by Zhang *et. al.* [130]. The main advantage of GCNN is that it can retain much more complex node information and learn from the global graph topology. Most importantly, GCNN consists of a SortPooling layer, which generates ordered input from an unordered node features of graph convolutions. Instead of summing of these node features, SortPooling arranges them in a consistent order and generates as output a sorted graph representation with a fixed size. In this way, the traditional convolutional neural networks can read nodes in a consistent order and be trained on this representation. In other words, the SortPooling layer acts as a bridge between graph convolution layers and traditional neural network layers. As a bridge

Algorithm 1 An algorithm to create directed graph from skeleton image.

Input: Skeleton image img_m^n of size $n \times m$ in boolean format.

Initialize directed graph g.
$neighbors \leftarrow 6 \times 2 \times 1$ permutations of list $[-1, 0, 1]$
$nIdx \leftarrow length(neighbors)$
while $n, m \neq 0$ **do**
    while $nIdx \neq 0$ **do**
        $nextPos_c^p = neighbors$ at position $nIdx$ ▷ Obtain indices of the neighbors.
        if pixels in img_m^n at positions $nextPos_c^p$ = True **then**
            $node1 \leftarrow img_m^n$
            $node2 \leftarrow img_m^n | m + nextPos_c, n + nextPos^p$
            $edge \leftarrow node1$ connected to $node2$
            $weight_{edge} \leftarrow \|edge\|$ ▷ Compute Norm.
            Add $node1$, $node2$, $weight_{edge}$ in graph g
        end if
    end while
end while
$CB \leftarrow centrality_{betweenness}(g)$
$g \leftarrow CB$

Output: g

between graph convolution layers and traditional layers, it can pass loss gradients back to the previous layers by remembering the sorted order of its input. This way, the training of the parameters of previous layers becomes feasible. One GCNN layer works as follows: (a) it extracts local substructure features of nodes and defines a consistent node ordering, and (b) a SortPooling layer sorts the node features under pre-set order and unifies input sizes.

To extract multi-scale substructure features, in this work, we stack $t = 5$ GCNN layers as follows [203]:

$$\mathbf{Z}^{t+1} = f(\tilde{\mathbf{D}}^{-1}\tilde{\mathbf{A}}\mathbf{Z}^t\mathbf{W}^t) \tag{7.1.2}$$

where $\mathbf{Z}^0$ represents node attributes, $\mathbf{Z}^t$ is the output of t^{th} GCNN layer, $\tilde{\mathbf{A}}$ represents the adjacency matrix of the graph, $\tilde{\mathbf{D}}^{-1}$ represents diagonal degree matrix of the graph, $\mathbf{W}$ represents a matrix of trainable graph convolution parameters, f represents a non-linear activation function, and $\mathbf{W}^t$ maps the node attributes of t^{th} GCNN layer to the node attributes in $t+1^{\text{th}}$ GCNN layer.

A concatenation operation is performed by SortPooling layer to horizontally stack the outputs of all GCNN layers. The final output is represented as $\mathbf{Z}^{1:t} = [\mathbf{Z}^1, ..., \mathbf{Z}^t]$. After SortPooling, we get a tensor $\mathbf{Z}^{\mathbf{sp}}$ of size $k \times \sum_1^t c_t$, where each row represents a node, each column represents the attributes of the corresponding node, and t represents the number of GCNN layer. The output tensor $\mathbf{Z}^{\mathbf{sp}}$ is reshaped into $k(\sum_1^t c_t) \times 1$ vector row-wise, where c_t represents the number of output channels of the GCNN layer t. Then, a one-dimensional (1D)

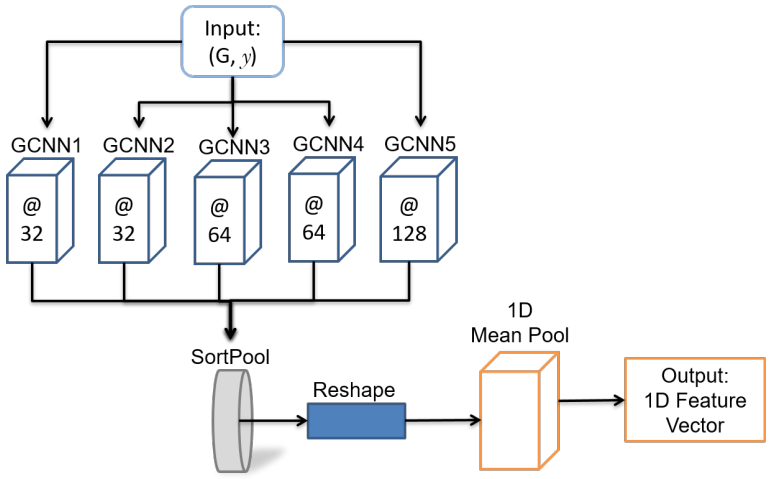


Figure 7.1.3: Block diagram of GCNN block in the proposed LF-IQA method. An input graph of arbitrary structure is first passed through multiple graph convolution layers, where node information is propagated between neighbors. Then the node features are sorted and pooled with a SortPooling layer, and passed to traditional 1D Mean Pooling layer in order to learn local patterns on node sequence.

mean pooling layer (GlobalAveragePooling) is added after the last GCNN operation in order to learn local patterns on the node sequence. In this work, the output of the first and second GCNN layers is 32 features, while the output of the third and fourth GCNN layers is 64 features. The last GCNN layer generates output features of length 128. It is worth pointing out that every GCNN layer is followed by a Dropout layer to prevent overfitting. Figure 7.1.3 shows block diagram of GCNN block of the proposed method.

7.2 Experimental Setup

To train and test the proposed method, we have used one light field image quality dataset Win5-LID. In the selected datasets, we picked only two distortions, JPEG and HEVC, due to limited memory sources. We divided each dataset into two content-independent training, and test subsets. In this division, test (possibly distorted) graphs generated from one reference can only be in one of the subsets, i.e., if graphs corresponding to a specific reference LF image are in the test subset, they are not present in the training subset and vice-versa. More specifically, we define a group of scenes as a group that contains the reference LFI and its corresponding distorted versions. Then, 90% of the groups are randomly selected for training, and the remaining 10% were used for testing. We report the correlation values only for the test subset.

We used SROCC and PLCC as performance evaluation methods. We compared the proposed NR LF-IQA method with the following state-of-the-art LF-IQA methods: SDFM [107],

LF-IQM [75], MDFM [180], Tensor-NLFQ [79], NR-LFQA [77], LF-QMLI [78], BELIF [182], Xiang *et al.* [204], TSSV-LFIQA [205], PM-BLFIQM [105], Cui *at al.* [103], and ALAS-DADS [83]. We also compared the method with the following 2D-FR IQA methods: SSIM [11], PSNR [165], CORNIA [64], BRISQUE [112], DIIVINE [114], and SSEQ [48].

We train the GCNN-LFIQA method using mini-batches of size 30, 200 epochs, and Mean Square Error (MSE) as the training loss. Also, we used the Stochastic Gradient Descent (SGD) optimizer [118] with a learning rate of 0.0001 to minimize the loss function. We implemented the proposed method using Keras [190], Stellar Graphs[1], and Networkx[2] libraries of Python. The method was trained and tested on 25GB GPU, running in a LINUX environment. The code of the proposed LF-IQA method is available for download on GitHub [3], under the general public license.

7.3 Experimental Results

Table 7.3.1 shows the correlation values obtained for the Win5-LID LFI quality dataset. The rows in this table show the results for the test dataset and for each distortion, with the 'All' row corresponding to the results obtained for the complete dataset. The bold values represent the highest correlation values in a row 'All'. Notice that the proposed method performs very well for the (complete) win5-LID dataset, with SROCC value of 0.9561 and a PLCC value of 0.9664. To have a fair comparison, we ran the tests on Win5-LID LF images distorted by HEVC and JPEG compression algorithms using 2D quality metrics, and included the results in this table. We see that 2D quality metrics have obtained lower correlation values in comparison with the proposed method.

Table 7.3.2 depicts the comparison results, containing the correlation values obtained with the proposed method and with state-of-the-art LFI-IQA methods. In this table, we grouped the NR and FR LF-IQA methods into three categories, taking into consideration the models used to map the pooled features into quality estimates. The categories include methods that use (1) a pre-defined function, (2) an SVR algorithm, or (3) a CNN-based approach. For simplicity, only the overall performance ('ALL') correlation values are reported for test dataset. Notice that, the proposed method achieves the highest correlation values among all LF-IQA methods.

To demonstrate the effectiveness of the proposed approach, and the input format, we performed a simplified ablation test. In this test, we performed 2 experiments (Exp1 and Exp2) using the Win5-LID dataset. In Exp1, we removed GCNN2, GCNN4, and GCNN5 layers from GCNN block, and trained the model. In Exp2, we trained the model as it is except for the

[1] https://stellargraph.readthedocs.io/
[2] https://networkx.org/documentation/stable/index.html
[3] https://bit.ly/3M0V1ur

Table 7.3.1: The SROCC and PLCC values for Win5-LID dataset.

Dataset	Distortion	PROPOSED		SSEQ		BRISQUE		SSIM		PSNR		CORNIA		DIIVINE	
		SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
Win5-LID	HEVC	0.9556	0.9656	0.6366	0.6546	0.4940	0.6005	0.8243	0.8620	0.5743	0.5364	0.6017	0.6517	0.4608	0.5084
	JPEG	0.9375	0.9502	0.7024	0.7719	0.6476	0.774	0.8722	0.9336	0.6955	0.7895	0.7456	0.7556	0.4773	0.5355
	ALL	**0.9561**	**0.9664**	0.6771	0.7031	0.5742	0.6132	0.8339	0.8960	0.6363	0.6733	0.6543	0.6899	0.4823	0.5173

input, i.e., we used Canny edge map for the input. Table 7.3.3 shows the SROCC and PLCC values obtained for these 2 experiments and for the complete proposed model. Note that the correlation values for Exp1, and Exp2 are significantly lower than the values obtained for the complete model with skeleton inputs. In other words, the complete model provides a better performance, in terms of correlation values, than the other variants of the model.

Table 7.3.2: SROCC and PLCC values obtained for state-of-the-art LF-IQA methods tested on Win5-LID dataset.

Category	Type	Methods	Year	Win5-LID	
				SROCC	PLCC
Pre-defined Functions	FR	SSIM	2002	0.8339	0.8960
	FR	PSNR	2014	0.6363	0.6733
	NR	BRISQUE	2012	0.5742	0.6132
	NR	CORNIA	2012	0.6543	0.6899
	NR	SSEQ	2014	0.6771	0.7031
	NR	DIIVINE	2011	0.4823	0.5173
	NR	SDFM [107]	2021	0.6742	0.7142
	NR	LFIQM [75]	2021	0.2485	0.3618
	NR	MDFM [180]	2021	0.8157	0.8591
	NR	Tensor-NLFQ [79]	2019	0.9101	0.9217
SVR-based	NR	NR-LFQA [77]	2021	0.9206	0.3876
	NR	LF-QMLI [78]	2021	0.8802	0.9038
	NR	BELIF [182]	2021	0.8719	0.8910
	NR	Xiang *et al.* [204]	2021	0.9190	0.9302
	NR	TSSV-LFIQA [205]	2021	0.9194	0.9274
	NR	PM-BLFIQM [105]	2021	0.8602	0.8930
CNN-based	NR	Cui *at al.* [103]	2021	0.9116	0.9262
	NR	Guo *et al.* [82]	2021	0.9032	0.9206
	NR	ALAS-DADS [83]	2021	0.9260	0.9257
	NR	**Proposed**	2022	**0.9561**	**0.9664**

Table 7.3.4 presents the time required to train and test/run the proposed GCNN-LFIQA method, using the Win5-LID dataset. We compared the time consumption with the results of Exp1, and Exp2 (see Table 7.3.3), which corresponds to using the Canny edge maps, and GCNN block with 3 GCNN layers of 1, 3, and 5. Notice that Exp1 requires 3.1 hours for pre-processing (generating graphs from skeleton images), 3 hours for training, and 17 seconds for testing the model using test subset of Win5-LID dataset. Exp2 takes large amount time for pre-processing, because now the number of nodes and edges have increased in Canny edge maps. Also, because the size of graphs has increased, the training and testing times have also increased. The proposed method requires an equal of amount time to prepare graphs in

Table 7.3.3: Comparison of proposed model (combination) with 2 variants of the model. Training/Test is performed on the Win5-LID dataset.

Dataset	Experiment	SROCC	PLCC
Win5-LID	Exp1: GCNN Block with 3 GCNN layers + Regression Block of Figure 7.1.1	0.7952	0.8127
	Exp2: GCNN Block + Regression Block of Figure 7.1.1 but with Canny edge map input	0.4099	0.4203
	GCNN-LFIQA: Model in Figure 7.1.1 with Frequency domain EPIs	**0.9561**	**0.9664**

Table 7.3.4: The time consumption of GCNN-LFIQA method on Win5-LID dataset.

Method	Pre-Processing (hours)	Training (hours)	Testing (seconds)
Exp1	3.1	3	17
Exp2	3.9	5.3	20
GCNN-LFIQA	3.1	5.3	25

comparison with Exp1, but since the training parameters are larger than Exp1, the training and testing times have slightly increased.

7.4 Conclusions

In this chapter, we presented a novel no-reference LF-IQA method that is based on Deep Graph Convolutional Neural Network (GCNN-LFIQA). Our method not only takes into account both LF angular and spatial information, but also learns the order of pixel information from input graphs. We prepare the graphs from skeleton images, that are generate from binary format of horizontal EPIs. Specifically, the method is composed of one input layer, that takes a pair of graphs and their corresponding subjective quality scores as labels, 5 GCNN layers, and a regression block for quality prediction. We tested the proposed method on Win5-LID dataset, obtaining highest correlation values when compared to other state-of-the art methods. We also a simplified ablation test. In summary, these quantitative tests showed that the proposed NR LF-IQA method is robust and accurate. As a future work, we will perform an investigation on graphs generated from different representations of light field images, and their impact on quality estimation.

Chapter 8

Summary and Future Work

8.1 Summary

In this thesis, our goal was to investigate effective spatial, angular, and temporal features, and develop general-purpose no-reference algorithms to assess the quality of distorted 2D and 4D light field images and videos. We used machine learning, and deep learning algorithms to achieve prediction accuracy. In Chapter 3, we presented a quality assessment method for 2D images, in which, we used the multiscale local binary patterns (MLBP) and saliency information. We used the random forest regressor for training and testing using saliency-weighted textural features extracted by MLBP, showing good results that outperform state-of-the-art methods. In the same chapter, we also presented a video quality assessment method, in which we introduced a novel concept of spatial and temporal saliency along with custom objective quality scores. The method used a single CNN model, which selected the most perceptually relevant patches using spatial and temporal saliency models. The proposed method did not require subjective quality scores to train the CNN, rather, it used computed objective quality scores as target quality scores for the video frames. Although the method had a much lower cost of data processing since only a small percentage of the total video was used, its accuracy performance was not affected. In fact, the method clearly outperformed other state-of-the-art quality assessment methods. The cross-database test has shown that our method is robust and consistent across different contents and types of distortions. In the future, we intend to expand our work using other video quality datasets.

For objective quality assessment of 4D light field images, we proposed seven LFI quality assessment (LF-IQA) methods in total. In Chapter 4, we presented two LF-IQA methods HVS-CNN and DNNF-LFIQA. Both methods were based on straightforward two-stream CNN architectures. However, the DNNF-LFIQA method processed horizontal and vertical EPIs in the frequency domain for training. In Chapter 5, we presented another two LF-IQA methods LSTM-DNN and LSTM-DP that are also based on two-streams network architec-

Table 8.1.1: SROCC and PLCC values obtained for the proposed LF-IQA methods when tested on the VALID, SMART, MPI, Win5-LID, and LFDD datasets.

Category	Type	Methods	Year	MPI		VALID		SMART		Win5-LID		LFDD	
				SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC	SROCC	PLCC
DNN based	NR	Chapter 4 HVS-CNN	2022	0.9411	0.9404	0.941	0.9388	**0.9364**	0.9294	0.9469	0.9361	-	-
	NR	Chapter 4 DNNF-LFIQA	2022	-	-	**0.9783**	**0.9883**	-	-	0.9357	0.9640	0.7810	0.7332
	NR	Chapter 5 LSTM-DNN	2022	0.9484	0.9700	-	-	-	-	0.9515	0.9680	0.8083	0.7432
	NR	Chapter 5 LSTM-DP	2022	**0.9608**	**0.9766**	0.9460	0.9696	-	-	-	-	0.7708	0.8293
	NR	Chapter 6 CNN-ACL	2022	-	-	0.9733	0.9808	-	-	**0.9688**	**0.9789**	0.7928	0.8077
	NR	Chapter 6 ACL-LSTM	2022	-	-	-	-	0.9004	**0.9375**	-	-	**0.8085**	**0.8448**
	NR	Chapter 7: GCNN-LFIQA	2022	-	-	-	-	-	-	0.9561	0.9664	-	-

tures. Both methods used a Long Short-Term Memory (LSTM) architecture with a diverse set of bottleneck features extracted using three pre-trained neural networks. In Chapter 6, we presented CNN-ACL and ACL-LSTM methods for LF-IQA. The CNN-ACL was based on CNN and Atrous Convolution Layers (ACL), and had the goal of exploring dense spatial and angular features extracted from horizontal and vertical epipolar plane images (EPIs). The ACL-LSTM method used CNN, ACL, and LSTM layers to learn the long-term dependency of the distortion-related features from EPIs. Finally, in Chapter 7, we presented the GCNN-LFIQA method for LF-IQA. The method was based on graph convolutional neural network, which predicted the quality of the LFI by processing a skeleton image of the horizontal EPIs.

We have summarized the results of the proposed seven LF-IQA methods in Table 8.1.1. In this table, each column shows the correlation coefficients of the proposed LF-IQA methods when tested in a specific LF-IQA dataset, with the bold values showing the highest correlations. Notice that all methods are deep-learning reference-free and based. For the dataset LFDD, which has complex LF contents, the ACL-LSTM method has the highest correlation values, while for the SMART dataset, the HVS-CNN method obtained the highest SROCC value and the ACL-LSTM method obtained the highest PLCC value. The method LSTM-DP obtained the highest correlations for the MPI dataset, while for the VALID dataset the method DNNF-LFIQA obtained the highest correlations. Finally, for the Win5-LID dataset, the CNN-ACL method obtained the highest correlations, while the method GCNN-LFIQA is the second best performing method with respect to the SROCC value and the LSTM-DNN method is the second best with respect to the PLCC.

8.2 Future Work

As part of the future work in LF image quality assessment, it would be interesting to investigate the complexity of the LFDD dataset. As shown in Table 8.1.1, the highest correlations obtained for this dataset was 0.8085 (SROCC) and 0.8448 (PLCC), which are lower numbers than what was obtained for the other datasets. Furthermore, we believe that the GCNN-LFIQA method can be improved by using different LFI formats. Currently, the method uses skeleton images of RGB horizontal EPIs, which limit the size of the graphs, keeping only the important nodes. More investigation will be performed to analyse distinct strategies to incorporate only important nodes.

A second part of the future work is to perform a comprehensive analysis of incorporating saliency into LF-IQA methods. Currently, there is one method available that incorporates 2D saliency models information into LF-IQA method. The 2D saliency models do not incorporate angular features of an LFI while computing saliency information. Therefore, we intend to conduct a study of the impact of saliency models on the performance of the LFI quality assessment. Additionally, since a LFI can be visualized using different rendering techniques, we intend to study which format will show the best saliency information. Our aim is to answer the following research question: "Where to look for saliency information to improve quality prediction?".

Finally, more subjective experiments are needed to establish larger quality LF datasets. These experiments should also provide the normalized subjective quality scores following a similar format across different datasets. More generic DNN-based objective quality methods are needed to estimate LF quality for different scenarios, different resolutions and formats, and different types of degradations. Furthermore, there is a lack of quality assessment methods for LF videos, given the absence of video LF datasets.

References

[1] *Lytro illum*. https://lytro.com, 2008. Accessed: 2019-10-18. ix, 14, 15

[2] Raytrix: *Raytrix r29 3d plenoptic light-field camera*, November 2021. https://raytrix.de/. ix, 14, 15

[3] Shi, L., S. Zhao, W. Zhou, and Z. Chen: *Perceptual evaluation of light field image*. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 41–45, Oct 2018. ix, xi, 16, 31, 32, 70, 86, 101, 102

[4] Zizien, Adam and Karel Fliegel: *LFDD: Light field image dataset for performance evaluation of objective quality metrics*. In Tescher, Andrew G. and Touradj Ebrahimi (editors): *Applications of Digital Image Processing XLIII*, volume 11510, pages 671 – 683. International Society for Optics and Photonics, SPIE, 2020. https://doi.org/10.1117/12.2568490. ix, 7, 17, 31, 32, 70, 86, 94

[5] Li, Jie and Yue Zhou: *Visual saliency based blind image quality assessment via convolutional neural network*. pages 550–557, 2017. x, 28, 39, 40, 41, 42

[6] Zhou, W., Z. Chen, and W. Li: *Dual-stream interactive networks for no-reference stereoscopic image quality assessment*. IEEE Transactions on Image Processing, 28(8):3946–3958, 2019. x, 49, 50

[7] Adhikarla, Vamsi Kiran, Marek Vinkler, Denis Sumin, Rafał Mantiuk, Karol Myszkowski, Hans Peter Seidel, and Piotr Didyk: *Towards a quality metric for dense light fields*. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. x, 31, 32, 51, 52, 53, 70

[8] Cisco: *Cisco annual internet report (2018–2023)*, February 2020. 1

[9] Bt.500-13, ITU R:: *methodology for the subjective assessment of the quality of television pictures*, oct 2019. https://www.itu.int/rec/R-REC-BT.500. 2

[10] REC, ITUT: P. 800.1: *mean opinion score (mos) terminology*,, jul 2019. https://www.itu.int/rec/R-REC-BT.500. 2

[11] Wang, Zhou, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli: *Image quality assessment: from error visibility to structural similarity*. IEEE transactions on image processing, 13(4):600–612, 2004. 2, 4, 32, 35, 53, 105

[12] Watson, A. B., G. Y. Yang, J. A. Solomon, and J. Villasenor: *Visibility of wavelet quantization noise*. IEEE Transactions on Image Processing, 6(8):1164–1175, Aug 1997, ISSN 1941-0042. 2

[13] Daly, Scott J.: *Visible differences predictor: an algorithm for the assessment of image fidelity*. In Rogowitz, Bernice E. (editor): *Human Vision, Visual Processing, and Digital Display III*, volume 1666, pages 2 – 15. International Society for Optics and Photonics, SPIE, 1992. https://doi.org/10.1117/12.135952. 2

[14] Lubin, Jeffrey: *Digital images and human vision*. chapter The Use of Psychophysical Data and Models in the Analysis of Display System Performance, pages 163–178. MIT Press, Cambridge, MA, USA, 1993, ISBN 0-262-23171-9. http://dl.acm.org/citation.cfm?id=197765.197782. 2

[15] Watson, Andrew: *Visual optimization of dct quantization matrices for individual images*. Proc. AIAA Computing in Aerospace, 9, February 1993. 2

[16] Mannos, J. and D. Sakrison: *The effects of a visual fidelity criterion of the encoding of images*. IEEE Transactions on Information Theory, 20(4):525–536, July 1974, ISSN 1557-9654. 2

[17] You, J., J. Korhonen, and A. Perkis: *Attention modeling for video quality assessment: Balancing global quality and local quality*. In *2010 IEEE International Conference on Multimedia and Expo*, pages 914–919, July 2010. 2, 28

[18] Xin Feng, Tao Liu, D. Yang, and Yao Wang: *Saliency based objective quality assessment of decoded video affected by packet losses*. In *2008 15th IEEE International Conference on Image Processing*, pages 2560–2563, Oct 2008. 2, 28

[19] Chandler, D. M. and S. S. Hemami: *Vsnr: A wavelet-based visual signal-to-noise ratio for natural images*. IEEE Transactions on Image Processing, 16(9):2284–2298, Sep. 2007, ISSN 1941-0042. 2

[20] Pinson, M. H. and S. Wolf: *A new standardized method for objectively measuring video quality*. IEEE Transactions on Broadcasting, 50(3):312–322, Sep. 2004, ISSN 0018-9316. 2

[21] Gu, Ke, Guangtao Zhai, Weisi Lin, and Min Liu: *The analysis of image contrast: From quality assessment to automatic enhancement*. IEEE transactions on cybernetics, 46(1):284–297, 2016. 2, 35

[22] Zhang, L., L. Zhang, X. Mou, and D. Zhang: *Fsim: A feature similarity index for image quality assessment*. IEEE Transactions on Image Processing, 20(8):2378–2386, Aug 2011, ISSN 1941-0042. 2

[23] Wang, Z. and Q. Li: *Information content weighting for perceptual image quality assessment*. IEEE Transactions on Image Processing, 20(5):1185–1198, May 2011, ISSN 1941-0042. 2

[24] Wang, Z., E. P. Simoncelli, and A. C. Bovik: *Multiscale structural similarity for image quality assessment.* In *The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003*, volume 2, pages 1398–1402 Vol.2, Nov 2003. 2

[25] Sheikh, H. R. and A. C. Bovik: *Image information and visual quality.* IEEE Transactions on Image Processing, 15(2):430–444, Feb 2006, ISSN 1941-0042. 2, 53

[26] Pessoa, A., A. Falcão, R. Nishihara, A. Silva, and R. Lotufo: *Video quality assessment using objective parameters based on image segmentation.* SMPTE Journal, 108(12):865–872, Dec 1999, ISSN 0036-1682. 3

[27] Wolf, Stephen and Margaret H. Pinson: *Spatial-temporal distortion metric for in-service quality monitoring of any digital video system.* In Tescher, Andrew G., Bhaskaran Vasudev, V. Michael Bove Jr., and Barbara Derryberry (editors): *Multimedia Systems and Applications II*, volume 3845, pages 266 – 277. International Society for Optics and Photonics, SPIE, 1999. https://doi.org/10.1117/12.371210. 3

[28] Winkler, Stefan: *Issues in vision modeling for perceptual video quality assessment.* Signal Processing, 78(2):231 – 252, 1999, ISSN 0165-1684. http://www.sciencedirect.com/science/article/pii/S0165168499000626. 3

[29] Algazi, V. R. and N. Hiwasa: *Perceptual criteria and design alternatives for low bit rate video coding.* In *Proceedings of 27th Asilomar Conference on Signals, Systems and Computers*, pages 831–835 vol.2, Nov 1993. 3

[30] Webster, Arthur A., Coleen T. Jones, Margaret H. Pinson, Stephen D. Voran, and Stephen Wolf: *Objective video quality assessment system based on human perception.* In Allebach, Jan P. and Bernice E. Rogowitz (editors): *Human Vision, Visual Processing, and Digital Display IV*, volume 1913, pages 15 – 26. International Society for Optics and Photonics, SPIE, 1993. https://doi.org/10.1117/12.152700. 3

[31] Battisti, F., M. Carli, Y. Liu, A. Neri, and P. Paudyal: *Distortion-based no-reference quality metric for video transmission over ip.* In *2015 International Symposium on Signals, Circuits and Systems (ISSCS)*, pages 1–4, July 2015. 3

[32] Zhu, Kongfeng, Chengqing Li, Vijayan Asari, and Dietmar Saupe: *No-reference video quality assessment based on artifact measurement and statistical analysis.* Circuits and Systems for Video Technology, IEEE Transactions on, 25:533–546, April 2015. 3

[33] Jia, Lixiu, Xuefei Zhong, Yan Tu, and Wenjuan Niu: *A no-reference video quality assessment metric based on ROI.* In Larabi, Mohamed Chaker and Sophie Triantaphillidou (editors): *Image Quality and System Performance XII*, volume 9396, pages 314 – 327. International Society for Optics and Photonics, SPIE, 2015. https://doi.org/10.1117/12.2083892. 3, 4

[34] Izumi, K., K. Kawamura, T. Yoshino, and S. Naito: *No reference video quality assessment based on parametric analysis of hevc bitstream.* In *2014 Sixth International Workshop on Quality of Multimedia Experience (QoMEX)*, pages 49–50, Sep. 2014. 3

[35] Eskicioglu, A. M. and P. S. Fisher: *Image quality measures and their performance*. IEEE Transactions on Communications, 43(12):2959–2965, Dec 1995, ISSN 1558-0857. 3

[36] 60, ITU T SG09 (Study Period 2001) Contribution: *Final report from the video quality experts group on the validation of objective models of video quality assessment, phase ii (fr-tv2)*, sep 2018. https://www.itu.int/md/T01-SG09-C-0060. 3

[37] Zhang, W., A. Borji, Z. Wang, P. Le Callet, and H. Liu: *The application of visual saliency models in objective image quality assessment: A statistical evaluation*. IEEE Transactions on Neural Networks and Learning Systems, 27(6):1266–1278, June 2016, ISSN 2162-2388. 3, 28

[38] Zhang, W. and H. Liu: *Study of saliency in objective video quality assessment*. IEEE Transactions on Image Processing, 26(3):1275–1288, March 2017, ISSN 1941-0042. 3, 28

[39] Feng, X., T. Liu, D. Yang, and Y. Wang: *Saliency inspired full-reference quality metrics for packet-loss-impaired video*. IEEE Transactions on Broadcasting, 57(1):81–88, March 2011, ISSN 1557-9611. 3, 28

[40] Ćulibrk, D., M. Mirković, V. Zlokolica, M. Pokrić, V. Crnojević, and D. Kukolj: *Salient motion features for video quality assessment*. IEEE Transactions on Image Processing, 20(4):948–958, April 2011, ISSN 1941-0042. 3, 28

[41] Hemami, Sheila S. and Amy R. Reibman: *No-reference image and video quality estimation: Applications and human-motivated design*. Signal Processing: Image Communication, 25(7):469 – 481, 2010, ISSN 0923-5965. http://www.sciencedirect.com/science/article/pii/S0923596510000688, Special Issue on Image and Video Quality Assessment. 4

[42] Guan-Hao Chen, Chun-Ling Yang, Lai-Man Po, and Sheng-Li Xie: *Edge-based structural similarity for image quality assessment*. In *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, volume 2, pages II–II, May 2006. 4

[43] Li, Qiaohong, Weisi Lin, and Yuming Fang: *Bsd: Blind image quality assessment based on structural degradation*. Neurocomputing, 236:93 – 103, 2017, ISSN 0925-2312. http://www.sciencedirect.com/science/article/pii/S092523121631390X, Good Practices in Multimedia Modeling. 4

[44] Wu, Qingbo, Hongliang Li, and King N. Ngan: *Gip: Generic image prior for no reference image quality assessment*. In Chen, Enqing, Yihong Gong, and Yun Tie (editors): *Advances in Multimedia Information Processing - PCM 2016*, pages 600–608, Cham, 2016. Springer International Publishing, ISBN 978-3-319-48896-7. 4

[45] Liu, Lixiong, Hongping Dong, Hua Huang, and Alan C Bovik: *No-reference image quality assessment in curvelet domain*. Signal Processing: Image Communication, 29(4):494–505, 2014. 4, 19, 35

[46] Saad, M. A., A. C. Bovik, and C. Charrier: *Blind prediction of natural video quality*. IEEE Transactions on Image Processing, 23(3):1352–1365, March 2014, ISSN 1057-7149. 4, 19, 43

[47] Li, Q., W. Lin, and Y. Fang: *No-reference quality assessment for multiply-distorted images in gradient domain*. IEEE Signal Processing Letters, 23(4):541–545, April 2016, ISSN 1070-9908. 4, 19, 43

[48] L. Liu, B. Liu, H. Huang and A.C. Bovik: *No-reference image quality assessment based on spatial and spectral entropies*. Signal Processing: Image Communication, June 2014. 4, 19, 35, 42, 105

[49] Hadizadeh, Hadi and Ivan V. Bajić: *No-reference image quality assessment using statistical wavelet-packet features*. Pattern Recognition Letters, 80:144 – 149, 2016, ISSN 0167-8655. http://www.sciencedirect.com/science/article/pii/S0167865516301349. 4

[50] Men, Hui, Hanhe Lin, and Dietmar Saupe: *Spatiotemporal feature combination model for no-reference video quality assessment*. IEEE, April 2018. 4, 19

[51] Ahn, S. and S. Lee: *Deep blind video quality assessment based on temporal human perception*. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 619–623, Oct 2018. 4, 26

[52] Moorthy, A. K. and A. C. Bovik: *Visual importance pooling for image quality assessment*. IEEE Journal of Selected Topics in Signal Processing, 3(2):193–201, April 2009, ISSN 1941-0484. 4

[53] Xue, W., L. Zhang, X. Mou, and A. C. Bovik: *Gradient magnitude similarity deviation: A highly efficient perceptual image quality index*. IEEE Transactions on Image Processing, 23(2):684–695, Feb 2014, ISSN 1941-0042. 4, 53

[54] Li, Y., L. Po, C. Cheung, X. Xu, L. Feng, F. Yuan, and K. Cheung: *No-reference video quality assessment with 3d shearlet transform and convolutional neural networks*. IEEE Transactions on Circuits and Systems for Video Technology, 26(6):1044–1057, June 2016, ISSN 1051-8215. 5, 25, 43

[55] Freitas, Pedro Garcia, Welington Y.L. Akamine, and Mylène C.Q. Farias: *Using multiple spatio-temporal features to estimate video quality*. Signal Processing: Image Communication, 64:1 – 10, 2018, ISSN 0923-5965. 5, 19, 43

[56] Singh, Ranjit and Naveen Aggarwal: *A distortion-agnostic video quality metric based on multi-scale spatio-temporal structural information*. Signal Processing: Image Communication, 74:299 – 308, 2019, ISSN 0923-5965. 5, 26, 43

[57] Moorthy, A. K. and A. C. Bovik: *Efficient video quality assessment along temporal trajectories*. IEEE Transactions on Circuits and Systems for Video Technology, 20(11):1653–1658, Nov 2010, ISSN 1558-2205. 5

[58] Seshadrinathan, Kalpana and Alan Bovik: *A structural similarity metric for video based on motion models*. Volume 1, pages I–869, May 2007. 5

[59] Ninassi, Alexandre, Olivier Le Meur, Patrick Le Callet, and Dominique Barba: *Considering temporal variations of spatial visual distortions in video quality assessment*. Selected Topics in Signal Processing, IEEE Journal of, 3:253 – 265, May 2009. 5

[60] Gujjunoori, Sagar and Madhu Oruganti: *Hvs based full reference video quality assessment based on optical flow*. pages 70–75, 2018. 5, 28

[61] Aabed, M. A. and G. AlRegib: *Peqaso: Perceptual quality assessment of streamed videos using optical flow features*. IEEE Transactions on Broadcasting, pages 1–12, 2018, ISSN 0018-9316. 5, 28

[62] Li, X., Q. Guo, and X. Lu: *Spatiotemporal statistics for video quality assessment*. IEEE Transactions on Image Processing, 25(7):3329–3342, July 2016, ISSN 1057-7149. 5, 19, 43

[63] Pinson, M. H. and S. Wolf: *A new standardized method for objectively measuring video quality*. IEEE Transactions on Broadcasting, 50(3):312–322, Sep. 2004, ISSN 0018-9316. 5, 43

[64] P. Ye, J. Kumar, L. Kang and D. Doermann: *Unsupervised feature learning framework for no-reference image quality assessment*. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2012. 5, 19, 35, 42, 105

[65] Zhang, M., C. Muramatsu, X. Zhou, T. Hara, and H. Fujita: *Blind image quality assessment using the joint statistics of generalized local binary pattern*. IEEE Signal Processing Letters, 22(2):207–210, Feb 2015, ISSN 1558-2361. 5

[66] Zhang, M., J. Xie, X. Zhou, and H. Fujita: *No reference image quality assessment based on local binary pattern statistics*. In *2013 Visual Communications and Image Processing (VCIP)*, pages 1–6, Nov 2013. 5

[67] Ye, P. and D. Doermann: *No-reference image quality assessment using visual codebooks*. IEEE Transactions on Image Processing, 21(7):3129–3138, July 2012, ISSN 1941-0042. 5

[68] Sandić-Stanković, Dragana, Dragan Kukolj, and Patrick Le Callet: *Dibr-synthesized image quality assessment based on morphological multi-scale approach*. EURASIP Journal on Image and Video Processing, 2017(1):4, Jul 2016, ISSN 1687-5281. `https://doi.org/10.1186/s13640-016-0124-7`. 5, 53

[69] Rouse, D. M. and S. S. Hemami: *Natural image utility assessment using image contours*. In *2009 16th IEEE International Conference on Image Processing (ICIP)*, pages 2217–2220, Nov 2009. 5

[70] Bolles, Robert C., H. Harlyn Baker, and David H. Marimont: *Epipolarplane image analysis: An approach to determining structure from motion*. In *INTERN..1. COMPUTER VISION*, pages 1–7, 1987. 6

[71] Amirpour, H., A. M. G. Pinheiro, M. Pereira, and M. Ghanbari: *Reliability of the most common objective metrics for light field quality assessment*. In *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2402–2406, 2019. 6

[72] ak, ali and Patrick Le Callet: *Investigating Epipolar Plane Image Representations for Objective Quality Evaluation of Light Field Images*. In *European Workshop on Visual Information Processing*, Rome, Italy, October 2019. 6, 17, 61, 70, 77, 86, 95

[73] Monteiro, Ricardo, Paulo Nunes, Nuno Rodrigues, and Sergio De Faria: *Light field image coding: objective performance assessment of lenslet and 4d lf data representations*. page 13, September 2018. 6, 17

[74] Mahmoudpour, Saeed and Peter Schelkens: *Cross data set performance consistency of objective quality assessment methods for light fields*. 2020 Twelfth International Conference on Quality of Multimedia Experience (QoMEX), 2020. 6, 55, 61, 70, 77, 86, 89, 95

[75] Paudyal, P., F. Battisti, and M. Carli: *Reduced reference quality assessment of light field images*. IEEE Transactions on Broadcasting, 65(1):152–165, March 2019. 6, 53, 61, 64, 70, 73, 77, 80, 86, 88, 95, 98, 105, 106

[76] Meng, C., P. An, X. Huang, C. Yang, and D. Liu: *Full reference light field image quality evaluation based on angular-spatial characteristic*. IEEE Signal Processing Letters, 27:525–529, 2020. 6, 53, 56

[77] Shi, Likun, Wei Ran Zhou, and Zhibo Chen: *No-reference light field image quality assessment based on spatial-angular measurement*. ArXiv, abs/1908.06280, 2019. 6, 17, 53, 70, 73, 77, 80, 105, 106

[78] Luo, Ziyuan, Wei Ran Zhou, Likun Shi, and Zhibo Chen: *No-reference light field image quality assessment based on micro-lens image*. ArXiv, abs/1908.10087, 2019. 6, 17, 53, 70, 73, 105, 106

[79] Zhou, Wei, Shi Likun, and Zhibo Chen: *Tensor oriented no-reference light field image quality assessment*, September 2019. 6, 53, 61, 64, 70, 73, 77, 80, 86, 88, 95, 98, 105, 106

[80] Ak, Ali, Suiyi Ling, and Patrick Le Callet: *NO-REFERENCE QUALITY EVALUATION OF LIGHT FIELD CONTENT BASED ON STRUCTURAL REPRESENTATION OF THE EPIPOLAR PLANE IMAGE*. In *The 1st ICME Workshop on Hyper-Realistic Multimedia for Enhanced Quality of Experience*, London, United Kingdom, July 2020. 6, 17, 53, 70, 73

[81] Fujita, S., K. Takahashi, and T. Fujii: *How should we handle 4d light fields with cnns?* In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 2600–2604, Oct 2018. 6

[82] Guo, Zixuan, Wei Gao, Haiqiang Wang, Junle Wang, and Songlin Fan: *No-reference deep quality assessment of compressed light field images*. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2021. 6, 69, 70, 72, 73, 76, 106

[83] Qu, Qiang, Xiaoming Chen, Vera Chung, and Zhibo Chen: *Light field image quality assessment with auxiliary learning based on depthwise and anglewise separable convolutions*. IEEE Transactions on Broadcasting, pages 1–14, 2021. 6, 26, 53, 61, 64, 70, 73, 86, 88, 95, 98, 105, 106

[84] Lamichhane, Kamal, Federica Battisti, Pradip Paudyal, and Marco Carli: *Exploiting saliency in quality assessment for light field images.* In *2021 Picture Coding Symposium (PCS)*, pages 1–5, 2021. 6, 26, 29, 53

[85] Zhao, Ping, Xiaoming Chen, Vera Chung, and Haisheng Li: *Delfiqe—a low-complexity deep learning-based light field image quality evaluator.* IEEE Transactions on Instrumentation and Measurement, 70:1–11, 2021. 6, 7, 26, 61, 70, 72, 73, 81, 95, 98

[86] Zhao, Ping: *Low-Complexity Deep Learning-Based Light Field Image Quality Assessment.* PhD thesis, 2021. `https://hdl.handle.net/2123/25977`, Includes publications. 6, 7, 26, 61, 64, 77, 79, 80, 86, 88, 95, 98

[87] Viola, Irene, Martin Řeřábek, and Touradj Ebrahimi: *Impact of interactivity on the assessment of quality of experience for light field content.* In *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6, 2017. 7, 65

[88] Grill-Spector, Kalanit and Rafael Malach: *The human visual cortex.* Annual Review of Neuroscience, 27(10.1146/annurev.neuro.27.070203.144220):649–677, 2004. 7

[89] Goodale, Melvyn A. and A.David Milner: *Separate visual pathways for perception and action.* Trends in Neurosciences, 15(1):20–25, 1992, ISSN 0166-2236. `https://www.sciencedirect.com/science/article/pii/0166223692903448`. 7

[90] James Algina, H. J. Keselman: *Comparing squared multiple correlation coefficients: Examination of a confidence interval and a test significance.* Psychological Methods, (1939-1463):76–83, 1999. 12, 13

[91] Bobko, P.: *Correlation and regression: Applications for industrial organizational psychology and management (2nd ed.).* CA: Sage Publications, 2001. 12, 13

[92] Gershun, A.: *The light field.* Journal of Mathematics and Physics, 18(1-4):51–151, 1939. 13

[93] Adelson, Edward H. and James R. Bergen: *The plenoptic function and the elements of early vision.* In *Computational Models of Visual Processing*, pages 3–20. MIT Press, Cambridge, MA, 1991. 13

[94] Levoy, Marc and Pat Hanrahan: *Light field rendering.* In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, page 31–42, New York, NY, USA, 1996. Association for Computing Machinery, ISBN 0897917464. 13

[95] Gortler, Steven J., Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen: *The lumigraph.* In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, page 43–54, New York, NY, USA, 1996. Association for Computing Machinery, ISBN 0897917464. 13

[96] Wu, G., B. Masia, A. Jarabo, Y. Zhang, L. Wang, Q. Dai, T. Chai, and Y. Liu: *Light field image processing: An overview.* IEEE Journal of Selected Topics in Signal Processing, 11(7):926–954, Oct 2017. 14

[97] Li, Nianyi, Jinwei Ye, Yu Ji, Haibin Ling, and Jingyi Yu: *Saliency detection on light field.* In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2806–2813, 2014. 17

[98] Tao, Michael, Sunil Hadap, Jitendra Malik, and Ravi Ramamoorthi: *Depth from combining defocus and correspondence using light-field cameras.* pages 673–680, December 2013. 17

[99] Viola, Irene and Touradj Ebrahimi: *An in-depth analysis of single-image subjective quality assessment of light field contents.* pages 1–6, June 2019. 17

[100] Bakir, N., S. A. Fezza, W. Hamidouche, K. Samrouth, and O. Déforges: *Subjective evaluation of light field image compression methods based on view synthesis.* In *2019 27th European Signal Processing Conference (EUSIPCO)*, pages 1–5, 2019. 17

[101] Medda, Daniele, Wei Song, and Cristian Perra: *Objective image quality analysis of convolutional neural network light field coding.* pages 163–168, October 2019. 17

[102] Fang, Y., K. Wei, J. Hou, W. Wen, and N. Imamoglu: *Light filed image quality assessment by local and global features of epipolar plane image.* In *2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM)*, pages 1–6, Sep. 2018. 17, 53

[103] Cui, Yueli, Mei Yu, Zhidi Jiang, Zongju Peng, and Fen Chen: *Blind light field image quality assessment by analyzing angular-spatial characteristics.* Digital Signal Processing, 117:103138, 2021, ISSN 1051-2004. `https://www.sciencedirect.com/science/article/pii/S1051200421001779`. 17, 105, 106

[104] Jiang, Gangyi, Zhijiao Huang, Mei Yu, Haiyong Xu, Yang Song, and Hao Jiang: *New quality assessment approach for dense light fields.* page 44, November 2018. 17, 53

[105] Liu, Yun, Gangyi Jiang, Zhidi Jiang, Zhiyong Pan, Mei Yu, and Yo Sung Ho: *Pseudo-reference sub-aperture images and micro-lens image based blind light field image quality measurement.* IEEE Transactions on Instrumentation and Measurement, pages 1–1, 2021. 17, 105, 106

[106] E. Palma, F. Battisti, M. Carli P. Astola and I. Tabus: *Subjective Quality Evaluation of Light Field Data Under Coding Distortions.* EUSIPCO 2020, 978-9-0827-9705-3 edition, apr 2020. 17

[107] Tian, Y., H. Zeng, J. Hou, J. Chen, J. Zhu, and K. Ma: *A light field image quality assessment model based on symmetry and depth features.* IEEE Transactions on Circuits and Systems for Video Technology, pages 1–1, 2020. 17, 53, 61, 64, 70, 73, 77, 80, 86, 88, 95, 98, 104, 106

[108] Fang, Y., K. Wei, J. Hou, W. Wen, and N. Imamoglu: *Light filed image quality assessment by local and global features of epipolar plane image.* In *2018 IEEE Fourth International Conference on Multimedia Big Data (BigMM)*, pages 1–6, Sep. 2018. 17

[109] Shan, L., P. An, C. Meng, X. Huang, C. Yang, and L. Shen: *A no-reference image quality assessment metric by multiple characteristics of light field images.* IEEE Access, 7:127217–127229, 2019. 17, 53

[110] Aravind, Pai: *Aravind pai' answer to ẅhy is deep learning called as such?"*, dec 2018. https://www.quora.com/Why-is-deep-learning-called-as-such. 18, 21

[111] Alpaydin, Ethem: *Introduction to Machine Learning*. The MIT Press, 2014, ISBN 0262028182, 9780262028189. 18, 19

[112] A. Mittal, A. K. Moorthy and A. C. Bovik: *No-reference image quality assessment in the spatial domain*. IEEE Transactions on Image Processing, 2012. 19, 35, 42, 105

[113] Freitas, Pedro Garcia, Welington YL Akamine, and Mylene CQ Farias: *No-reference image quality assessment based on statistics of local ternary pattern*. In *Quality of Multimedia Experience (QoMEX), 2016 Eighth International Conference on*, pages 1–6. IEEE, 2016. 19, 35

[114] Moorthy, A. K. and A. C. Bovik: *Blind image quality assessment: From scene statistics to perceptual quality*. EEE Transactions Image Processing, pages 3350–3364, December 2011. 19, 42, 105

[115] Freitas, Pedro Garcia, Welington YL Akamine, and Mylène CQ Farias: *Blind image quality assessment using multiscale local binary patterns*. Journal of Imaging Science and Technology, 60(6):60405–1, 2016. 19, 33

[116] Saishruthi, Swaminathan: *Linear regression — detailed view*, feb 2018. https://towardsdatascience.com/linear-regression-detailed-view-ea73175f6e86. 19

[117] Kim, Daeho, Jinah Kim, and Jaeil Kim: *Elastic exponential linear units for convolutional neural networks*. Neurocomputing, 406:253–266, 2020, ISSN 0925-2312. https://www.sciencedirect.com/science/article/pii/S0925231220304240. 22, 59, 68, 74

[118] Sutskever, Ilya, James Martens, George Dahl, and Geoffrey Hinton: *On the importance of initialization and momentum in deep learning*. In *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*, ICML'13, page III–1139–III–1147. JMLR.org, 2013. 22, 53, 62, 78, 86, 95, 105

[119] Zhang, Jun, Yamei Liu, Shengping Zhang, Ronald Poppe, and Meng Wang: *Light field saliency detection with deep convolutional networks*. IEEE Transactions on Image Processing, 29:4421–4434, 2020. 23, 29, 85, 93

[120] Swasono, D. I., H. Tjandrasa, and C. Fathicah: *Classification of tobacco leaf pests using vgg16 transfer learning*. In *2019 12th International Conference on Information Communication Technology and System (ICTS)*, pages 176–181, July 2019. 23

[121] Vasudev, Rakshith: *Understanding and calculating the number of parameters in convolution neural networks (cnns)*, feb 2011. https://towardsdatascience.com/understanding-and-calculating-the-number-of-parameters-in-convolution-neural-networks-cnns-fc88790d530d. 23

[122] Saha, Sumit: *A comprehensive guide to convolutional neural networks — the eli5 way*, dec 2018. https://towardsdatascience.com/a-comprehensive-guide-to-convolutional-neural-networks-the-eli5-way-3bd2b1164a53. 23, 24

[123] Wang, Yequan, Minlie Huang, Xiaoyan Zhu, and Li Zhao: *Attention-based LSTM for aspect-level sentiment classification*. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615, Austin, Texas, November 2016. Association for Computational Linguistics. https://www.aclweb.org/anthology/D16-1058. 24

[124] Sun, Lin, Kui Jia, Kevin Chen, Dit Yan Yeung, Bertram E. Shi, and Silvio Savarese: *Lattice long short-term memory for human action recognition*. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2166–2175, 2017. 24

[125] blog colah: *Understanding lstm networks*, aug 2015. http://colah.github.io/posts/2015-08-Understanding-LSTMs/. 25

[126] Kang, L., P. Ye, Y. Li, and D. Doermann: *Convolutional neural networks for no-reference image quality assessment*. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1733–1740, June 2014. 25

[127] Domonkos, Varga: *No-reference video quality assessment based on the temporal pooling of deep features*. Apr 2019. 25

[128] Kim, J. and S. Lee: *Fully deep blind image quality predictor*. IEEE Journal of Selected Topics in Signal Processing, 11(1):206–220, Feb 2017. 26

[129] Domonkos, Varga and Tamás Szirányi: *No-reference video quality assessment via pretrained cnn and lstm networks*. Signal, Image and Video Processing, Jun 2019. 26

[130] Zhang, Muhan, Zhicheng Cui, Marion Neumann, and Yixin Chen: *An end-to-end deep learning architecture for graph classification*. AAAI'18/IAAI'18/EAAI'18. AAAI Press, 2018, ISBN 978-1-57735-800-8. 26, 102

[131] Kipf, Thomas N. and Max Welling: *Semi-supervised classification with graph convolutional networks*. CoRR, abs/1609.02907, 2016. http://arxiv.org/abs/1609.02907. 26

[132] Zhang, L. and W. Lin: *Introduction to Visual Attention*, pages 1–24. IEEE, 2013, ISBN 9781118060056978047082813 7. 27

[133] Itti, L., C. Koch, and E. Niebur: *A model of saliency-based visual attention for rapid scene analysis*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(11):1254–1259, Nov 1998, ISSN 1939-3539. 27, 29

[134] Harel, Jonathan, Christof Koch, and Pietro Perona: *Graph-based visual saliency*. In *Proceedings of the 19th International Conference on Neural Information Processing Systems*, NIPS'06, pages 545–552, Cambridge, MA, USA, 2006. MIT Press. http://dl.acm.org/citation.cfm?id=2976456.2976525. 27, 29

[135] Zhang, L. and W. Lin: *Application of Attention Models in Image Processing*, pages 271–303. IEEE, 2013, ISBN 9781118060056978047082813 7. 28

[136] Sadaka, N. G., L. J. Karam, R. Ferzli, and G. P. Abousleman: *A no-reference perceptual image sharpness metric based on saliency-weighted foveal pooling*. In *2008 15th IEEE International Conference on Image Processing*, pages 369–372, Oct 2008. 28

[137] Rao, D. V., N. Sudhakar, I. R. Babu, and L. P. Reddy: *Image quality assessment complemented with visual regions of interest*. In *2007 International Conference on Computing: Theory and Applications (ICCTA'07)*, pages 681–687, March 2007. 28

[138] Zhang, Lin, Ying Shen, and Hongyu Li: *Vsi: A visual saliency-induced index for perceptual image quality assessment*. IEEE Transactions on Image Processing, 23(10):4270–4281, 2014. 28

[139] Farias, Mylène CQ and Welington YL Akamine: *On performance of image quality metrics enhanced with visual attention computational models*. Electronics letters, 48(11):631–633, 2012. 28

[140] Engelke, Ulrich, Hagen Kaprykowsky, Hans Jürgen Zepernick, and Patrick Ndjiki-Nya: *Visual attention in quality assessment*. IEEE Signal Processing Magazine, 28(6):50–59, 2011. 28

[141] Gu, Ke, Shiqi Wang, Huan Yang, Weisi Lin, Guangtao Zhai, Xiaokang Yang, and Wenjun Zhang: *Saliency-guided quality assessment of screen content images*. IEEE Transactions on Multimedia, 18(6):1098–1110, 2016. 28

[142] Chao Li, Bin Zhan, Shuo Zhang, and Hao Sheng: *Saliency detection with relative location measure in light field image*. In *2017 2nd International Conference on Image, Vision and Computing (ICIVC)*, pages 8–12, June 2017. 29

[143] Wang, Tiantian, Yongri Piao, Huchuan Lu, Xiao Li, and Lihe Zhang: *Deep learning for light field saliency detection*. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8837–8847, 2019. 29

[144] Gill, Ailbhe, Emin Zerman, Martin Alain, Mikael Le Pendu, and Aljosa Smolic: *Focus guided light field saliency estimation*. In *2021 13th International Conference on Quality of Multimedia Experience (QoMEX)*, pages 213–218, 2021. 29

[145] Mazumdar, Pramit, Kamal Lamichhane, Marco Carli, and Federica Battisti: *A feature integrated saliency estimation model for omnidirectional immersive images*. Electronics, 8(12), 2019, ISSN 2079-9292. https://www.mdpi.com/2079-9292/8/12/1538. 29

[146] Zhang, Jianming and Stan Sclaroff: *Saliency detection: A boolean map approach*. In *2013 IEEE International Conference on Computer Vision*, pages 153–160, 2013. 29

[147] Zhang, Jianming and Stan Sclaroff: *Exploiting surroundedness for saliency detection: A boolean map approach*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 38(5):889–902, 2016. 29

[148] Vu, P. V. and D. M. Chandler: *Vis3: An algorithm for video quality assessment via analysis of spatial and spatiotemporal slices.* Journal of Electronic Imaging, 2014. 29, 30, 40, 42, 43

[149] K. Seshadrinathan, R. Soundararajan, A. C. Bovik and L. K. Cormack: *Study of subjective and objective quality assessment of video.* EEE Transactions on Image Processing, 19(6):1427–1441, June 2010. 29, 30, 43

[150] Ponomarenko, Nikolay, Lina Jin, Oleg Ieremeiev, Vladimir Lukin, Karen Egiazarian, Jaakko Astola, Benoit Vozel, Kacem Chehdi, Marco Carli, Federica Battisti, *et al.*: *Image database tid2013: Peculiarities, results and perspectives.* Signal Processing: Image Communication, 30:57–77, 2015. 29, 30, 35

[151] Larson, Eric C and Damon M Chandler: *Most apparent distortion: full-reference image quality assessment and the role of strategy.* Journal of Electronic Imaging, 19(1):011006–011006, 2010. 29, 30, 35

[152] Sheikh, Hamid R, Muhammad F Sabir, and Alan C Bovik: *A statistical evaluation of recent full reference image quality assessment algorithms.* IEEE Transactions on image processing, 15(11):3440–3451, 2006. 29, 30, 35

[153] Viola, Irene and Touradj Ebrahimi: *Valid: Visual quality assessment for light field images dataset.* 10th International Conference on Quality of Multimedia Experience (QoMEX), Sardinia, Italy, page 3, 2018. `https://mmspg.epfl.ch/downloads/valid/`. 31, 32, 86

[154] Paudyal, P., F. Battisti, M. Sjöström, R. Olsson, and M. Carli: *Toward the perceptual quality evaluation of compressed light field images.* IEEE Transactions on Broadcasting, PP(99):1–16, 2017, ISSN 0018-9316. 31, 32

[155] Rerabek, Martin and Touradj Ebrahimi: *New light field image dataset.* January 2016. 32

[156] Paudyal, P., R. Olsson, M. Sjostrom, F. Battisti, and M. Carli: *Smart: a light field image quality dataset.* In *Procs. of the ACM Multimedia Systems 2016 Conference, (MMSYS)*, 2016. 32, 94

[157] Boyce, Jill, Karsten Suehring, Xiang Li, and Vadim Seregin: *Jvet-j1010: Jvet common test conditions and software reference configurations*, July 2018. 32

[158] Institute, Fraunhofer Heinrich Hertz: *High efficiency video coding (hevc)*, aug 2022. `https://hevc.hhi.fraunhofer.de/`. 32

[159] Li, Yun, Mårten Sjöström, Roger Olsson, and Ulf Jennehag: *Scalable coding of plenoptic images by using a sparse set and disparities.* IEEE Transactions on Image Processing, 25(1):80–91, 2016. 32

[160] Freitas, Pedro Garcia, Sana Alamgeer, Welington Y. L. Akamine, and Mylène C. Q. Farias: *Blind image quality assessment based on multiscale salient local binary patterns.* In *Proceedings of the 9th ACM Multimedia Systems Conference*, MMSys '18, page 52–63, New York, NY, USA, 2018. Association for Computing Machinery, ISBN 9781450351928. `https://doi.org/10.1145/3204949.3204960`. 33

[161] Alamgeer, Sana, Muhammad Irshad, and Mylène CQ Farias: *Cnn-based no-reference video quality assessment method using a spatiotemporal saliency patch selection procedure*. Journal of Electronic Imaging, 30(6):063001, 2021. 33

[162] Zhang, Jianming and Stan Sclaroff: *Exploiting surroundedness for saliency detection: a boolean map approach*. IEEE transactions on pattern analysis and machine intelligence, 38(5):889–902, 2016. 34

[163] Fernández-Delgado, Manuel, Eva Cernadas, Senén Barro, and Dinani Amorim: *Do we need hundreds of classifiers to solve real world classification problems*. Journal of Machine Learning Research, 15(1):3133–3181, 2014. 34

[164] Freitas, Pedro Garcia, Welington Y. L. Akamine, and MylèneC. Q. Farias: *Blind image quality assessment using multiscale local binary patterns*. 2017. 35

[165] Sze, Vivienne, Madhukar Budagavi, Gary Sullivan, and Editors: *High Efficiency Video Coding (HEVC): Algorithms and Architectures*. July 2014. 35, 53, 105

[166] Pedregosa, Fabian, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, *et al.*: *Scikit-learn: Machine learning in python*. Journal of Machine Learning Research, 12(Oct):2825–2830, 2011. 36

[167] Hintze, Jerry L and Ray D Nelson: *Violin plots: a box plot-density trace synergism*. The American Statistician, 52(2):181–184, 1998. 37

[168] Bosse, S., D. Maniry, K. Müller, T. Wiegand, and W. Samek: *Deep neural networks for no-reference and full-reference image quality assessment*. IEEE Transactions on Image Processing, 27(1):206–219, Jan 2017, ISSN 1941-0042. 39

[169] Zhang, Lin, Zhongyi Gu, and Hongyu Li: *Sdsp: A novel saliency detection method by combining simple priors*. 2013 IEEE International Conference on Image Processing, pages 171–175, 2013. 40

[170] Marat, Sophie, Tien Ho Phuoc, Lionel Granjon, Nathalie Guyader, Denis Pellerin, and Anne Guérin-Dugué: *Modelling spatio-temporal saliency to predict gaze direction for short videos*. International journal of computer vision, 82(3):231, 2009. 40

[171] Farnebäck, Gunnar: *Two-frame motion estimation based on polynomial expansion*. pages 363–370, 2003. 40

[172] Zhang, Pengfei, Yu Cao, and Benyuan Liu: *Multi-stream single shot spatial-temporal action detection*. August 2019. 40

[173] Simonyan, Karen and Andrew Zisserman: *Two-stream convolutional networks for action recognition in videos*. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*, volume abs/1406.2199 of *NIPS'14*. MIT Press, 2014. 40

[174] Bampis, CG, Z Li, AK Moorthy, I Katsavounidis, A Aaron, and AC Bovik: *Live netflix video quality of experience database.* Online: http://live. ece. utexas. edu/research/LIVE _ NFLXStudy/index. html, 2016. 42

[175] Kingma, D., Ba J.: Adam: *A method for stochastic optimization.* arXiv preprint, 2014. 44

[176] Mahapattanakul, Puttatida: *From human vision to computer vision- how far off are we?*, 2019. `https://becominghuman.ai/from-human-vision-to-computer-vision`. 49

[177] Canny, J.: *A computational approach to edge detection.* IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8(6):679–698, 1986. 51

[178] Chotrov, Dimo, Zlatka Uzunova, Yordan Yordanov, and Stoyan Maleshkov: *Mixed-reality spatial configuration with a zed mini stereoscopic camera.* November 2018. 51

[179] Bolles, R., H. Baker, and D. Marimont: *Epipolar-plane image analysis: An approach to determining structure from motion.* International Journal of Computer Vision, 1:7–55, 2004. 51

[180] Tian, Yu, Huanqiang Zeng, Lu Xing, Jing Chen, Jianqing Zhu, and Kai Kuang Ma: *A multi-order derivative feature-based quality assessment model for light field image.* Journal of Visual Communication and Image Representation, 57:212 – 217, 2018, ISSN 1047-3203. 53, 70, 73, 77, 80, 105, 106

[181] Tian, Y., H. Zeng, J. Hou, J. Chen, and K. Ma: *Light field image quality assessment via the light field coherence.* IEEE Transactions on Image Processing, 29:7945–7956, 2020. 53

[182] Shi, L., S. Zhao, and Z. Chen: *Belif: Blind quality evaluator of light field image with tensor structure variation index.* In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 3781–3785, 2019. 53, 70, 73, 77, 80, 105, 106

[183] Xiang, J., M. Yu, H. Chen, H. Xu, Y. Song, and G. Jiang: *Vblfi: Visualization-based blind light field image quality assessment.* In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, 2020. 53, 70, 73

[184] Wang, Z. and Q. Li: *Information content weighting for perceptual image quality assessment.* IEEE Transactions on Image Processing, 20(5):1185–1198, 2011. 53

[185] Lin, Y. and J. Wu: *Quality assessment of stereoscopic 3d image compression by binocular integration behaviors.* IEEE Transactions on Image Processing, 23(4):1527–1542, 2014. 53

[186] Zhou Wang and A. C. Bovik: *A universal image quality index.* IEEE Signal Processing Letters, 9(3):81–84, 2002. 53

[187] Chen, Ming Jun, Che Chun Su, Do Kyoung Kwon, Lawrence K. Cormack, and Alan C. Bovik: *Full-reference quality assessment of stereopairs accounting for rivalry.* Signal Processing: Image Communication, 28(9):1143 – 1155, 2013, ISSN 0923-5965. `http://www.sciencedirect.com/science/article/pii/S0923596513000787`. 53

[188] Mittal, A., A. K. Moorthy, and A. C. Bovik: *No-reference image quality assessment in the spatial domain*. IEEE Transactions on Image Processing, 21(12):4695–4708, 2012. 53

[189] Vu, P. V., C. T. Vu, and D. M. Chandler: *A spatiotemporal most-apparent-distortion model for video quality assessment*. In *2011 18th IEEE International Conference on Image Processing*, pages 2505–2508, 2011. 53

[190] Chollet, Francois: *Keras*, 2021. `https://keras.io/`. 53, 62, 70, 78, 87, 95, 105

[191] JPEG, ISO/IEC JTC 1/SC29/WG1: *Verification model software version 2.1 on jpeg pleno light field coding*. Technical Report Doc. N83034, ISO/IEC JTC 1/SC29/WG1, mar 2019. 59

[192] Alamgeer, Sana, Muhammad Irshad, and Mylène C. Q. Farias: *Light field image quality assessment method based on deep graph convolutional neural network: Research proposal*. In *Proceedings of the 13th ACM Multimedia Systems Conference*, MMSys '22, page 357–361, New York, NY, USA, 2022. Association for Computing Machinery, ISBN 9781450392839. 66

[193] Liu, Shuying and Weihong Deng: *Very deep convolutional neural network based image classification using small training sample size*. In *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*, pages 730–734, 2015. 69, 76

[194] Yang, Kaiyu, Klint Qinami, Li Fei-Fei, Jia Deng, and Olga Russakovsky: *Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the imagenet hierarchy*. FAT* '20, New York, NY, USA, 2020. Association for Computing Machinery, ISBN 9781450369367. `https://doi.org/10.1145/3351095.3375709`. 69, 76

[195] PhiCong, Huy, Stuart Perry, Eva Cheng, and Xiem HoangVan: *Objective quality assessment metrics for light field image based on textural features*. Electronics, 11(5), 2022, ISSN 2079-9292. `https://www.mdpi.com/2079-9292/11/5/759`. 72

[196] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun: *Deep residual learning for image recognition*. CoRR, abs/1512.03385, 2015. `http://arxiv.org/abs/1512.03385`. 76

[197] Chollet, François: *Xception: Deep learning with depthwise separable convolutions*. CoRR, abs/1610.02357, 2016. `http://arxiv.org/abs/1610.02357`. 76

[198] ITU-T: *Subjective video quality assessment methods for multimedia applications*. Recommendation ITU-T P.910, 2008. 78

[199] Chen, Liang Chieh, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan Yuille: *Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs*. IEEE Transactions on Pattern Analysis and Machine Intelligence, PP, June 2016. 85

[200] Harris, Christopher G. and M. J. Stephens: *A combined corner and edge detector*. In *Alvey Vision Conference*, 1988. 101

[201] Zhang, T. Y. and C. Y. Suen: *A fast parallel algorithm for thinning digital patterns*. Commun. ACM, 27(3):236–239, mar 1984, ISSN 0001-0782. https://doi.org/10.1145/357994.358023. 101

[202] Brandes, Ulrik: *A faster algorithm for betweenness centrality*. The Journal of Mathematical Sociology, 25:163 – 177, 2001. 101

[203] Monti, Federico, Fabrizio Frasca, Davide Eynard, Damon Mannion, and Michael M. Bronstein: *Fake news detection on social media using geometric deep learning*. CoRR, abs/1902.06673, 2019. http://arxiv.org/abs/1902.06673. 103

[204] Xiang, J., M. Yu, G. Jiang, H. Xu, Y. Song, and Y. S. Ho: *Pseudo video and refocused images based blind light field image quality assessment*. IEEE Transactions on Circuits and Systems for Video Technology, pages 1–1, 2020. 105, 106

[205] Pan, Z., M. Yu, G. Jiang, H. Xu, and Y. S. Ho: *Combining tensor slice and singular value for blind light field image quality assessment*. IEEE Journal of Selected Topics in Signal Processing, pages 1–1, 2021. 105, 106

Appendix A

Papers Resulting From This Thesis

A.1 Conference Papers

1. *Blind image quality assessment based on multiscale salient local binary patterns* - In Proceedings of the 9th ACM Multimedia Systems Conference - 2018.
2. *Perceptual quality assessment of enhanced images using a crowd-sourcing framework* - IS&T International Symposium on Electronic Imaging - 2020.
3. *Image Quality Assessment of Underwater Images Using Multi-Scale Salient Local Binary Patterns* - IS&T International Symposium on Electronic Imaging - 2021.
4. *Light field image quality assessment method based on deep graph convolutional neural network: research proposal* - In Proceedings of the 13th ACM Multimedia Systems Conference - 2022.

A.2 Journal Papers

1. *A Two-stream HVS-CNN based Visual Quality Assessment Method for Light Field Images* - Journal of Multimedia Tools And Applications (MTAP) - 2022.
2. *CNN-based no-reference video quality assessment method using a spatiotemporal saliency patch selection procedure* - Journal of Electronic Imaging (SPIE) - 2021.

A.3 Accepted Papers

1. *No-Reference Light Field Image Quality Assessment Method Based on a Long-Short Term Memory Neural Network* - IEEE International Conference on Multimedia And Expo (ICME) - 2022.

2. *Light Field Image Quality Assessment with Dense Atrous Convolutions* - IEEE International Conference in Image Processing (ICIP) - 2022.
3. *Deep Learning-Based Light Field Image Quality Assessment Using Frequency Domain Inputs* - The 14th International Conference on Quality of Multimedia Experience (QoMEX) - 2022.
4. *Designing a user-centric solution for perceptually-efficient streaming of 360-degree edited videos* - IS&T International Symposium on Electronic Imaging - 2022.

A.4 First Page of Published Papers

Blind Image Quality Assessment Based on Multiscale Salient Local Binary Patterns

Pedro Garcia Freitas, Sana Alamgeer, Welington Y.L. Akamine, and Mylène C.Q. Farias
Dept. of Electrical Engineering and Dept. of Computer Science
University of Brasília
Brasília, DF, Brazil
{pedrogarcia,mylene}@ieee.org,{sanaalamgeer,welingtonylakamine}@gmail.com

ABSTRACT

Due to the rapid development of multimedia technologies, over the last decades image quality assessment (IQA) has become an important topic. As a consequence, a great research effort has been made to develop computational models that estimate image quality. Among the possible IQA approaches, blind IQA (BIQA) is of fundamental interest as it can be used in most multimedia applications. BIQA techniques measure the perceptual quality of an image without using the reference (or pristine) image. This paper proposes a new BIQA method that uses a combination of texture features and saliency maps of an image. Texture features are extracted from the images using the local binary pattern (LBP) operator at multiple scales. To extract the salient areas of an image, i.e. the areas of the image that are the main attractors of the viewers' attention, we use computational visual attention models that output saliency maps. These saliency maps can be used as weighting functions for the LBP maps at multiple scales. We propose an operator that produces a combination of multiscale LBP maps and saliency maps, which is called the multiscale salient local binary pattern (MSLBP) operator. To define which is the best model to be used in the proposed operator, we investigate the performance of several saliency models. Experimental results demonstrate that the proposed method is able to estimate the quality of impaired images with a wide variety of distortions. The proposed metric has a better prediction accuracy than state-of-the-art IQA methods.

CCS CONCEPTS

• **Information systems** → **Multimedia streaming**; • **General and reference** → *Metrics*; • **Computing methodologies** → Machine learning approaches;

MMSys'18, June 12–15, 2018, Amsterdam, Netherlands

ACM ISBN 978-1-4503-5192-8/18/06.
https://doi.org/10.1145/3204949.3204960

KEYWORDS

Quality Metrics, Performance Assessment, Blind Image Quality Assessment, Quality of Experience

ACM Reference Format:
Pedro Garcia Freitas, Sana Alamgeer, Welington Y.L. Akamine, and Mylène C.Q. Farias. 2018. Blind Image Quality Assessment Based on Multiscale Salient Local Binary Patterns. In *MMSys'18: 9th ACM Multimedia Systems Conference, June 12–15, 2018, Amsterdam, Netherlands*. ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3204949.3204960

1 INTRODUCTION

The popularity of multimedia services over the Internet have changed users' requirements, specially in terms of quality. In a recent report, Conviva® has shown that viewers are demanding a delivered multimedia content with a higher quality [9]. In the context of images and videos, higher quality content generally corresponds to larger file sizes, which demands higher network bandwidth and storage space. In fact, as predicted by Cisco® [8], today, most Internet traffic corresponds to multimedia content. It is worth pointing out that the quality of a multimedia content can be altered in any stage of the communication chain, such as capture, compression, transmission, reproduction, and display. As users' demands for a higher quality of content increases, it is important to design automatic tools that are able to predict the quality of the visual stimuli in any of these stages. Therefore, there is currently a great need for techniques that automatically estimate image and video quality in multimedia applications.

Objective image quality assessment (IQA) methods measure image quality using computer algorithms instead of human beings. For instance, mean squared error (MSE) and peak-to-noise ratio (PSNR) are fidelity metrics that can be used to measure the similarity of images with same content and distortion type. Nevertheless, MSE and PSNR scores often do not correlate well with subjective scores, i.e. with the perceived image quality [49]. For an objective method to be used in multimedia applications, its estimates must be well correlated with quality scores from publicly available image quality databases, which are collected by performing psychophysical experiments (with human subjects). These experiments use standardized experimental methodologies to obtain quality scores for a broad range of images processed with a diverse number of algorithms and procedures.

Objective image quality metrics can be classified according to the amount of available reference information (original or

CNN-based no-reference video quality assessment method using a spatiotemporal saliency patch selection procedure

Sana Alamgeer,* Muhammad Irshad, and Mylène C. Q. Farias
University of Brasília, Department of Electrical Engineering, Brasília, Brazil

Abstract. We propose a yet lightweight no-reference (NR) video quality assessment (VQA) method, which uses a convolution neural network (CNN) architecture. The proposed method implements a spatiotemporal saliency patch selection procedure that crops the frame into small nonoverlapping blocks of images (patches) and selects the most perceptually relevant ones. The selected patches are then forwarded to the CNN. To determine which patches are the most relevant, spatial and temporal saliency features are computed for each frame. The proposed method does not require subjective scores to train the CNN. It uses objective quality scores as target quality scores for each video frame, which are computed using an NR image quality assessment method. Given the lack of large annotated video quality databases, this is an advantage of the proposed method. Finally, although it has much smaller cost of data-processing, compared with other state-of-the-art methods, the proposed NR-VQA obtains robust and competitive results. © *2021 SPIE and IS&T* [DOI: 10.1117/1.JEI.30.6.063001]

Keywords: video quality assessment; saliency; convolution neural network; objective quality scores.

Paper 210107 received Feb. 28, 2021; accepted for publication Oct. 13, 2021; published online Nov. 1, 2021.

1 Introduction

In the last decades, there has been a tremendous increase in the popularity of video applications, with 82% of the internet traffic being currently video data.[1] Since the success or popularity of a video service is correlated to the quality of experience of the end user,[2] it is often important to assess the quality of the video signal at the client side, and the quality assessment is performed using quality assessment methods.

Quality assessment methods are algorithms that estimate the quality of videos (VQA) or images (IQA) either objectively or subjectively. Subjective quality assessment methods estimate the quality of images/videos by performing psychophysical experiments, where participants assign a score to each image/video. An estimate of the quality is given by the mean observer score (MOS), which is computed by averaging the scores given to a test image/video by all participants. Although subjective image/video quality assessment (VQA) methods are considered as ground-truth in image/video quality, these methods are expensive and time consuming. Objective image/VQA methods, on the other hand, estimate the quality of image/video using computational algorithms (quality metrics), which are faster, cheaper, and can be more easily incorporated in a multimedia application. In this work, henceforth, we use acronyms IQA and VQA to refer to objective image quality assessment (IQA) methods and video quality assessment methods, respectively.

VQA methods can be classified as full-reference (FR), reduced-reference (RR), and no-reference (NR) methods. FR VQA methods, which require the reference (pristine) content, are frequently the best performing metrics.[3] RR VQA methods require sending (or embedding) features from the original content to the receiver/user,[4] whereas NR VQA methods estimate quality blindly without having access to the original.[5] Unfortunately, both FR and RR metrics cannot be used in real-time applications where the reference or even a small amount of the reference

*Address all correspondence to Sana Alamgeer, sanaalamgeer@gmail.com

https://doi.org/10.2352/ISSN.2470-1173.2020.9.IQSP-066

Perceptual Quality Assessment of Enhanced Images Using a Crowd-Sourcing Framework

Muhammad Irshad*[1] *, Alessandro R. Silva*[2] *, Sana Alamgeer*[1] *, Mylène C.Q. Farias*[1] *;
[1] ***Department of Electrical Engineering,*** [2] ***Department of Computer Science,***
University of Brasilia, Brazil.

Abstract

In this work, we present a psychophysical study, in which, we analyzed the perceptual quality of images enhanced with several types of enhancement algorithms, including color, sharpness, histogram, and contrast enhancements. To estimate and compare the qualities of enhanced images, we performed a psychophysical experiment with 35 source images, obtained from publicly available databases. More specifically, we used images from the Challenge Database, the CSIQ database, and the TID2013 database. To generate the test sequences, we used 12 different image enhancement algorithms, generating a dataset with a total of 455 images. We used a Double Stimulus Continuous Quality Scale (DSCQS) experimental methodology, with a between-subjects approach where each subject scored a subset of the total database to avoid fatigue. Given the high number of test images, we designed a crowd-sourcing interface to perform an online psychophysical experiment. This type of interface has the advantage of making it possible to collect data from many participants. We also performed an experiment in a controlled laboratory environment and compared its results with the crowd-sourcing results. Since there are very few quality enhancement databases available in the literature, this works represents a contribution to the area of image quality.

Keywords: Enhancement; Perceptual Quality Assessment; Crowd-Sourcing Framework, Subjective Quality Assessment.

Introduction

Image enhancement is frequently used to improve or restore the visual quality of images and videos. Currently, there are several image enhancement algorithms, but there is not yet a performance metric that is able to estimate the performance of these methods. Since the final consumers of the resulting enhanced visual content are human viewers, the performance of these algorithms should be measured by estimating the visual quality of the enhanced images, taking into consideration the human visual system [20].

Image quality can be estimated using subjective (psychophysical experiments) and objective (quality metrics) methods [9, 21]. Subjective methods are simply psychophysical experiments where participants rate one or more aspects of a set of processed images. Most often, these experiments are performed in a controlled environment (e.g. a laboratory), following standard recommendations for the environment conditions and experimental methodologies [6]. It worth pointing out that although data (subjective scores) collected in psychophysical experiments are considered as ground-truth, these experiments are time-consuming and expensive. Objective quality methods, on the other hand, are algorithms (implemented in hardware or software) that automatically estimate the quality of an image [14, 10]. These methods are designed and tested using subjective scores as ground-truth.

The area of image and video quality has achieved great progress in the last decades [2]. But, although the performance accuracy of quality metrics has improved, there are still many challenges in this area. Among them is the design of objective quality metrics for enhanced contents. Since most of the quality metrics have been designed to capture visual distortions, they are not able to quantify the changes in quality introduced by enhancement algorithms. Therefore, currently, there is a need for quality metrics that can automatically estimate the quality of enhanced images and videos. It is worth pointing out that developing quality metrics for enhanced images is a challenge due to the lack of quality databases containing enhanced images and their respective (ground-truth) subjective quality scores.

In this paper, our goal is to introduce a quality database for enhanced images. Up to our knowledge, currently, there is only one image enhancement quality database that can be used for research in image quality [19]. However, this database contains images of low resolution that were processed manually, using a professional graphics editing software (Adobe Photoshop) to produce the best possible enhanced images. In our database, we used images of a higher resolution, which are enhanced with twelve different image enhancement algorithms. Our goal was to produce a set of images that were like consumer applications contents. Also, we performed a crowd-sourcing subjective experiment to obtain quality scores for all database images. With this experiment, we were able to obtain a large and diverse pool of participants.

Database Content Generation

Figure 1 shows a block diagram of the strategy used to generate the database. Our first step was to choose 35 original (source - SRC) images. These images were taken from three image quality databases, to allow for future comparisons of enhanced and degraded images. More specifically, we took 5 SRC images from the CSIQ database [18], 5 original images from the TID2013 database [16], and 25 original images from the ChallengeDB database [3]. Table 6 (in the Appendix) shows a list of the SRC images, along with their names in the corresponding databases. These chosen source contents are diverse, in terms of spatial activity, semantic content, and color distribution. In Figure 2, the first row (SRCs) shows examples of SRC images taken from the (a-b) TID2013, (c-d) CSIQ, and (e-f) ChallengeDB databases.

Our next step consists of choosing the enhancement algo-

https://doi.org/10.2352/ISSN.2470-1173.2021.9.IQSP-265

No-reference Image Quality Assessment of Underwater Images Using Multi-Scale Salient Local Binary Patterns

Muhammad Irshad[1], Camilo Sanchez-Ferreira[2], Sana Alamgeer[1], Carlos H. Llanos[3], and Mylène C.Q. Farias[1]

[1]Department of Electrical Engineering, University of Brasília, Brazil.
[2]Department of Physics, University of Cauca, Colombia.
[3]Department of Mechanical Engineering, University of Brasília, Brazil.

Abstract

Images acquired in underwater scenarios may contain severe distortions due to light absorption and scattering, color distortion, poor visibility, and contrast reduction. Because of these degradations, researchers have proposed several algorithms to restore or enhance underwater images. One way to assess these algorithms' performance is to measure the quality of the restored/enhanced underwater images. Unfortunately, since reference (pristine) images are often not available, designing no-reference (blind) image quality metrics for this type of scenario is still a challenge. In fact, although the area of image quality has evolved a lot in the last decades, estimating the quality of enhanced and restored images is still an open problem. In this work, we present a no-reference image quality evaluation metric for enhanced underwater images (NR-UWIQA) that uses an adapted version of the multi-scale salient local binary pattern operator to extract image features and a machine learning approach to predict quality. The proposed metric was tested on the UID-LEIA database and presented good accuracy performance when compared to other state-of-the-art methods. In summary, the proposed NR-UWQIA method can be used to evaluate the results of restoration techniques quickly and efficiently, opening a new perspective in the area of underwater image restoration and quality assessment.

Keywords: Underwater image enhancement; Image quality assessment; Quality metrics, full-reference, no-reference; Underwater image formation model; Saliency; Multiscale Salient Local Binary Patterns;

Introduction

Underwater images are often characterized by a poor visibility since the light travelling in the water medium is attenuated and, consequently, the captured scenes may be poorly contrasted and hazy. More specifically, light attenuation is produced by absorption and scattering processes. Absorption removes the light energy while scattering changes the direction of the light. Therefore, underwater images may have different types of degradations, including limited-range visibility, non-uniform lightening, low contrast, blurring, diminished color, bright artifacts, and noise. In other words, the visual aspect of underwater images may vary a lot depending on the water medium's characteristics, including the types of particles present in the water and the water depth [26]. Figure 1 shows examples of images captured underwater in three different scenarios: shallow water, deep water, and muddy water. Notice that, generally, degradations of images captured underwater are stronger than degradations of images captured over-the-air [39]. Often the quality of underwater images is not adequate for the to be used by image and computer vision algorithms, requiring the use of restoration or enhacement algorithms [39].

Given the importance of the overall quality of underwater-captured images for ocean engineering and scientific research, there are in the literature several methods for restoring or enhancing the quality of underwater images [16, 19]. Therefore, the use of underwater images in computer vision and image processing applications often depends on the success restoration and enhancement algorithms [35, 3, 15]. To determine the performance of these algorithms, we must estimate the quality of the restored/enhanced images as perceived by human viewers. Unfortunately, most methods used to estimate the performance of these algorithms do not consider human perception or image quality. One of the reasons is that subjective quality experiments, which are considered as the ground truth in image quality research, are costly and time-consuming [21]. Moreover, these methods are unfeasible for real-time applications and system integration. One viable option to estimate the quality of restored or enhanced underwater images and, therefore, the restoration algorithm's performance is to use objective image quality assessment (IQA) methods.

IQA methods are algorithms capable of automatically estimate the quality of an image. These methods can be divided into three classes: (a) full-reference (FR) IQA methods, where a reference image is needed to estimate the quality; (2) reduced reference (RR) IQA methods, where partial information about the reference image is available; or (3) no-reference (NR) IQA methods, which blindly estimate quality without having access to the reference or pristine image. For underwater images scenario, where a reference image is not available, we must use NR-IQA methods to estimate the perceptual quality of restored and degraded images. IQA methods can be used to evaluate the restoration process's success and determine if the images are adequate for the target underwater engineering and monitoring applications. So far, a few researchers have proposed IQA methods specifically for underwater images. For example, Sanchez *et al.* [37] have proposed a restoration algorithm for underwater that uses an NR-IQA method as a performance metric for the optimization algorithm.

Although in the last decades a lot of progress has been made in the area of image quality assessment, designing metrics to estimate the quality of enhanced and restored images remains a challenge [8]. As mentioned earlier, the final quality of underwater images depend on the marine habitats where the images are captured, which often introduce specific chroma, saturation, and con-

Multimedia Tools and Applications
https://doi.org/10.1007/s11042-022-13436-4

A two-stream cnn based visual quality assessment method for light field images

Sana Alamgeer[1] · **Mylène C.Q. Farias**[1]

Received: 4 July 2021 / Revised: 23 March 2022 / Accepted: 2 July 2022

Abstract
Light Field (LF) cameras are able to capture both the intensity and the direction of light rays from the scene. This rich information demands a certain amount of memory and bandwidth for storage and transmission and, to alleviate this requirement, the LF content is processed and compressed. These operations often add degradations to the LF content that may affect their visual quality, requiring the use of methods to estimate the visual quality as perceived by the end consumer. In this paper, we propose a no-reference LF image quality assessment (LF-IQA) method that is based on a two-stream CNN architecture. The two-stream CNN extracts rich distortion-related spatial and angular binocular characteristics of LF contents to estimate their quality. More specifically, the first stream extracts angular information by processing Canny maps of Epipolar Plane Images (EPIs) generated from the corresponding LF contents, while the second stream extracts spatial information by processing mean canny maps generated from canny maps of sub-aperture images (SAIs). We also propose a novel approach to generate multiple epipolar-plane images - the MultiEPL. Results show that the proposed LF-IQA method outperforms state-of-the-art methods.

Keywords Image quality assessment · Epipolar planes · Canny edge detector · Two-stream convolution neural network · 4D Light field images

1 Introduction

A Light Field Image (LFI) describes the set of light rays traveling in angular direction at every point in 3-Dimensional (3D) space. LFI is defined by a 4D plenoptic function $L(u, v, s, t)$, where (u, v) and (s, t) represent the angular and spatial domains, respectively. The spatial and angular domains provide parallax and depth information that allow performing adjustments after the image is captured. Given these properties, light field (LF)

✉ Sana Alamgeer
sanaalamgeer@gmail.com

Mylène C.Q. Farias
mylene@ieee.org

[1] Department of Electrical Engineering, University of Brasília, Brasília, Brazil

Published online: 30 July 2022

Research Proposal: Light Field Image Quality Assessment Method based on Deep Graph Convolutional Neural Network

Sana Alamgeer*, Muhammad Irshad, , and Mylène C.Q. Farias
sanaalamgeer@gmail.com*
Department of Electrical Engineering
University of Brasília
Brazil

ABSTRACT

This paper contains the research proposal of Sana Alamgeer that was presented at the MMSys 2022 doctoral symposium. Unlike regular images that represent only light intensities, Light Field (LF) contents carry information about the intensity of light in a scene, including the direction light rays are traveling in space. This allows for a richer representation of our world, but requires large amounts of data that need to be processed and compressed before being transmitted to the viewer. Since these techniques may introduce distortions, the design of Light Field Image Quality Assessment (LF-IQA) methods is important. The majority of LF-IQA methods based on traditional Convolutional Neural Network (CNN) have limitations, i.e. they are unable to increase the receptive field of a neuron-pixel to model non-local image features. In this work, we propose a novel no-reference LF-IQA method that is based on Deep Graph Convolutional Neural Network (GCNN). Our method not only takes into account both LF angular and spatial information, but also learns the order of pixel information. Specifically, the method is composed of one input layer that takes a pair of graphs and their corresponding subjective quality scores as labels, 4 GCNN layers, fully connected layers, and a regression block for quality prediction. Our aim is to develop the quality prediction method with maximum accuracy for distorted LF content.

CCS CONCEPTS

• **Software and its engineering**; • **Computing methodologies** → **Spectral methods**;

KEYWORDS

4D Light Field, Image Quality Assessment, Graph Convolutional Neural Network, Epipolar Plane Images

ACM Reference Format:
Sana Alamgeer*, Muhammad Irshad, , and Mylène C.Q. Farias. 2022. Research Proposal: Light Field Image Quality Assessment Method based on Deep Graph Convolutional Neural Network. In *13th ACM Multimedia Systems Conference (MMSys '22), June 14–17, 2022, Athlone, Ireland.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3524273.3533927

MMSys '22, June 14–17, 2022, Athlone, Ireland

ACM ISBN 978-1-4503-9283-9/22/06…$15.00
https://doi.org/10.1145/3524273.3533927

1 INTRODUCTION

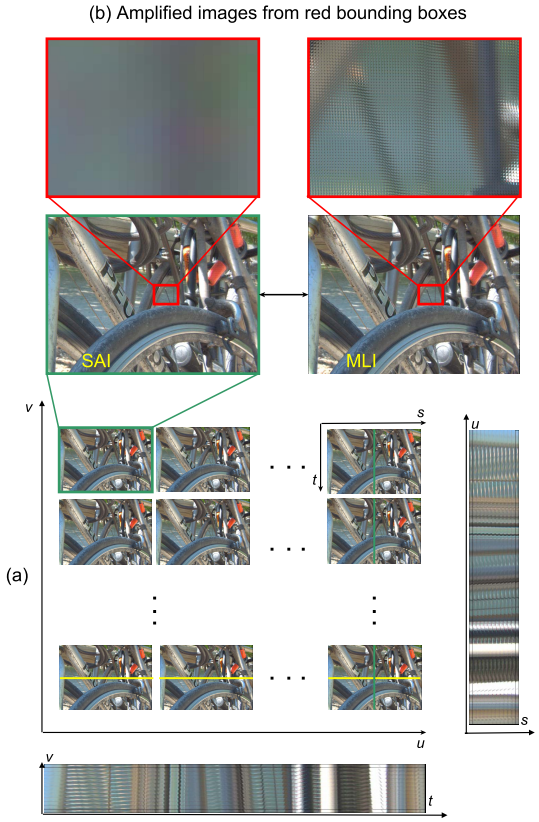


Figure 1: Illustration of MLI, SAI and EPI. (a) A 12×12 grid of 144 SAIs of an LFI (8bit-HEVC-I01-I01P1R1) from VALID dataset [15] with corresponding Vertical (extracted from green line) and Horizontal (extracted from yellow line) EPIs. (b) MLI of dimension 8138×5642 and SAI of dimension 626×434 with amplified images generated from red bounding boxes.

The advancement of imaging technologies has produced plenoptic devices that can capture and display visual information to describe objects in the 3D space from any point-of-view. Depending on the capturing device, this visual information can correspond to holograms, light fields (LF), or point clouds imaging formats. In the particular case of LF contents, the cameras [4, 16] capture both angular