GraphIQA: Learning Distortion Graph Representations for Blind Image Quality Assessment

Simeng Sun*, Tao Yu*, Jiahua Xu, Wei Zhou and Zhibo Chen, *Senior Member, IEEE,*

***Abstract*—A good distortion representation is crucial for the success of deep blind image quality assessment (BIQA). However, most previous methods do not effectively model the relationship between distortions or the distribution of samples with the same distortion type but different distortion levels. In this work, we start from the analysis of the relationship between perceptual image quality and distortion-related factors, such as distortion types and levels. Then, we propose a Distortion Graph Representation (DGR) learning framework for IQA, named GraphIQA, in which each distortion is represented as a graph, *i.e.*, DGR. One can distinguish distortion types by learning the contrast relationship between these different DGRs, and can infer the ranking distribution of samples from different levels in a DGR. Specifically, we develop two sub-networks to learn the DGRs: a) Type Discrimination Network (TDN) that aims to embed DGR into a compact code for better discriminating distortion types and learning the relationship between types; b) Fuzzy Prediction Network (FPN) that aims to extract the distributional characteristics of the samples in a DGR and predicts fuzzy degrees based on a Gaussian prior. Experiments show that our GraphIQA achieves state-of-the-art performance on many benchmark datasets of both synthetic and authentic distortions. The code is available at http://staff.ustc.edu.cn/ chenzhibo/resources/2021/GraphIQA.html.**

***Index Terms*—blind image quality assessment, graph representation learning, and pre-training.**

I. Introduction

With the rapid development of social networks, a massive amount of digital images have been produced. They could be distorted in any stage of the whole media technical chain, from acquisition, processing, compression to transmission and consumption. Therefore, a reliable image quality assessment (IQA) metric is critical for measuring multimedia model results and guiding its optimization.

Within the scope of IQA, no-reference or blind image quality assessment (NRIQA or BIQA) has drawn much attention since the references are often not available in many real-world applications. Meanwhile, learning-based BIQA methods perform well thanks to the powerful fitting capacity of deep neural networks [1]–[9].

A good representation could help the training of the target task [10]. Particularly, in situations where labeled data are hard to reach, the representations obtained by unsupervised [11]–[13] or semi-supervised [4], [14], [15] learning can serve as auxiliary information to solve the supervised learning tasks. Recently, the research on representation learning has helped to make breakthroughs in various fields [10], [11], [14], [16]. In IQA, the improvement of performance and model generalization ability is also inseparable from the efficient representations of distorted data [4], [6]–[8], [15], [17].

Many methods improve IQA model performance by learning a good representation of distortion, so as to better serve the quality score regression. One common approach is introducing an auxiliary distortion classification task in latent space to enforce the feature representations to be discriminative to distortion types [7], [8], [18], which is one of the important factors affecting image perceptual quality. Although such type classification task can assist in IQA tasks, the representations obtained by these methods may suffer from at least two issues: 1) they cannot distinguish the level of distortion, which is also an important factor in image perceptual quality; 2) they are not robust when being adapted to the IQA task for authentic distortion due to the uncertainty of distortion type and non-homogeneity of the authentic distortions. To address the first issue, Zhang *et al.* [4] propose to employ an extra distortion-level classification task. However, they ignore intrinsic distribution properties among distortion levels. For example, assuming a scene where there are three images with distortion level-1, 2 and 5 respectively, the methods based on classification task fail to model their ranking relationship. As the levels are treated as independent categories, the ranking relationship, where level-1 samples are more similar to level-2 samples than level-5 samples, can't be discriminated. Xu *et al.* [17] address the issue by designing a rank model for each distortion to learn the ranking relationship among levels. This method cannot efficiently handle unseen distortion types as there is no corresponding rank model for this type. Then Liu *et al.* [15] propose a siamese network to learn to rank two images sampled from the same distortion. However, it ignores modeling the distortion type. For the second issue, Zhang *et al.* [4] attempt to directly perform bilinear pooling of the synthetic and authentic feature sets to achieve better performance on the two kinds of distorted data simultaneously. However, in this method, two pre-trained networks are required to handle synthetic and authentic distortions at the same time. Besides, the computational complexity of bilinear pooling for fusing the two features is also high.

In this work, we model the relationship between distortion type and distortion level as a hierarchical model based on our observations (details are described in Section III-A) and the conclusions in the mentioned work [4], [15], [17]. That is, learning to rank the samples from specific distortion types

* Equal contribution.

Simeng Sun, Tao Yu, Jiahua Xu, Wei Zhou and Zhibo Chen are with the Department of Electronic Engineer and Information Science, University of Science and Technology of China, Hefei, Anhui, 230026, China (e-mail: smsun20@mail.ustc.edu.cn; yutao666@mail.ustc.edu.cn; skylarxu@tencent.com; weichou@mail.ustc.edu.cn; chenzhibo@ustc.edu.cn).

Corresponding Author: Zhibo Chen.

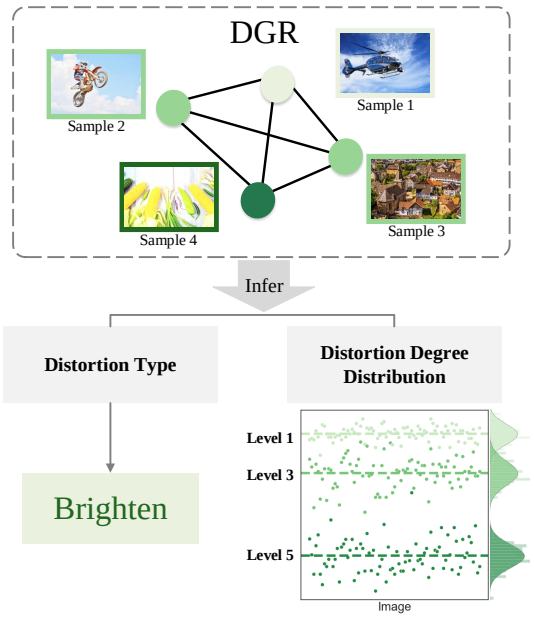


Fig. 1: The core idea of GraphIQA. We develop DGR to represent each distortion. The DGR can be utilized to infer distortion type and level based on its internal structure. The DGR learning also considers the rating deviation of image content for better prediction. The learned DGRs with plentiful distortion prior can help improve IQA accuracy.

and discriminate their levels are beneficial for obtaining better representations for the IQA task. Therefore we introduce graph representation, which is suitable for modeling the hierarchical structure when giving proper definitions of node and edge. In detail, each graph itself is used to represent a particular distortion type, while its node distribution in a specific graph is used to represent different distortion levels. In addition, in order to make the learned representations robust across distortion types (including synthetic distortion, authentic distortion, and multiple distortions), we propose to learn the relationships between distortion types by drawing on metric learning methods. Overall, the proposed method is a two-stage method. In the pre-training stage, we explore modeling distortion types and levels with a single high-efficient model and learn better distortion representations from distortion contrast relationships and their internal distributions. In the finetuning stage, the learned representations of distortion are used to assist IQA task on the target dataset. To this end, we propose a novel BIQA framework that integrates graph representation learning, dubbed GraphIQA.

The proposed GraphIQA model is trained to build the distortion graph representation (DGR) for each specific distortion. In each DGR, the nodes represent the feature of samples and the edges illustrate their correlation. The core idea of GraphIQA is shown in Fig. 1, where DGR is constructed from two aspects: (a) distinguishing the distortion type by contrasting the DGRs of different distortions; (b) predicting the most likely distortion level of a distorted image according to the internal topological relationship in each DGR. To achieve these two goals, we correspondingly design Type Discrimination Network (TDN) and Fuzzy Prediction Network (FPN) to learn the DGRs respectively. In detail, the TDN encodes DGR to a low-dimensional code to distinguish distortion types by aggregating the global information of nodes and the relationship between them. Specifically, it discriminates each distortion type and learns a robust representation of relationship between types by enforcing a triplet loss [19] on the top of the extracted code. Then the FPN extracts the distributional characteristics of the samples in DGR and predicts fuzzy levels based on a Gaussian prior considering the subjective quality ratings are often biased by image content. The visualization experiments show that the learned DGRs can model the relationship between perceptual image quality and distortion-related factors. Benefit from the DGR GraphIQA achieves the state-of-the-art performance on the most of the typical synthetic distorted IQA datasets (*e.g.* after finetuning, LIVE [20] and CSIQ [21]). The experiments also demonstrate that GraphIQA can be migrated to multiply distorted data (*e.g.*, LIVEMD [22]) and authentic distorted data (*e.g.*, KonIQ-10k [23] or (LIVEC) [24])), and obtain better performance. Note that the data used for pre-training DGRs is within easy reach as it only requires synthetic distorted data and their labels of distortion type and level for weakly supervised training. Our contributions can be summarized as follows:

- We investigate the inherent relationship between distortion-related factors and their effects on perceptual quality and propose an effective Distortion Graph Representation (DGR) learning framework dubbed GraphIQA for general-purpose BIQA task.
- To encourage better graph representation learning for the relationship modeling of distortion-related factors in GraphIQA, we well design a Type Discrimination Network (TDN) and a Fuzzy Prediction Network (FPN) to learn the proposed DGR.
- The proposed DGR can be conveniently applied in most downstream IQA tasks including IQA of synthetic distortion, authentic distortion and multiply distortion, and help GraphIQA achieve the state-of-the-art performance.

The rest of the manuscript is arranged as follows. Recent progress on blind image quality assessment and graph representation learning are introduced in Section II. The motivation and details of the proposed GraphIQA framework are introduced in Section III, and corresponding experiments are illustrated in Section IV. We conclude this paper in the last section.

II. Related Work

A. Blind Image Quality Assessment

Blind Image Quality Assessment(BIQA) can be categorized into distortion-specific methods [25]–[28] and general-purpose algorithms [29]–[43]. The distortion-specific BIQA methods are favored for their higher accuracy and robustness, when distortion types or distortion process is already known. However, their application scope is limited, as the authentic distortion dataset is mixed with complex distortions and the type of distortion is not clearly specified [22], [24]. Therefore,

the research on general-purpose methods has become particularly important and received extensive attention recently. Natural scene statistics (NSS) is one of the powerful tools for general-purpose BIQA, as quality degradations can cause deviation from the originally statistical properties of natural scene images [30]–[32], [35], [44]–[46]. For example, Saad *et al.* [30] leverage the statistics of local DCT coefficients as the feature for image quality assessment, while Moorthy *et al.* [47] leverage the feature obtained from the wavelet transform. To simplify the process of feature extraction, Mittal *et al.* [31] propose the method using the NSS in the spatial domain directly. And Zhang *et al.* [32] leverage not only the statistics of the mean subtracted contrast normalized coefficients, but also the statistics of gradients.

Recently, benefit from its ability to efficiently and adaptively extract distortion-aware features, the deep learning-based general-purpose BIQA methods have drawn considerable attention [4], [7], [15], [17], [18], [48]–[52]. Kim *et al.* [48] propose an efficient approach and prove that using backbone pre-trained on large classification dataset ImageNet [49] can improve the performance of IQA. Based on this, Talebi *et al.* [50] propose a DCNNs-based model to predict the perceptual distribution of IQA scores instead of the mean value. Similarly, Zeng *et al.* [51] propose the probabilistic quality representation to describe the image subjective score distribution. Noticing that, in synthetic distortion data, effectively utilizing distortion-related information is a common approach to help the learning of representation for IQA task, Kang *et al.* [18] introduce a compact multi-task network into IQA in which type identification task and IQA share all the internal structure. Ma *et al.* [7] introduce a two-stage training strategy, where a distortion type identification sub-network is first trained, and then a sub-network for IQA task is added. Though multi-task related methods have brought progress in IQA, this is difficult to be utilized on authentically distorted datasets, as the representation learned by type classification task can't handle totally unseen distortion types. For better performance on both synthetic and authentic distortion, Zhang *et al.* [4] combine two sub-networks, one of which is trained on type classification task aimed to extract features to represent synthetic distortion, the other of which is ImageNet pre-trained model aimed to extract semantic features. Then two kinds of features are fused by bilinear pooling to predict the subjective quality score. Another way of pre-training strategy is learning from rankings. Xu *et al.* [15] train a Siamese Network to rank images in terms of image quality by using synthetically generated distortions for which relative image quality is known. And to learn type-specific ranking rules, Xu *et al.* [17] train a ranking model for each clustering of distortion type. The former only considers the rankings between samples while ignoring another important distortion-related factor, *i.e.*, distortion type. The latter performs rank learning for each type, requiring training the same number of branching networks as the distortion type, which leads to a great increase in network complexity when types get more.

Here, we design a novel framework to learn better representation for both relationships between distortion types and distortion levels (*i.e.*, learning to discriminate then rank). The main idea can be concluded as two aspects: one is modeling the distortion-related factors as graph model instead of plane model realized by classification task, the other is learning the relationship between distortion types for better generalization to unseen distortion types.

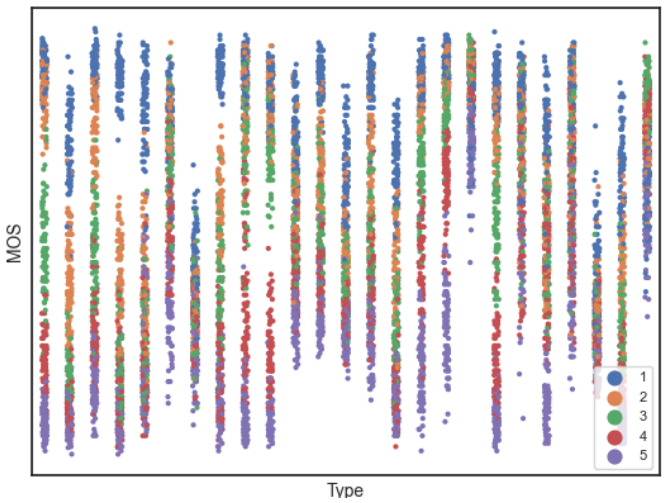
B. Graph Representation Learning

A graph can represent data that are generated from non-Euclidean domains with relationships and inter-dependency between data [16]. The challenge in graph representation learning is finding a way to properly represent/encode the graph structure so that it can be easily integrated into the machine learning model. Most of the traditional methods are based on hand-crafted features, such as statistics or kernel functions. Recently, encouraged by the success of CNNs in the computer vision field, a large number of methods that are based on automatically learned low-dimensional embeddings to encode the structure of graphs have been developed. Having the ability of neighborhood aggregation, Graph convolutional networks (GCNs) have been successfully applied to many tasks [53], [53]–[56]. Graph attention network (GAT) [57] further integrates masked self-attention mechanism in GCN. Different from the aggregation method of weighted sum in GCN, Hamiton *et al.* [58] propose GraphSAGE, which introduced an inductive learning mode. By training the model to aggregate neighbor nodes using max-pooling and LSTMs [59], GraphSAGE is extended to inductive learning task, so that it can achieve the generalization for unknown nodes. However, the mentioned methods are based on neighborhood aggregation resulting in the shallow representation of graph, which prevents the model from obtaining adequate global information. Therefore, Hu *et al.* [60] propose hierarchical graph convolutional network (H-GCN) with a graph pooling mechanism to solve the above problem, showing great improvement. In this paper, we are inspired that the effect of distortion on the perceptual image quality is not only manifested in the characteristics of distortion itself, but also the distribution of samples at different levels under that distortion. There is a hierarchical relationship between distortion type and level, which is appropriate to be modeled as a graph. Therefore, we introduce graphs to efficiently represent various distortions, which will be used to help the representation learning of distortion then improve the performance on IQA task.

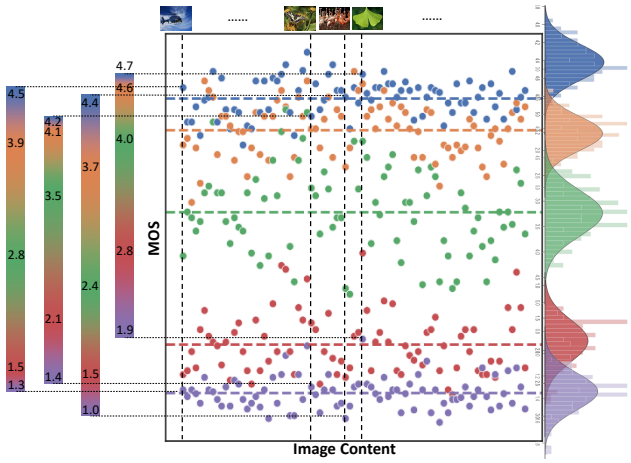
III. Methodology

A. Motivation

Subjective image quality assessment is commonly obtained by collecting mean opinion scores from many subjects, which is labor-intensive and impractical. Recently, the learning-based methods have drawn much attention as the high efficiency and accuracy. As widely accepted, the human visual system has different sensitivity to a different distortion types and levels [61], [62], and thus leveraging them to optimize the IQA task is a common approach. Most of the existing methods regard the different distortion types as a plane model, which is achieved by type or level classification task or learning to rank. They are proved to somehow bring in the improvements



(a)



(b)

Fig. 2: Statistical analysis of Kadid-10k database. (a) shows the Type-MOS (Mean Opinion Score) distribution where different colors denote the distortion level. (b) the detailed distribution of a specific distortion type. We select four image content as examples and illustrate their specific MOS under each level on the left, which shows that images with different content have different changes in MOS when level changes from 1 to 5. We also show the MOS distribution of each level in the right, where each distribution generally tends to the Gaussian distribution.

to IQA tasks but fail to model the relationship between types and levels even other distortion related factors.

To further investigate how perceptual image quality is affected by distortion-related factors, we start from the analysis of IQA datasets. To get a more generalized conclusion, our analysis is based on Kadid-10k dataset [28], which is a large-scale dataset including $10,125$ images with 25 distortion types and 5 distortion levels. As observed from the statistics of Kadid-10k that is shown in Fig. 2(a), the distortion types are crucial influential factors to the distribution of IQA scores which is consistent with our common knowledge. Meanwhile, the distributions of diverse distortions also have differences, one of which is shown in Fig. 2(b) in detail. As shown in Fig. 2(b), IQA scores present a sequential distribution according to different distortion levels, that is, the higher the level (means to be of more serious distortion) the lower the IQA scores. There is also a constant rule between samples with various levels, such as that the samples with level-2 are much more similar with samples with level-1 than them with level-5. In short, the distortion levels are the discrete points sampled on the degradation curve, in addition to their characteristics, they also satisfy the ranking relationship. Thus the level prediction problem is a regression problem. Besides, the samples with the same distortion and level tend to have similar characteristic of distortion, so that the scores tend to cluster together. This lead to the opinion that perceptual image quality is still affected by image content. According to our statistics of samples with the same type and level, under the same type and level, the vibration of scores still exists, and it obeys the Gaussian distribution.

In addition to the characteristics of each distortion-related factor, we also observe that the relationship between distortion type and distortion level tends to be a hierarchical model. That is, when we analyze the level, we tend to the hierarchical relationship where the IQA task first needs to consider type, and in each type, it needs further consider the level distribution of samples. This is more in line with the human eye's analysis habit when analyzing distorted images, and is applied in some existing methods [15], [17]. Learning the ranking relationship between samples of totally different types and different levels will not provide prior knowledge for analysis, but will make it harder for analysis. Similar to the analysis of level, the prior knowledge of image content can be easier to capture when analyzing samples with fixed type and level.

Motivated by these observations, we conclude that the relationship between type and level is hierarchical and propose the use of the graph. Then we integrate the graph representation learning method to learn the distortion graph representations (DGRs). They can simultaneously represent the characteristic of each distortion and its internal structure related to the distribution of samples with different levels, so as to model the character of distortion-related factors and their hierarchical relationship at the same time. Correspondingly, we learn DGRs from two aspects, which are type discrimination task and fussy level prediction task. The detail will be described in detail in the following sub-sections.

B. Distortion Graph Representation

We build DGR as shown in Fig. 3, whose nodes represent samples, while edges indicate the relationships between each of them. The important symbols to be used in network together with their definitions are noted of in the TABLE I. The DGR of distortion k is formulated as $G_k = (V_k, E_k)$, in which the V_k denotes the set of nodes and the E_k denotes the set of edges to describe the relationship between nodes. Specifically, the anchor batch with N samples from the same distortion type is first fed into a CNN backbone such as ResNet50 [63] to obtain the feature set $\mathcal{F}_k = \{f_i | i = 1, 2, ..., N\}$ where $f_i \in \mathbb{R}^C$ and C is the feature dimension. The extracted feature f_i from each

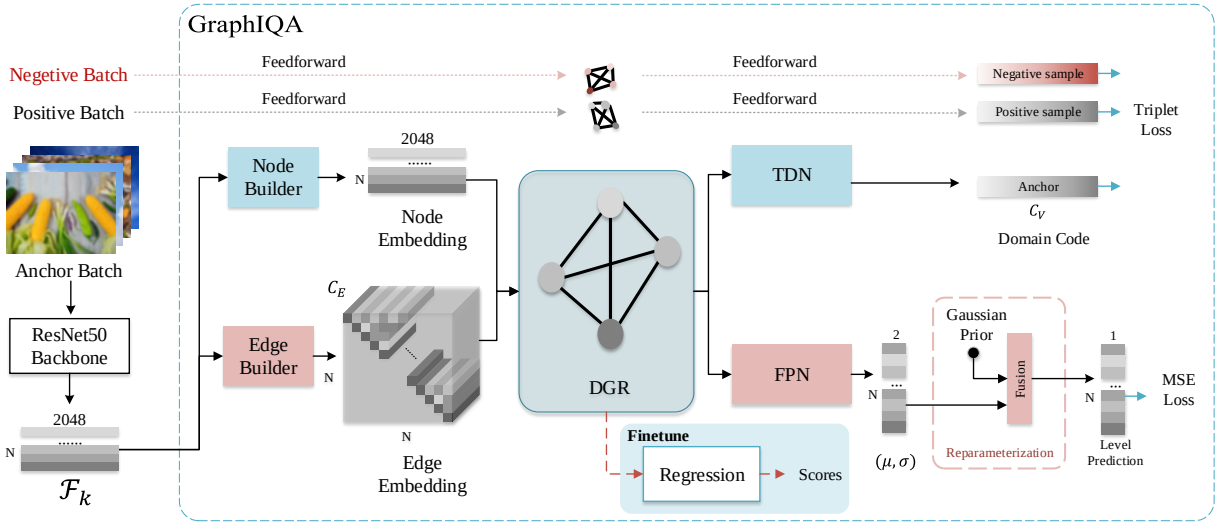


Fig. 3: The illustration of the proposed GraphIQA. We first train the networks to learn Distortion Graph Representations (DGRs). Each DGR is sample from one specific distortion type. For each iteration, three DGRs (Anchor batch, Positive batch, and Negative batch) are samples to train a model to capture the relationship between distortion types with triplet loss. The learned DGRs are utilized to improve IQA performance by finetuning the regression network on a target IQA dataset. Note that GraphIQA doesn't require MOS or DMOS supervision in the first stage.

TABLE I: The notations of important symbols.

Symbol	Description
G	Distortion representation graph (DGR)
$\mathcal{F}$	The set of features obtained from backbone
f	The specific feature obtained from backbone
V	The set of Node embeddings in a graph
v	The specific Node embedding
E	The union of Edge embeddings in a graph
e	The specific Edge embedding
A	Adjacency matrix
W_{EB}, W_{TDN}	Weights of GCN
y_{code}	The domain code
y	The prediction of level

sample is used as the initialization of the node in V_k, while the similarity between each node serves as the initialization of the edge in $\mathcal{E}_k$ which is commonly expressed as a 2D adjacency matrix $A_k \in \mathbb{R}^{N\times N}$. However, considering the complexity of the relationship between samples, we expand the 2D adjacency matrix to a 3D adjacency matrix $A_k \in \mathbb{R}^{N\times N\times C_E}$ where the representation of each edge is a vector with dimension size C_E instead of a scalar. To build the DGRs, we design two learnable modules: Node Builder and Edge Builder respectively.

a) Node Builder (NB): In DGRs, it is desirable that the representation of each node embedding should contain more distortion-related information so that it can be further used as a "clue" to distinguish from different distortion types. Therefore, we use a learnable network NB, composed of fully connected layers, to optimize node embedding, which is formulated as

$$V_k = \{v_{k,i} | v_{k,i} = F_{NB}(f_{k,i};\theta), v_{k,i} \in \mathbb{R}^C, i = 1,2,...,N\}, \tag{1}$$

where $v_{k,i}$ denotes the node embedding of i-th sample, and θ denotes the network parameters of NB.

b) Edge Builder (EB): To obtain rich information about the contrast relationship between nodes, we expand the adjacency matrix to 3D, *i.e.*, we represent the connection between two nodes by a vector. We take the edge vectors $e^0_{k,i,j} \in \mathbb{R}^C$ as initial edge embedding E^0_k, which is the result of dot multiplication between each node embedding. The edge embedding is further optimized to represent internal structure by a graph convolution network (GCN) [64]. In detail, given the edge embedding E^0_k and $A_{E_k} \in \mathbb{R}^{N^2\times N^2}$ as input, the process of computation of each layer for the GCN with L layers can be formulated as:

$$E^{l+1}_k = \text{ReLU}(\hat{A}_{E_k} E^l_k W^l_{EB}), \tag{2}$$

where

$$D = \text{diag}(\sum\nolimits_{p=1}^{N^2} (A_{E_k} + I)_p), \tag{3}$$

$$\hat{A}_{E_k} = D A_{E_k} D. \tag{4}$$

The initialization of edge embedding E^0_k serves as the input of the first layer of edge builder, and E^l_k denotes the output of the GCN l-th layer. W_l is the trainable parameter of GCN l-th layer. In the end, the optimized edge embedding of DGR is defined as:

$$E_k = \{e_{k,i,j} | e_{k,i,j} \in \mathbb{R}^{C_E}, i,j = 1,2,...,N\}, \tag{5}$$

where the C_E is the dimension of edge embedding, which is set much smaller than C to reduce computational complexity.

C. Domain Graph Optimization

As shown in Fig. 3, to equip the DGRs with the ability to both representing each distortion and the relationship between

distortion levels, GraphIQA learns DGRs from the following two aspects. a) To learn the representation of distortion types that can be distinguished from other type, and the contrast relationship between them for better generalization, we design the TDN; b) To learn the distribution of distortion levels based on considering the content impact, we design the FPN.

a) Type Discrimination Network (TDN): TDN is used to obtain the typical compact representation of each DGR, which helps to distinguish it from the others. Specifically, we design a GCN to aggregate global information from node embedding and relationships from edge embedding. The process is formulated as follow:

$$V_k^{l+1} = \text{ReLU}(\hat{A}_{V_k} V_k^l W_{TDN}^l), \tag{6}$$

in which the V_k is the node embedding and the $A_{V_k} \in \mathbb{R}^{N\times N}$ is the adjacency matrix of nodes, which is calculated by transforming edge embedding E_k through the average pooling across channels.The process of transforming edge embedding is noticed as Node Pooling. $\hat{A}_{V_k}$ is defined similar to Equation (4). The output of the TDN will be a vector with dimension C_V, named as code y_{code}. Then triplet loss $\mathcal{L}_{dist}$ [19] is utilized to learn the contrast representation of different distortion types. It is achieved by aggregating the anchor DGR and the DGR of the same distortion while separating it from the DGR of the other distortion. In detail, the forward propagation will be conducted three times to get triplet with three different input batches. The three sub-graphs are obtained by "Anchor Batch", "Positive Batch " and "Negative Batch" as it is shown in Fig. 3. To simplify the diagram, we use dotted lines to represent the repeated forward propagation process. The loss function is formulated as:

$$\begin{aligned}\mathcal{L}_{dist} = max(d(y_{code}^{Anchor}, y_{code}^{+}) \\ - d(y_{code}^{Anchor}, y_{code}^{-}) + margin), 0),\end{aligned} \tag{7}$$

where d denotes L2 distance, and y_{code}^{+} is the DGR representing the same type with anchor DGR (positive sample in Fig. 3) while y_{code}^{-} is the DGR representing the different types (negative sample in Fig. 3). Here, triplet loss can not only learn more subtle difference between distortion types, but also learn a contrast relationship between distortion types, which avoids the network overfitting to the distortion types in the training set.

b) Fuzzy Predictor Network (FPN): We design an FPN to predict levels while considering the uncertainty caused by image content. Even if samples share the same distortion type and level, their perceptual image quality still is different due to their different content. According to Fig. 2(b), we assume that the scores of samples at the same level are distributed near the average score, obey the Gaussian distribution. Then we perform the prediction by randomly sampling from the Gaussian prior distribution $\mathcal{N}(\mu, \sigma^2)$. As this process is not differentiable, the reparametrization trick [65] is used to ensure end-to-end training of the network. In detail, the ϵ is sampled from Normal distribution $\mathcal{N}(0, 1)$, and then mapped to an arbitrary Gaussian distribution according to the generated hyper-parameters:

$$y_i = \mu_i + \sigma_i \epsilon, \epsilon \in \mathcal{N}(0, 1), \tag{8}$$

where μ and σ are the predicted mean and scale generated by the hyper predictor, as is shown in Fig. 3. Because the prediction of distortion level is not only achieved by analyzing the distortion-related representation of nodes, but also the comparison between nodes to estimate the level of distortion, both the node embedding and edge embedding are needed. Specifically, node embedding V_k is fed into FPN directly, while edge embedding E_k is averaged across the rows, which is noticed as Edge Pooling, to aggregate the information from the current node and all the other neighboring nodes, as $E_k' = [\sum_j e_{k,i,j}]/N$. Then mean square error (MSE) loss function is utilized when training the hyper predictor:

$$\mathcal{L}_{level} = \sum\nolimits_i |y_i - y_i'|^2, \tag{9}$$

where y_i' denotes the target level. The entire model will be trained end-to-end to minimize the combination of above loss functions, which are weighted by hyper-parameter λ:

$$\mathcal{L} = \mathcal{L}_{dist} + \lambda \mathcal{L}_{level}. \tag{10}$$

The detail of the architecture is shown in Fig. 4.

D. Finetune and Inference

Benefiting from the improved representation ability of DGRs, GraphIQA shows the potential for better fulfillment of IQA tasks. Specifically, when finetuning on the target dataset, both the node embedding and edge embedding are used to regress the IQA scores. And there is no need for representation building modules (*i.e.*, TDN and FPN). Specifically, the node embedding and edge embedding is concatenated as a vector, and fed into the regression module. Here the edge embedding is self-loop edges, which contains rich distortion level priors learned from the pre-training phase. The regression module is a small and simple network with two fully connected layers. When using DGRs for IQA on authentically distorted datasets, the prediction of authentic distortion datasets is achieved based on the Gaussian prior distribution so that it can better handle the unknown distortion type. Then the entire model is finetuned to minimize the MSE between ground truth (which is MOS/DMOS) and predicted scores, which is defined as:

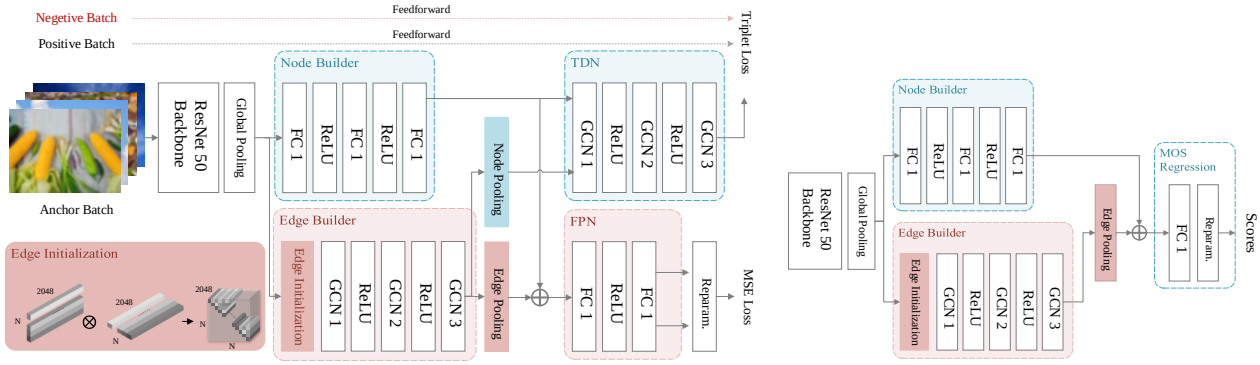
$$\mathcal{L}_{scores} = \frac{1}{N_f} \sum_{i=1}^{N_f} |c_i - c_i'|^2, \tag{11}$$

where the N_f denotes the mini-batch size. It is worth noting that as it already has the ability to infer the DGR, GraphIQA can support any size of input batch in the inference stage.

IV. Experiments

A. Experiments Setting

a) Dataset: During pre-training, we use Kadid-10k or Kadis-700k [28]. The former is a large synthetic distorted



(a) GraphIQA framework (b) Finetune GraphIQA on target database

Fig. 4: The illustration of the architecture of GraphIQA, in which (a) is the architecture for pre-training while (b) is the architecture for finetuning.

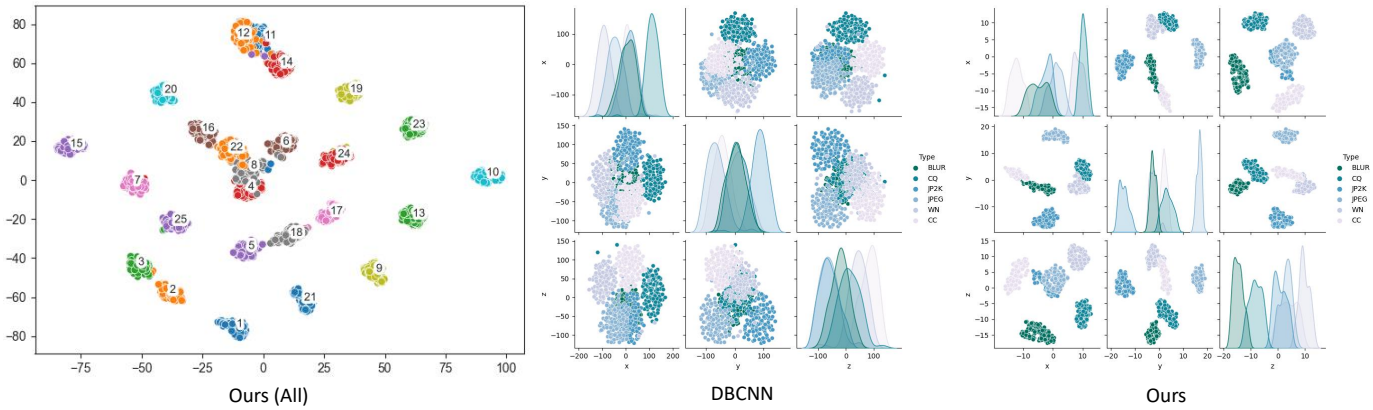


Fig. 5: Visualization results of the representations which are generated by our method. To obtain the comparison results with DBCNN [4], the visualization results of the representations of the common distortion types are shown in the right sub-figure. It shows that the DGRs in our method are more discriminable than the representation learned by the classification task in DBCNN.

TABLE II: Evaluation on clustering performance of samples in each DGR on Kadid-10K dataset, which intuitively shows the performance of the DGR on level representation. The performance is measured by three metrics: homogeneity (means all of the observations with the same class label are in the same cluster, 1 means the best), completeness (means all members of the same class are in the same cluster, 1 means the best), and V-measure (the combination of both homogeneity and completeness, 1 means the best).

Type	Homo.	Comp.	V-m.	Type	Homo.	Comp.	V-m.
GB	0.815	0.735	0.773	MN	0.689	0.639	0.663
LB	0.850	0.772	0.809	Denoise	0.803	0.797	0.800
MB	0.664	0.765	0.711	Brighten	0.573	0.511	0.524
CD	0.424	0.449	0.437	Darken	0.401	0.408	0.404
CS	0.271	0.350	0.306	MS	0.213	0.265	0.236
CQ	0.557	0.583	0.570	Jitter	0.706	0.635	0.669
CSA1	0.554	0.571	0.562	NEP	0.193	0.216	0.204
CSA2	0.353	0.373	0.362	Pixelate	0.778	0.709	0.742
JP2K	0.549	0.614	0.580	Quan.	0.373	0.385	0.379
JPEG	0.717	0.659	0.687	CB	0.177	0.329	0.230
WN	0.780	0.706	0.741	HS	0.519	0.534	0.526
WNCC	0.662	0.794	0.772	CC	0.300	0.427	0.452
IN	0.772	0.712	0.741				

database containing 81 images with 25 distortion types[1] and 5 distortion levels. The latter is a large-scale synthetic distortion dataset with 700,000 distorted images with 25 distortion types and 5 distortion levels for each type. We use Kadid-10k dataset as validation set to select the hyper-parameters for pre-trained model.

For target datasets, we choose two datasets with authentic distortion (KonIQ-10k [23] and LIVE Challenge (LIVEC) [24]), two datasets with synthetic distortion (LIVE [20] and CSIQ [21]) and a dataset with multiple distortions (LIVEMD [22]). KonIQ-10k consists of 10073 images which are selected from the large public multimedia database YFCC100m [66]. Those samples try to cover a wide and uniform quality distortion. LIVEC contains 1162 images taken from different photographers with various cameras. LIVE contains 779 images with 5 distortion types and CSIQ

[1]GB: Gaussian blur; LB: Lens blur; MB: Motion blur; CD: Color diffusion; CS: Color shift; CQ: Color quantization; CSA1: Color saturation 1; CSA2: Color saturation 2; WN: White noise; WNCC: White noise in color component; IN: Impulse noise; MN: Multiplicative noise; MS: Mean shift; NEP: Non-eccentricity patch; Quan.: Quantization; CB: Color block; HS: High sharpen; CC: Contrast change

contains 866 images with 6 distortion types. LIVEMD contains 450 distorted images with 2 multiple distortion types (i.e., blur-jpeg and blur-noise). When finetuning, we split the dataset into a training set and a test set according to [2], [4]. Specifically, for synthetic distorted datasets, to avoid content overlapping between the training set and test set, we first randomly split the source images according to the ratio of $8:2$, and then assign the corresponding distorted images to obtain the training set and test set. For authentic distorted datasets, there is no image content overlapping, we directly split the whole dataset into a training set and a test set according to the ratio of $8:2$. All the results are obtained by training and test on the specific target dataset 10 times with a randomly splitting operation, and the average results are reported.

b) Evaluation Metrics: We mainly adopt two commonly used metrics, which are Spearman's rank order correlation coefficient (SRCC) and Pearson's linear correlation coefficient (PLCC) to measure the prediction monotonicity and prediction accuracy. Both of them range from -1 to 1 and a higher value indicates better performance.

c) Implementation Details: We implement our model by PyTorch, and both training and testing are conducted on the NVIDIA 2080Ti GPUs. For data augmentation, when pre-training the GraphIQA model, we randomly sample from each distortion type and randomly crop them into 224×224 patches 25 times, as there tends to be local distortion in the training database. The hyper-parameter λ for loss function is set as 0.25. The margin of the triplet loss function is set to 0.1. We use Adam [67] optimizer to pre-train our representation model for 350000 steps with mini-batch size of 32. Learning rate is set to 1×10^{-5}. The dimension size of node embedding C_V is set to 256, and the size of edge embedding C_E is set to 64 (The detailed experiments on hyper-parameters are shown in Section IV-E). During finetuning, the input samples are randomly cropped into 224×224 at 10 times (some large images are resized to proper size firstly and randomly cropped to 224×224). We use Adam [67] optimizer to finetune on IQA task for 20 epochs with the mini-batch size of 32. The learning rate for finetuning is set to 5×10^{-6}. During the testing stage, all the testing images are randomly cropped to 10 224×224 patches, and their corresponding prediction scores are averaged to get the final quality scores.

B. DGR Performance Evaluation

We evaluate the effectiveness of the proposed DGR from three aspects.

a) Visualization and Clustering Evaluation: We visualize distribution of learned DGRs using *t-SNE* [69], as are shown in Fig. 5 and 6 respectively, which is then compared with representations generated by DBCNN [4]. We randomly sample 128 times from six typical distortion types in Kadid-10k dataset and feed into Kadis-700k pre-trained model (without image content overlap) to get the dimension-reduced embedding. To reduce the bias of the visualization caused by *t-SNE* during dimension reducing, we visualize the distribution of three-dimension space. We can see that the DGRs are well-clustering representations according to their corresponding

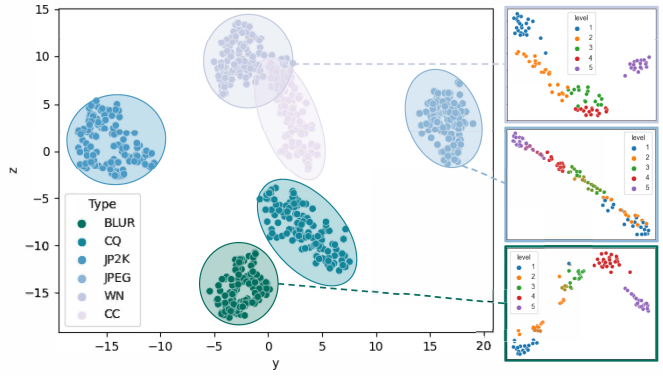


Fig. 6: Visualization results of DGRs and their intrinsic structure. The DGRs themselves are highly discriminable. Besides, their internal distribution is in accordance with the ranking relationship of distortion levels, indicating that the nodes of the DGR also have the ability to represent levels.

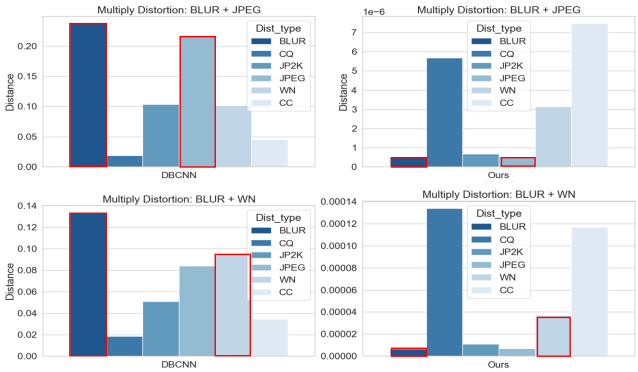


Fig. 7: Visualization of interpretability for unseen distortion. This figure shows the distance between representations of the unseen multiple distortion type, *i.e.*, blur and jpeg (noted as BLUR+JPEG), and blur and noise (BLUR+WN). It proves that our method has the potential of interpretability that the representations of BLUR+JPEG are closest to them of BLUR and JPEG. and the representations of BLUR+WN are closest to them of BLUR.

distortion types on the whole, but representations generated by DBCNN are less discriminable. We also visualize the internal distribution of each DGR, as shown in Fig. 6. From them, we can observe that the node embeddings are not only clustering well according to the distortion level, but also show a regular pattern according to the order of levels. TABLE II provides the clustering performance for all 25 distortion types in the Kadid-10k dataset, which is measured by homogeneity, completeness and V-measure. All of the metrics used for the measurement of clustering performance is range from 0 to 1, and the 0 means poor clustering performance while the 1 means the best. Considering that the prediction of levels is sampled from Gaussian prior distribution to model the influence by image content, the high accuracy is not our main concern. In TABLE II, most of the results are higher than 60%, which further proves the powerful representation capability of each DGR to characterize each level.

TABLE III: SRCC comparison in cross distortion type on Kadid-10k dataset. We test the IQA performance for one distortion type at a time, and performance for all the distortion types are shown in table. All the best results are highlighted in bold.

Dist. type	GB	LB	MB	CD	CS	CQ	CSA1	CSA2	JP2K	JPEG	WN	WNCC
WaDIQaM [68]	0.879	0.730	0.730	0.833	0.421	0.806	0.148	0.836	0.539	0.530	0.897	0.925
MetaIQA [3]	0.946	0.917	0.926	0.892	**0.785**	0.717	0.304	**0.931**	**0.945**	0.912	0.905	0.930
Ours (w/o)*	0.925	0.875	0.915	0.811	0.725	0.642	**0.501**	0.618	0.941	0.822	0.817	0.875
Ours	**0.958**	**0.938**	**0.951**	**0.926**	0.738	**0.873**	0.462	0.929	0.938	**0.944**	**0.916**	**0.955**

Dist. type	IN	MN	Denoise	Brighten	Darken	MS	Jitter	NEP	Pixelate	Quan.	CB	HS	CC
WaDIQaM [68]	0.814	0.884	0.765	0.685	0.272	0.348	0.778	0.348	0.700	0.735	0.160	0.558	0.421
MetaIQA [3]	**0.867**	0.925	0.899	0.783	0.622	0.556	0.928	0.418	0.809	**0.877**	0.513	0.437	0.438
Ours (w/o)*	0.811	0.911	0.858	0.412	0.707	0.071	**0.949**	0.541	0.800	0.639	0.334	0.749	0.049
Ours	0.845	**0.951**	**0.922**	**0.889**	**0.806**	**0.745**	0.943	**0.677**	**0.873**	0.867	**0.626**	**0.904**	**0.825**

* denotes the performance of the proposed GraphIQA without being finetuned on target datasets.

b) Interpretability for unseen distortions: To verify that the DGRs have better generalization capability, we compare the interpretability for unseen distortions of our models and DBCNN. We test on multiple distortion dataset (LIVEMD). There are two multiple distortion types which are BLUR and JPEG, and BLUR and white noise. We first test the interpretability only on the former multiple distortion type. The distance between embedding of samples of multiple distortions and every single distortion is calculated, which is shown in Fig. 7. With our method, the distance between BLUR+JPEG and BLUR is the closest, and JPEG is the second closest. However, the representation generated by DBCNN is not interpretable. For the distortion BLUR+WN, though considering that features of the latter are highly unstable with different mixing methods (e.g., mixing order), it also shows the potential of interpretability. That is, when compared to the distance between the BLUR+JPEG distortion type and Noise, the distance between the BLUR+WN distortion type and Noise is much closer. It shows that benefit from the learning of contrastive relationships between distortion types, our method is able to learn a more robust representation that even can somehow handle the unseen distortion types.

c) Leave-One Evaluation on Kadid-10k Dataset: To further validate the contribution of DGRs to IQA task, we test the IQA performance for each distortion type. We compare our method with two CNN based BIQA methods by using the Leave-One-Distortion-Out cross validation. In detail, there is one distortion type left for testing and the rest of types are used as a training set. All of the results are obtained by using the source code provided by their authors in the same training-testing conditions. With all the best results highlighted in bold, we can see from the TABLE III that GraphIQA scheme can already achieve competing performance without finetuning on the target dataset. After finetuning on the training set, the performance can get even better that we reach the best on most of the distortion types (18 out of 25).

C. Comparison with the State-of-the-arts

We compare our GraphIQA with the state-of-the-art (SOTA) BIQA methods including hand-craft feature based methods [31], [32], [37], deep learning based synthetic IQA methods [7], [15], [18], [68], [70], [74] and deep learning based authentic IQA methods [2]–[4], [51], [71]–[73]. All of the experiments are conducted 10 times to avoid the bias of randomness.

TABLE IV: Comparison with SOTA methods on multiple datasets with SRCC and PLCC metrics. Among the compared methods we focus on comparing with the classical ones of using representational learning to improve IQA performance. These methods are listed at the bottom of the tables.

	SRCC				
	KonIQ	LIVEC	LIVE	CSIQ	LIVEMD
BRISQUE [31]	0.665	0.608	0.939	0.746	0.886
ILNIQE [32]	0.507	0.432	0.902	0.806	0.876
HOSA [37]	0.671	0.640	0.946	0.741	0.913
BIECON [70]	0.618	0.595	0.961	0.815	0.909
WaDIQaM [68]	0.797	0.671	0.954	**0.955**	-
SFA [71]	0.856	0.812	0.883	0.796	-
PQR [51]	0.880	**0.857**	0.965	0.873	-
UNIQUE [72]	0.896	0.854	0.969	0.902	-
HyperIQA [2]	0.905	**0.856**	0.962	0.920	-
MetaIQA [3]	0.850	0.802	-	-	-
MetaIQA+ [73]	**0.909**	0.852	-	-	-
CaHDC [74]	-	-	0.965	0.903	**0.927**
CNNIQA++ [18]	-	-	0.965	0.892	**0.927**
RankIQA [15]	-	0.641	**0.981**	0.892	0.908
MEON [7]	-	-	0.951	0.852	0.924
DBCNN [4]	0.872	0.852	0.967	0.946	**0.927**
Ours	**0.911**	0.845	**0.979**	**0.947**	**0.930**
	PLCC				
	KonIQ	LIVEC	LIVE	CSIQ	LIVEMD
BRISQUE [31]	0.681	0.629	0.935	0.829	0.917
ILNIQE [32]	0.523	0.508	0.865	0.808	0.863
HOSA [37]	0.694	0.678	0.947	0.823	0.926
BIECON [70]	0.651	0.613	0.962	0.823	0.933
WaDIQaM [68]	0.805	0.680	0.963	**0.973**	-
SFA [71]	0.872	0.833	0.895	0.818	-
PQR [51]	0.884	**0.882**	0.971	0.901	-
UNIQUE [72]	0.876	0.890	0.968	0.927	-
HyperIQA [2]	**0.922**	**0.882**	0.966	0.943	-
MetaIQA [3]	0.887	0.835	-	-	-
MetaIQA+ [73]	**0.922**	0.852	-	-	-
CaHDC [74]	-	-	0.964	0.914	**0.950**
CNNIQA++ [18]	-	-	0.966	0.905	0.924
RankIQA [15]	-	0.675	**0.982**	0.912	0.929
MEON [7]	-	-	0.955	0.864	**0.940**
DBCNN [4]	0.881	**0.865**	0.971	**0.959**	0.934
Ours	**0.915**	0.862	**0.980**	**0.959**	**0.940**

a) Single Database Evaluations: The results are shown in TABLE IV, and the best results are highlighted in bold. Our approach outperforms all of the methods on both synthetic distortion and multiple distortion datasets (LIVE, CSIQ and LIVEMD). And for authentic distortion datasets (KonIQ-10k

TABLE V: SRCC comparison on individual type in LIVE and CSIQ dataset.

Dataset	LIVE						CSIQ						
Type	JP2K	JPEG	WN	GB	FF	Total	JP2K	JPEG	WN	GB	CC	FN	Total
BRISQUE [31]	0.929	0.965	**0.982**	0.964	0.828	0.939	0.840	0.806	0.723	0.820	0.804	0.378	0.746
ILNIQE [32]	0.894	0.941	0.981	0.915	0.833	0.902	0.906	0.899	0.850	0.858	0.501	0.874	0.806
HOSA [37]	0.935	0.954	0.975	0.954	0.954	0.946	0.818	0.733	0.604	0.841	0.716	0.500	0.741
BIECON [70]	0.952	0.974	0.980	0.956	0.923	0.961	0.954	0.942	0.902	0.946	0.523	0.884	0.815
WaDIQaM [68]	0.942	0.953	0.982	0.938	0.923	0.954	0.947	0.853	**0.974**	**0.979**	0.923	0.882	**0.955**
HyperIQA [2]	0.949	0.961	0.982	0.926	0.936	0.962	**0.960**	0.934	0.927	0.915	0.874	0.931	0.920
DBCNN [4]	0.955	0.972	0.980	0.935	0.930	0.967	0.953	0.940	0.948	0.947	0.870	0.940	0.946
Ours (w/o)*	0.901	0.825	0.385	0.791	0.908	0.711	0.859	0.912	0.883	0.800	0.029	0.854	0.705
Ours	**0.979**	**0.978**	0.978	**0.978**	**0.979**	**0.979**	0.947	**0.947**	0.948	0.947	**0.947**	**0.948**	0.947

* denotes the performance of the proposed GraphIQA without being finetuned on target datasets.

and LIVEC), Our method can obtain comparable performance with methods that are well designed for authentic distorted data [2]. Notably, in our method, none of the authentic distorted data and multiply distorted data is used in the pre-training stage. This shows that our pre-trained model shows better generalization ability to be transferred to other distortion domains. Observed from the results with pre-trained models which are trained on Kadid-10k dataset and Kadis700k dataset, with a large-scale dataset the pre-trained model with proposed method can get better performance, especially on synthetic distorted data. This suggests that the learned DGRs can be utilized to deal with both synthetically and authentically distorted images.

We also present the performance comparison of our approach on individual distortion types. We choose LIVE and CSIQ which are unseen in our pre-training stage for a fair comparison. The results are shown in TABLE V. Compared with some methods, our approach, which is pre-trained without using the annotation of MOS/DMOS, noticed as Ours (w/o) in the table, can still get comparable performance on some distortion types. After finetuning on the target dataset MOS/DMOS annotations, the performance gets better and more consistent over individual distortion types. This shows that the DGRs do provide rich effective prior for IQA, while having better generalization.

b) Generalization Evaluation: At first, we run the cross dataset tests on both synthetically distorted dataset pairs (LIVE and CSIQ) and authentically distorted dataset pairs (KoinIQ and LIVE). We select the most competing methods, DBCNN [75] and HyperIQA [2] for comparison. In the implementation, we use one dataset as a training set and the other one is used as a testing set. The results are shown in TABLE VI, it can be observed that our approach can get comparable performance with the other methods. This is because of the strong generalization ability of our approach.

Then, to explore the generalization of GraphIQA on the quality assessment of enhanced images, we compare the performance on the de-hazed quality assessment dataset [76] and the de-raining quality assessment dataset [77], which is shown in TABLE VII. As is shown, our method can achieve better performance on de-hazed dataset and comparable performance on de-raining dataset when compared with Res50 (the same backbone), though the enhanced image is un-known for pre-training and there is no design targeting the distortion properties on enhanced images in our method. This verifies the generalization of GraphIQA.

TABLE VI: Cross-dataset evaluation to verify the generalization.

Dataset		DBCNN	HyperIQA	Ours
Train	Test			
KonIQ	LIVEC	0.734	0.773	**0.798**
LIVEC	KonIQ	0.**788**	0.733	0.773
CSIQ	LIVE	0.909	0.940	**0.945**
LIVE	CSIQ	0.775	**0.834**	0.830

TABLE VII: Evaluation on enhanced quality assessment dataset to verify the generalization

	De-Hazed Dataset		De-Raining Dataset	
	SRCC	PLCC	SRCC	PLCC
VGG19	0.8258	0.8310	0.4145	0.4139
Res50	0.8361	0.8422	0.4299	0.4231
Ours (Res50)	0.8386	0.8497	0.4237	0.4179

D. Ablation Study

To evaluate the efficiency of each component in our approach. We conduct ablation study on both synthetic and authentic distortion datasets, *i.e.*, KonIQ and LIVE. And when selecting the size of node embedding and edge embedding, we conduct the linear evaluation experiments on validation set, Kadid-10k dataset, as [12], where the parameters of DGRs generation part (Backbone, Node Builder and Edge Builder) are fixed and the linear convolutional layers are trained to regress the MOS. The SRCC results are reported.

TABLE VIII: SRCC evaluation of ablation study on KonIQ and LIVE dataset. Res50 denotes the ResNet model. In pre-training setting, the marked part means it is pre-trained with distortion type and level on Kadid10k dataset. In finetuning setting, the marked part means it is used to predict final scores.

Setting								Dataset	
Pre-train				Finetune					
Res50	DGRs	FPN	cls	Res50	DGRs	Nodes	Edges	KonIQ	LIVE
				✓				0.904	0.964
✓	✓	✓		✓				0.899	0.956
✓	✓	✓		✓	✓	✓		0.898	0.970
✓	✓	✓		✓	✓		✓	0.298	0.855
✓	✓		✓	✓	✓	✓	✓	0.899	0.969
✓	✓	✓		✓	✓	✓	✓	**0.911**	**0.979**

TABLE IX: The linear evaluation of the dimension size of the edge and the node on Kadid-10k dataset.

Edge	1	16	32	64	96
SRCC	0.8246	**0.8316**	0.7015	0.8270	0.8137
Node	32	64	128	256	300
SRCC	0.8227	0.8034	0.7932	**0.8270**	0.8079

a) Model Components: As the scheme of using backbone ResNet50 is treated as the baseline, all the components are integrated into it, as shown in TABLE VIII. We first verify the efficiency of each component in the pre-training stage. It is observed that ImageNet pre-trained model can get better performance on the authentically distorted images, but inferior performance on synthetically distorted images. However, training the distortion representation without the model of a graph may degrade the performance. When training the representation for level, as the prediction of level is a regression problem instead of a classification task, the performance of using FPN is better than using the softmax classifier. Than we verify the effectiveness of DGR and its components. The performance of utilizing pre-trained backbone together with edge embedding and node embedding separately is not as good as using them in combination, especially for edge embedding. This is because the most of information on edges has been aggregated into nodes during the training stage. When combining both of them, the GraphIQA can get better performance compared with baseline. Then we further provide the performance on ImageNet pre-trained backbone. As is shown in the table, the ImageNet pre-trained backbone performs better on authentically distorted data while worse on synthetically distorted data which is consistent with the comments in [4]. However, with our method, the pre-trained model can obtain better improvement on synthetic distortion dataset while maintaining the performance on authentic distortion dataset. What is worth noting is that only the single pre-trained model trained with synthetic data is used in our method. It is shown that with the proposed method, a better representation of synthetically distorted data can be learned which is par for the course. Meanwhile, as the representations are obtained by learning the relative relationship between synthetic distortion types, the representation is more robust to various distortion types than that obtained by classification task, so that it can be transferred to authentically distorted data much easier.

b) Edge Embedding Size and Node Embedding Size: Then, we compare the performance on different sizes of edge embedding in TABLE IX, which are 1, 16, 32, 64 and 96. We can see that DGRs with edge embedding set as 16 can get better performance, which demonstrates the effectiveness of the proposed 3D adjacency matrix. As for node embedding size, we compared the performance on size 32, 64, 128, 256 and 300. And the performance is the best when it is set as 256. It is noted that when one hyper-parameter is under selection, the others are set as the initial hyper-parameter set, where node embedding size is set as 256, edge embedding size is set as 64 and margin size is set as 0.2.

TABLE X: Exploring the best architecture of our model.

Node Builder				
FC layer	1	2	3	4
SRCC	0.8021	0.8052	**0.8270**	0.7988
Edge Builder				
GCN layer	1	2	3	4
SRCC	0.7409	0.7956	**0.8270**	0.8112
TDN				
GCN layer	1	2	3	4
SRCC	0.8009	0.8014	**0.8270**	-

E. Experiments on architecture and hyper-parameters.

For each setting of hyperparameters and architecture, we conduct pre-training process until the loss function converges. We conduct the linear evaluation experiments on validation set, Kadid-10k dataset, as [12], where the parameters of DGRs generation part (Backbone, Node Builder and Edge Builder) are fixed and the linear convolutional layers are trained to regress the MOS. Then the SRCC results are reported.

a) Architecture: In this section, we provide experiments of network architecture on KonIQ dataset [23], which is shown in Fig. 4 and the results are shown in TABLE X. All the results are tested on models trained on $100,000$ epochs. All the other parameters are kept consistent, and only the parameters to be compared are changed. For the Node Builder, we test the performance of 1, 2, 3, and 4 fully connected (fc) layers. It is observed that when Node Builder with 3 fully connected layers achieves the best performance. We also test the performance on different number of GCN layers of Edge Builder and TDN. Edge Builder with 3 graph convolutional layers and TDN with 3 graph convolutional layers, our model achieves the best performance (when the number of graph convolutional layers in TDN is 4, the pre-training is unstable).

We also test the relationship between the number of epochs and performance on LIVEC dataset and CSIQ dataset. In the pre-training process, the model's ability to distinguish and represent each distortion is improved, leading to performance improvements on both the synthetic distortion dataset and the authentic distortion dataset. However, long-term training cannot continue to improve the performance, because overfitting to synthetic distortion dataset leads to poor generalization on unknown distortion types.

b) Complexity Analysis: In this section, we analyze the complexity of each module in GraphIQA. We list the parameter amount (noted as Param.) of each module in GraphIQA in TABLE XI. The total amount is 34.9M. It is notable that TDN and FPN do not participate in the finetuning and inference stage, which makes the actual amount is 29.9M.

TABLE XI: The number of parameters of modules.

	Backbone	NB	EB	TDN	FPN
Param.	23.5M	3.7M	2.7M	2.8M	2.2M

c) Margin of triplet loss: The margin setting in the loss function will directly affect how well the network can discriminate the distortion types. The small the margin, the greater the discriminability of the learned DGR. When the

TABLE XII: Linear evaluation results with different margins of triplet loss.

Margin	0	0.2	0.5	1	soft margin [78]
SRCC	0.7939	**0.8270**	0.8130	0.7464	0.7957

distance between the two graphs is smaller than the margin the loss is set as 0. When soft-margin is used, there is no truncation in the loss function, and the distance between similar samples can be as small as possible. Experiments on different margins of triplet loss are provided in TABLE XII. It is observed that when it is set as 0.2, the GraphIQA gets the best performance on the linear evaluation result.

V. Conclusion

In this paper, we integrate graph representation learning into IQA and propose a novel framework GraphIQA to learn DGRs. Having the ability to represent the characteristics of each distortion and the internal structure, GraphIQA can not only generate DGRs as prior knowledge when processing known distortions but also infer the influence of unknown distortions on the perceptual image quality. For future work, for better distortion representation of distortion, a more complex graph structure can be considered to optimize the existing model, e.g., the integration of hyper-nodes. To explore richer application scenarios, based on our learned DGR, we will challenge interpretable IQA problems. Besides, noting that DGR can well represent distortions, we will also try to utilize GraphIQA to participate in helping image restoration tasks, such as denoising or deblurring.

References

[1] Z. Chen, J. Xu, C. Lin, and W. Zhou, "Stereoscopic omnidirectional image quality assessment based on predictive coding theory," *IEEE Journal of Selected Topics in Signal Processing*, no. 1, pp. 103–117, 2020.

[2] S. Su, Q. Yan, Y. Zhu, C. Zhang, X. Ge, J. Sun, and Y. Zhang, "Blindly assess image quality in the wild guided by a self-adaptive hyper network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 3667–3676.

[3] H. Zhu, L. Li, J. Wu, W. Dong, and G. Shi, "Metaiqa: Deep meta-learning for no-reference image quality assessment," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 14 143–14 152.

[4] W. Zhang, K. Ma, J. Yan, D. Deng, and Z. Wang, "Blind image quality assessment using a deep bilinear convolutional neural network," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 1, pp. 36–47, 2020.

[5] W. Zhou, Z. Chen, and W. Li, "Dual-stream interactive networks for no-reference stereoscopic image quality assessment," *IEEE Transactions on Image Processing*, vol. 28, no. 8, pp. 3946–3958, 2019.

[6] L. Kang, P. Ye, Y. Li, and D. Doermann, "Convolutional neural networks for no-reference image quality assessment," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1733–1740.

[7] K. Ma, W. Liu, K. Zhang, Z. Duanmu, Z. Wang, and W. Zuo, "End-to-end blind image quality assessment using deep neural networks," vol. 27, no. 3, pp. 1202–1213, 2018.

[8] S. A. Golestaneh and K. Kitani, "No-reference image quality assessment via feature fusion and multi-task learning," *arXiv preprint arXiv:2006.03783*, 2020.

[9] Z. Chen, W. Zhou, and W. Li, "Blind stereoscopic video quality assessment: From depth perception to overall experience," *IEEE Transactions on Image Processing*, vol. 27, no. 2, pp. 721–734, 2017.

[10] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[11] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," in *4th International Conference on Learning Representations, ICLR 2016*, 2016.

[12] K. He, H. Fan, Y. Wu, S. Xie, and R. Girshick, "Momentum contrast for unsupervised visual representation learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9729–9738.

[13] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin, "Emerging properties in self-supervised vision transformers," in *9th International Conference on Learning Representations, ICLR 2021*, 2021.

[14] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT*. Association for Computational Linguistics, 2019, pp. 4171–4186.

[15] X. Liu, J. Van De Weijer, and A. D. Bagdanov, "Rankiqa: Learning from rankings for no-reference image quality assessment," in *Proceedings of the IEEE International Conference on Computer Vision*, 2017, pp. 1040–1049.

[16] W. L. Hamilton, R. Ying, and J. Leskovec, "Representation learning on graphs: Methods and applications," *IEEE Data Eng. Bull.*, vol. 40, no. 3, pp. 52–74, 2017.

[17] L. Xu, J. Li, W. Lin, Y. Zhang, L. Ma, Y. Fang, and Y. Yan, "Multi-task rank learning for image quality assessment," vol. 27, no. 9, pp. 1833–1843, 2016.

[18] L. Kang, P. Ye, Y. Li, and D. Doermann, "Simultaneous estimation of image quality and distortion via multi-task convolutional neural networks," in *2015 IEEE international conference on image processing (ICIP)*. IEEE, 2015, pp. 2791–2795.

[19] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 815–823.

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

[21] E. C. Larson and D. M. Chandler, "Most apparent distortion: full-reference image quality assessment and the role of strategy," *Journal of electronic imaging*, vol. 19, no. 1, p. 011006, 2010.

[22] D. Jayaraman, A. Mittal, A. K. Moorthy, and A. C. Bovik, "Objective quality assessment of multiply distorted images," in *2012 Conference record of the forty sixth asilomar conference on signals, systems and computers (ASILOMAR)*. IEEE, 2012, pp. 1693–1697.

[23] H. Lin, V. Hosu, and D. Saupe, "Koniq-10k: Towards an ecologically valid and large-scale iqa database," *arXiv preprint arXiv:1803.08489*, 2018.

[24] D. Ghadiyaram and A. C. Bovik, "Massive online crowdsourced study of subjective and objective picture quality," *IEEE Transactions on Image Processing*, vol. 25, no. 1, pp. 372–387, 2015.

[25] L. Li, H. Zhu, G. Yang, and J. Qian, "Referenceless measure of blocking artifacts by tchebichef kernel analysis," *IEEE Signal Processing Letters*, vol. 21, no. 1, pp. 122–125, 2013.

[26] L. Li, W. Lin, X. Wang, G. Yang, K. Bahrami, and A. C. Kot, "No-reference image blur assessment based on discrete orthogonal moments," *IEEE Transactions on Cybernetics*, vol. 46, no. 1, pp. 39–50, 2015.

[27] H. Liu, N. Klomp, and I. Heynderickx, "A no-reference metric for perceived ringing artifacts in images," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 4, pp. 529–539, 2009.

[28] H. Lin, V. Hosu, and D. Saupe, "Kadid-10k: A large-scale artificially distorted iqa database," in *2019 Eleventh International Conference on Quality of Multimedia Experience (QoMEX)*. IEEE, 2019, pp. 1–3.

[29] W. Zhou, L. Shi, Z. Chen, and J. Zhang, "Tensor oriented no-reference light field image quality assessment," *IEEE Transactions on Image Processing*, vol. 29, pp. 4070–4084, 2020.

[30] M. A. Saad, A. C. Bovik, and C. Charrier, "Blind image quality assessment: A natural scene statistics approach in the dct domain," *IEEE Transactions on Image Processing*, vol. 21, no. 8, pp. 3339–3352, 2012.

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

[32] L. Zhang, L. Zhang, and A. C. Bovik, "A feature-enriched completely blind image quality evaluator," *IEEE Transactions on Image Processing*, vol. 24, no. 8, pp. 2579–2591, 2015.

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

[34] W. Xue, X. Mou, L. Zhang, A. C. Bovik, and X. Feng, "Blind image quality assessment using joint statistics of gradient magnitude and laplacian features," *IEEE Transactions on Image Processing*, vol. 23, no. 11, pp. 4850–4862, 2014.

[35] K. Gu, G. Zhai, X. Yang, and W. Zhang, "Using free energy principle for blind image quality assessment," *IEEE Transactions on Multimedia*, vol. 17, no. 1, pp. 50–63, 2015.

[36] P. Ye, J. Kumar, L. Kang, and D. Doermann, "Unsupervised feature learning framework for no-reference image quality assessment," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 1098–1105.

[37] J. Xu, P. Ye, Q. Li, H. Du, Y. Liu, and D. Doermann, "Blind image quality assessment based on high order statistics aggregation," *IEEE Transactions on Image Processing*, vol. 25, no. 9, pp. 4444–4457, 2016.

[38] D. Ghadiyaram and A. C. Bovik, "Perceptual quality prediction on authentically distorted images using a bag of features approach," *arXiv preprint arXiv:1609.04757*, 2016.

[39] Y. Fang, K. Ma, Z. Wang, W. Lin, Z. Fang, and G. Zhai, "No-reference quality assessment of contrast-distorted images based on natural scene statistics," vol. 22, no. 7, pp. 838–842, 2014.

[40] L. Shi, W. Zhou, Z. Chen, and J. Zhang, "No-reference light field image quality assessment based on spatial-angular measurement," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 4114–4128, 2020.

[41] Z. Chen, J. Lin, N. Liao, and C. W. Chen, "Full reference quality assessment for image retargeting based on natural scene statistics modeling and bi-directional saliency similarity," *IEEE Transactions on Image Processing*, vol. 26, no. 11, pp. 5138–5148, 2017.

[42] Q. Jiang, F. Shao, W. Lin, K. Gu, G. Jiang, and H. Sun, "Optimizing multistage discriminative dictionaries for blind image quality assessment," *IEEE Transactions on Multimedia*, vol. 20, no. 8, pp. 2035–2048, 2017.

[43] J. Guan, S. Yi, X. Zeng, W.-K. Cham, and X. Wang, "Visual importance and distortion guided deep image quality assessment framework," *IEEE Transactions on Multimedia*, vol. 19, no. 11, pp. 2505–2520, 2017.

[44] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a "completely blind" image quality analyzer," *IEEE Signal processing letters*, vol. 20, no. 3, pp. 209–212, 2012.

[45] Q. Li, W. Lin, J. Xu, and Y. Fang, "Blind image quality assessment using statistical structural and luminance features," *IEEE Transactions on Multimedia*, vol. 18, no. 12, pp. 2457–2469, 2016.

[46] B. Yan, B. Bare, and W. Tan, "Naturalness-aware deep no-reference image quality assessment," *IEEE Transactions on Multimedia*, vol. 21, no. 10, pp. 2603–2615, 2019.

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

[48] J. Kim, H. Zeng, D. Ghadiyaram, S. Lee, L. Zhang, and A. C. Bovik, "Deep convolutional neural models for picture-quality prediction: Challenges and solutions to data-driven image quality assessment," *IEEE Signal processing magazine*, vol. 34, no. 6, pp. 130–141, 2017.

[49] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[50] H. Talebi and P. Milanfar, "Nima: Neural image assessment," *IEEE Transactions on Image Processing*, vol. 27, no. 8, pp. 3998–4011, 2018.

[51] H. Zeng, L. Zhang, and A. C. Bovik, "A probabilistic quality representation approach to deep blind image quality prediction," *arXiv preprint arXiv:1708.08190*, 2017.

[52] X. Yang, F. Li, and H. Liu, "Ttl-iqa: Transitive transfer learning based no-reference image quality assessment," *IEEE Transactions on Multimedia*, 2020.

[53] J. Xu, W. Zhou, and Z. Chen, "Blind omnidirectional image quality assessment with viewport oriented graph convolutional networks," *IEEE Transactions on Circuits and Systems for Video Technology.*, vol. 31, no. 5, pp. 1724–1737, 2021.

[54] R. v. d. Berg, T. N. Kipf, and M. Welling, "Graph convolutional matrix completion," *arXiv preprint arXiv:1706.02263*, 2017.

[55] S. Yan, Z. Li, Y. Xiong, H. Yan, and D. Lin, "Convolutional sequence generation for skeleton-based action synthesis," in *Proceedings of the IEEE International Conference on Computer Vision*, 2019, pp. 4394–4402.

[56] J. Fu, W. Zhou, and Z. Chen, "Bayesian spatio-temporal graph convolutional network for traffic forecasting," *arXiv preprint arXiv:2010.07498*, 2020.

[57] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, "Graph attention networks," *arXiv preprint arXiv:1710.10903*, 2017.

[58] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," in *Advances in neural information processing systems*, 2017, pp. 1024–1034.

[59] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[60] F. Hu, Y. Zhu, S. Wu, L. Wang, and T. Tan, "Hierarchical graph convolutional networks for semi-supervised node classification," in *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019*. ijcai.org, 2019, pp. 4532–4539.

[61] S. A. Golestaneh and D. M. Chandler, "No-reference quality assessment of jpeg images via a quality relevance map," in *IEEE Signal Processing Letters*, vol. 21, no. 2. IEEE, 2013, pp. 155–158.

[62] R. Hassen, Z. Wang, and M. M. Salama, "Image sharpness assessment based on local phase coherence," in *IEEE Transactions on Image Processing*, vol. 22, no. 7, 2013, pp. 2798–2810.

[63] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[64] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *5th International Conference on Learning Representations, ICLR 2017*, 2017.

[65] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *2nd International Conference on Learning Representations, ICLR 2014*, 2014.

[66] B. Thomee, D. A. Shamma, G. Friedland, B. Elizalde, K. Ni, D. Poland, D. Borth, and L.-J. Li, "Yfcc100m: The new data in multimedia research," *Communications of the ACM*, vol. 59, no. 2, pp. 64–73, 2016.

[67] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015*, 2015.

[68] S. Bosse, D. Maniry, K.-R. Müller, T. Wiegand, and W. Samek, "Deep neural networks for no-reference and full-reference image quality assessment," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 206–219, 2017.

[69] G. C. Linderman, M. Rachh, J. G. Hoskins, S. Steinerberger, and Y. Kluger, "Fast interpolation-based t-sne for improved visualization of single-cell rna-seq data," *Nature methods*, vol. 16, no. 3, pp. 243–245, 2019.

[70] J. Kim and S. Lee, "Fully deep blind image quality predictor," *IEEE Journal of selected topics in signal processing*, vol. 11, no. 1, pp. 206–220, 2016.

[71] D. Li, T. Jiang, W. Lin, and M. Jiang, "Which has better visual quality: The clear blue sky or a blurry animal?" *IEEE Transactions on Multimedia*, vol. 21, no. 5, pp. 1221–1234, 2018.

[72] W. Zhang, K. Ma, G. Zhai, and X. Yang, "Uncertainty-aware blind image quality assessment in the laboratory and wild," *IEEE Transactions on Image Processing*, vol. 30, pp. 3474–3486, 2021.

[73] H. Zhu, L. Li, J. Wu, W. Dong, and G. Shi, "Generalizable no-reference image quality assessment via deep meta-learning," *IEEE Transactions on Circuits and Systems for Video Technology*, 2021.

[74] J. Wu, J. Ma, F. Liang, W. Dong, G. Shi, and W. Lin, "End-to-end blind image quality prediction with cascaded deep neural network," *IEEE Transactions on Image Processing*, vol. 29, pp. 7414–7426, 2020.

[75] W. Zhang, K. Ma, G. Zhai, and X. Yang, "Learning to blindly assess image quality in the laboratory and wild," *arXiv preprint arXiv:1907.00516*, 2019.

[76] X. Min, G. Zhai, K. Gu, X. Yang, and X. Guan, "Objective quality evaluation of dehazed images," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 8, pp. 2879–2892, 2018.

[77] Q. Wu, L. Wang, K. N. Ngan, H. Li, F. Meng, and L. Xu, "Subjective and objective de-raining quality assessment towards authentic rain image," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 11, pp. 3883–3897, 2020.

[78] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737*, 2017.

Simeng Sun received the B.Sc. degree in information engineering from South China University of Technology in 2019. She is currently pursuing the M.Sc. degree with the Department of Electronic Engineering and Information Science, University of Science and Technology of China. Her current research interests include image quality assessment, image/video compression, computer vision and deep learning.

Tao Yu is currently pursuing the PhD degree with the Department of Electronic Engineering and Information Science, University of Science and Technology of China. He received the B. S. degree in Electronics and Information Engineering in Anhui University in 2018. His research interests include computer vision and reinforcement learning.

Jiahua Xu received the M.Sc. degree from the Department of Electronic Engineering and Information Science, University of Science and Technology of China, Hefei, China. She is currently an Engineer with Tencent Shannon Lab. Her current research interests include image/video quality assessment, computer vision and deep learning.

Wei Zhou (S'19) is currently a Postdoctoral Fellow in the Dept. of Electrical and Computer Engineering at University of Waterloo, Canada and received the Ph.D. degree from the University of Science and Technology of China in 2021 (joint with the University of Waterloo from 2019 to 2021). Dr. Zhou was a visiting graduate student in National Institute of Informatics, Tokyo, Japan in 2017; a research assistant with Intel from 2016 to 2018; a research intern in Microsoft and Alibaba, in 2018 and 2019, respectively. Dr. Zhou's research interests span multimedia computing, perceptual image processing, and computational vision.

Zhibo Chen (M'01-SM'11) received the B. Sc., and Ph.D. degree from Department of Electrical Engineering Tsinghua University in 1998 and 2003, respectively. He is now a full professor in University of Science and Technology of China. His research interests include image and video compression, visual quality of experience assessment, learning based visual signal representation and coding. He has more than 160 publications and over 80 granted patent applications. Some of his standard proposals have been adopted in MPEG/VCEG on video coding and ITU-T P.1202 on video quality assessment. He won the prestigious National Natural Science Award (second class) in 2018. He is IEEE senior member, Secretary of IEEE Visual Signal Processing and Communications Technical Committee (VSPC-TC). He has served as Associate Editor of IEEE Trans. on Circuits and Systems for Video Technology, Guest Editor for the IEEE Open Journal of Circuits and Systems, TPC chair of IEEE PCS 2019 and organization committee member of ICIP 2017 and ICME 2013, served as TPC member in IEEE ISCAS and IEEE VCIP.