RankDVQA-mini: Knowledge Distillation-Driven Deep Video Quality Assessment

Chen Feng[1], Duolikun Danier[1], Haoran Wang[1], Fan Zhang[1], Benoit Vallade[2], Alex Mackin[2], and David Bull[1]

[1]*Visual Information Laboratory, University of Bristol, Bristol, BS1 5DD, United Kingdom*
[1]*{chen.feng, duolikun.danier, yp22378, fan.zhang, dave.bull}@bristol.ac.uk*
[2]*Amazon Prime Video, 1 Principal Place, Worship Street, London, EC2A 2FA, United Kingdom*
[2]*{valladeb, acmackin}@amazon.co.uk*

2024 Picture Coding Symposium (PCS) | 979-8-3503-5848-3/24/$31.00 ©2024 IEEE | DOI: 10.1109/PCS60826.2024.10566364

***Abstract*—Deep learning-based video quality assessment (deep VQA) has demonstrated significant potential in surpassing conventional metrics, with promising improvements in terms of correlation with human perception. However, the practical deployment of such deep VQA models is often limited due to their high computational complexity and large memory requirements. To address this issue, we aim to significantly reduce the model size and runtime of one of the state-of-the-art deep VQA methods, RankDVQA, by employing a two-phase workflow that integrates pruning-driven model compression with multi-level knowledge distillation. The resulting lightweight full reference quality metric, RankDVQA-mini, requires less than 10% of the model parameters compared to its full version (14% in terms of FLOPs), while still retaining a quality prediction performance that is superior to most existing deep VQA methods. The source code of the RankDVQA-mini has been released at https://chenfeng-bristol.github.io/RankDVQA-mini/ for public evaluation.**

***Index Terms*—Video quality assessment, deep learning, model compression, knowledge distillation, RankDVQA-mini**

I. Introduction

Objective video quality assessment plays an essential role in many video processing applications [1]. It can, for example, be used for comparing the performance of compression algorithms, or within these algorithms to guide model optimisation (e.g., in rate-quality optimisation or as a loss function for training learning-based methods) [2, 3]. Over the last two decades, VQA methods have experienced significant advances, from the conventional quality metrics based on classic signal measures to more recent contributions optimised using deep learning techniques.

The most commonly used conventional quality metrics[1] are PSNR and SSIM [4], which measure pixel-wise distortions and the similarity between the test and reference content, respectively. To improve their correlation with visual perception, many perceptual-inspired objective quality metrics have been developed, including SSIM variants [5–7], VQM [8], MOVIE [9], PVM [10] and MAD [11]. Enhanced methods exist [12, 13] that integrate these perceptual quality metrics together with video features into a regression-based framework to further improve the overall correlation with subjective opinions. One important metric in this class is VMAF [14]. VMAF has been widely adopted for evaluating the performance of video compression and processing algorithms due to its excellent and consistent performance.

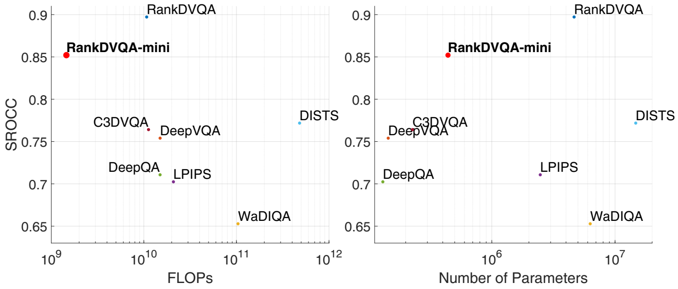

Fig. 1. Complexity (in terms of either FLOPs or model size) versus correlation performance (in SROCC) plot for benchmarked deep VQA methods. FLOPs are calculated for inference on a 256×256 sequence of 12 frames.

More recently, VQA metrics have been further enhanced through deep learning techniques. Various metrics have been reported, which have been built using Convolutional Neural Networks (CNNs); notable examples include DeepVQA [15], C3DVQA [16], LPIPS [17] and DISTS [18]. Although these methods have been reported to offer promising results compared to conventional and regression-based VQA methods, they are constrained by the lack of reliable large and diverse training databases (the model is typically trained on a small video quality dataset with limited subjective ground truth labels) and do not show consistent generalisation performance. To address this issue, RankDVQA [19] was proposed that employs a ranking-inspired training methodology. This supports the use of a large scale training database for model optimisation, and achieves the state-of-the-art generalisation performance compared to other existing methods.

Although deep VQA methods offer the potential to outperform conventional and regression-based quality metrics, they are often associated with high computational complexity, which restricts their deployment in practical applications [20]. To address this challenge, network pruning [21] and

Research reported in this paper was supported by an Amazon Research Award, Fall 2022 CFP. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not reflect the views of Amazon. We also appreciate the funding from the China Scholarship Council, University of Bristol, and the UKRI MyWorld Strength in Places Programme (SIPF00006/1).

[1] In this work, we solely focus on full reference scenarios where the reference content of the distorted video is available.

knowledge distillation [22, 23] techniques can be employed to prune a large network (denoted by "teacher") and transfer its knowledge to the pruned smaller model (the "student"). Such techniques have been applied in numerous fields [24–26], demonstrating that a compact model can achieve comparable performance to its larger counterpart.

In this work, we apply these techniques to deep VQA, producing a lightweight network that achieves competitive performance without the need to incorporate additional modules. Specifically, we employ a two-stage pipeline to condense the RankDVQA [19] network. RankDVQA is selected due to its state-of-the-art performance and model generalisation capabilities. Inspired by previous work [24, 27], we first apply a sparsity-inducing pruning technique to substantially reduce the number of RankDVQA parameters - retaining only 10% of the original model's parameters. The pruned model is then trained using a multi-level knowledge distillation strategy [28], learning from the teacher model, a pre-trained original RankDVQA. As shown in Fig. 1, the final compact model, RankDVQA-mini, retains 96% of RankDVQA's performance in terms of SROCC, while removing 90.12% of its parameters. Moreover, RankDVQA-mini reduces the Floating Point Operations (FLOPs) count of the original model by 86.42% (from 10.731G to 1.457G). We believe that this is the first time that model pruning and knowledge distillation have been used to optimise a deep VQA metric – a step towards low-complexity deep VQA.

II. Proposed Method

RankDVQA has been selected as the anchor VQA model for complexity reduction in this work. RankDVQA [19] is one of the latest and best performing learning-based video quality metrics. It is based on a ranking-inspired training strategy that enables the development of large and reliable training databases without performing expensive subjective tests. It consistently achieves superior performance compared to other conventional, regression-based and deep VQA methods.

The architecture of RankDVQA consists of two parts: the PQANet, which uses convolutional and SWIN transformer [29] layers for feature extraction and local quality prediction, and the STANet, which refines the assessment using adaptive spatio-temporal pooling. Since the model size of STANet (14.0K parameters) is much smaller than that of PQANet (4.59M), in this work we solely focus on reducing the complexity of the PQANet model.

A. Pruning RankDVQA

Sparsity-inducing Optimisation. Model pruning in the context of PQANet aims to simplify the network by inducing sparsity in parameters, thereby providing guidance for removing unnecessary model parameters while maintaining its performance. This is achieved by adding an L1 norm regularisation term to the training loss function, $\mathcal{L}_{prune}$, and applying the OBProx-SG optimiser [30]:

$$\mathcal{L}_{prune} = \mathcal{L}_{s1} + \lambda \cdot ||\theta||_1. \tag{1}$$

Here, $\mathcal{L}_{s1}$ represents the original binary cross entropy loss for training the PQANet [19]. λ is a hyper-parameter that controls the sparsity level (its empirical value equals 0.1). θ refers to the parameters of the PQANet model [30].

The optimisation process here is expected to identify any relatively unimportant parameters. After around 30 epochs, the sparse model retains those parameters that make a relatively significant contribution to overall model performance. The number of non-zero parameters reduces from around 4.59 million to 0.44 million. This compact model serves as a starting point for subsequent model compression.

Model Pruning. The sparse model obtained above is an important indicator for model pruning. We define the density of each layer, $\mathcal{D}_L$, as the proportion of its non-zero parameters:

$$\mathcal{D}_L = \frac{\text{number of nonzero parameters in } L}{\text{total parameters of } L}, \tag{2}$$

where L denotes a specific layer with parameters. Consequently, the contribution of each layer to the model performance can be identified. In the pruning stage, the density value is employed as the compression ratio of each layer.

To prune the model, the redundant channels in the original model are removed, starting from the last layer and proceeding backwards. The number of input channels $C_{in,L}$ for any given layer, L, is reduced to $\mathcal{D}_L \cdot C_{in,L}$, with $\mathcal{D}_L$ symbolising the compression ratio equivalent to the layer's density. Adjacent layers are then adjusted in tandem to maintain the integrity of the network. As a result, the number of output channels in these layers is decreased proportionally. The final compact PQANet model contains only 9.58% of the original model weights due to the significant reduction of redundant channels.

B. Multi-level Knowledge Distillation

After model pruning, knowledge distillation is employed to enhance the performance of the compact PQANet model. Our approach differs from the traditional knowledge distillation framework [22–25], which only focuses on the difference between the output of student and teacher models at the instance level. Inspired by recent work [28], we adopt a multi-level logit knowledge distillation strategy, which extends this process to two additional levels: batch and class, to enhance the learning efficiency of the student model from the teacher. Fig. 2 shows the workflow of this approach.

Instance-level Alignment. This level inherits the conventional knowledge distillation method [23–25], which minimises the divergence between the outputs of the teacher and student models at the instance level. During optimisation, the student model is trained to decrease the difference in prediction from the teacher model for each instance. Specifically, PQANet takes as input (in the training process - details can be found in [19]) two distorted sequences $\mathbf{D}_1, \mathbf{D}_2$ and their respective references $\mathbf{R}_1, \mathbf{R}_2$, and outputs the probability of $\mathbf{D}_1$ having higher quality than $\mathbf{D}_2$. Let $p_{teacher}$ denote the output of the teacher network (the original PQANet) on one such training instance, and let $p_{student}$ be the output of the

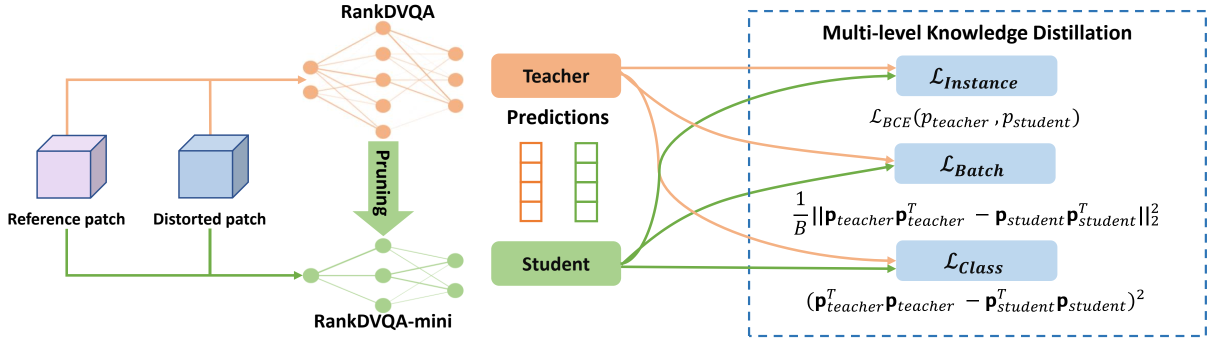
RankDVQA
Reference patch
Distorted patch
Pruning
RankDVQA-mini
Teacher
Predictions
Student
Multi-level Knowledge Distillation
$\mathcal{L}_{Instance}$
$\mathcal{L}_{BCE}(p_{teacher}, p_{student})$
$\mathcal{L}_{Batch}$
$\frac{1}{B}||\mathbf{p}_{teacher}\mathbf{p}_{teacher}^T - \mathbf{p}_{student}\mathbf{p}_{student}^T||_2^2$
$\mathcal{L}_{Class}$
$(\mathbf{p}_{teacher}^T\mathbf{p}_{teacher} - \mathbf{p}_{student}^T\mathbf{p}_{student})^2$

Fig. 2. The framework of the RankDVQA-mini with the multi-level knowledge distillation, after the model pruning and compression, obtaining the RankDVQA (teacher) and RankDVQA-mini (student) predictions. The predictions are matched respectively through multi-level alignment, which consists of instance-level, batch-level, and class-level alignments. We take batch size B = 4 as an example to demonstrate this approach.

student network (the pruned PQANet) for the same instance. The instance level loss function reads:

$$\mathcal{L}_{\text{Instance}} = \mathcal{L}_{\text{BCE}}(p_{teacher}, p_{student}). \tag{3}$$

Here the Binary Cross Entropy (BCE) loss measures the divergence between $P_{teacher}$ and $P_{student}$,

$$\begin{aligned}\mathcal{L}_{\text{BCE}} = -\big(&p_{teacher}\log(p_{student}) \\ &+ (1 - p_{teacher})\log(1 - p_{student})\big).\end{aligned} \tag{4}$$

Batch-level Alignment. At the batch level, we aim to train the student to mimic the inner-instance correlation (within a training batch) predicted by the teacher. Specifically, let $\mathbf{p}_{teacher}, \mathbf{p}_{student} \in (0,1)^{B\times 1}$ denote the outputs of the teacher and the student on a batch of B training instances, the batch-level distillation loss is defined as

$$\mathcal{L}_{\text{Batch}} = \frac{1}{B}||\mathbf{p}_{teacher}\mathbf{p}_{teacher}^T - \mathbf{p}_{student}\mathbf{p}_{student}^T||_2^2, \tag{5}$$

where the Gram matrix $\mathbf{pp}^T \in (0,1)^{B\times B}$ models the inner-instance relationships.

Class-level Alignment. The class-level alignment in [28], matches the category correlation modelled by the student and the teacher, i.e. the relationship between different classes (e.g. the 1000 classes in ImageNet [31]). In our case, the classification is binary, and the loss function is written as:

$$\mathcal{L}_{\text{Class}} = (\mathbf{p}_{teacher}^T\mathbf{p}_{teacher} - \mathbf{p}_{student}^T\mathbf{p}_{student})^2, \tag{6}$$

Multi-level Loss. Three levels of loss functions will be combined to act as the knowledge distillation term in the training loss function:

$$\mathcal{L}_{\text{Multi-level}} = \mathcal{L}_{\text{Instance}} + \mathcal{L}_{\text{Batch}} + L_{\text{Class}}, \tag{7}$$

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{Multi-level}} + \alpha\mathcal{L}_{\text{s1}}, \tag{8}$$

where $\mathcal{L}_{\text{s1}}$ denotes the original loss function [19] for the PQANet as mentioned in Equation (1), and α is a hyperparameter to allocate weight to the distillation loss term (set as 0.1). Our method combines three losses at different levels, guiding the student model to mimic the teacher model's behaviour across instance-level, batch-level and class-level predictions.

III. Results and Discussion

The compact version of PQANet was trained for 30 epochs on the same training set as the original model [19], which consists of approximately 20K patch pairs from the CVPR 2022 CLIC video compression challenge [35] and BVI-DVC [36] datasets. The STANet used for spatio-temporal pooling in the second stage remains the same, and has been retrained based on the output of PQANet, also with the same training databases as in [19]. We use AdaMAX optimisation [37] with hyper-parameters $\beta 1$=0.9 and $\beta 2$=0.999 in the training process. Training and evaluation were executed on the compute cluster [38] at the University of Bristol (GPU nodes with 2.4GHz Intel CPUs and two NVIDIA P100 graphic cards).

To evaluate model generalisation performance, we followed the same experiment setup as in [19], using eight different HD VQA datasets for performance benchmarking: NFLX [14], NFLX-P [14], BVI-HD [39], CC-HD [40], CC-HDDO [41], MCL-V [42], SHVC [43], VQEGHD3 [44]. These databases contain various distortion types produced by spatial resolution adaptation and video compression.

To benchmark the performance of RankDVQA-mini we compared its correlation performance with eleven full reference quality assessment methods, including three conventional quality metrics: PSNR, SSIM [4], MS-SSIM [5][2]; and seven deep quality assessment methods[3]: WaDIQA [32], DeepQA [33], DeepVQA [15], C3DVQA [16], DISTS [18], LPIPS [17], RankDVQA [19], and two regression-based VQA approach, ST-GREED [34] and VMAF [14].

To assess the correlation performance of these VQA methods with subjective ground truth, the Spearman Rank Order Correlation Coefficient (SROCC) was calculated, for each database, between predicted quality indices and subjective scores. Additionally, to test the significance of performance difference, an F-test was performed between the proposed method, RankDVQA-Mini, and other tested metrics based on

[2] It is noted that these image quality metrics are calculated based on luma components only.

[3] The selection of deep VQA methods is based on the performance reported in their original publications and on the availability of their pre-trained models.

TABLE I

Performance and complexity figures of the proposed method and other benchmark approaches on eight test databases. The values in each cell x(y) correspond to the SROCC value (x) and F-test result (y) at 95% confidence interval. y=1 suggests that the metric is superior to RankDVQA-mini (y=-1 if the opposite is true), while y=0 indicates that there is no significant difference.

SROCC(F-test)	NFLX	NFLX-P	BVI-HD	CC-HD	CC-HDDO	MCL-V	SHVC	VQEGHD3	**Overall**	FLOPs (G)	#P (M)
PSNR	0.6218 (-1)	0.6596 (-1)	0.6143 (-1)	0.6166 (-1)	0.7497 (-1)	0.4640 (-1)	0.7380 (-1)	0.7518 (-1)	0.6520	—	—
SSIM [4]	0.5638 (-1)	0.6054 (-1)	0.5992 (-1)	0.7194 (-1)	0.8026 (-1)	0.4018 (-1)	0.5446 (-1)	0.7361 (-1)	0.6216	—	—
MS-SSIM [5]	0.7136 (-1)	0.7394 (-1)	0.7652 (0)	0.7534 (-1)	0.8321 (0)	0.6306 (-1)	0.8007 (0)	0.8457 (0)	0.7601	—	—
WaDIQA [32]	0.5713 (-1)	0.6593 (-1)	0.6646 (-1)	0.6516 (-1)	0.7041 (-1)	0.6072 (-1)	0.6731 (-1)	0.6910 (-1)	0.6528	104.04	6.287
DeepQA [33]	0.7298 (-1)	0.6995 (-1)	0.7106 (-1)	0.6202 (-1)	0.6705 (-1)	0.6832 (-1)	0.7176 (-1)	0.7881 (-1)	0.7024	14.976	0.131
LPIPS [17]	0.6793(-1)	0.7859 (-1)	0.6670 (-1)	0.6838 (-1)	0.7678 (-1)	0.6579 (-1)	0.6360 (-1)	0.8075 (0)	0.7107	20.857	2.472
DeepVQA [15]	0.7352 (-1)	0.7609 (-1)	0.7330 (-1)	0.6924 (-1)	0.8120 (0)	0.6142 (-1)	0.8041 (0)	0.7805 (-1)	0.7540	14.990	0.144
C3DVQA [16]	0.7730 (-1)	0.7714 (-1)	0.7393 (-1)	0.7203 (-1)	0.8137 (0)	0.7126 (0)	0.8194 (0)	0.7329 (-1)	0.7641	11.236	0.227
DISTS [18]	0.7787 (-1)	0.9325 (0)	0.7030 (-1)	0.6303 (-1)	0.7442 (-1)	0.7792 (0)	0.7813 (0)	0.8254 (0)	0.7718	481.07	14.715
ST-GREED [34]	0.7470 (-1)	0.7445 (-1)	0.7769 (0)	0.7738 (0)	0.8259 (0)	0.7226 (0)	0.7946 (0)	0.8079 (0)	0.7842	—	—
VMAF 0.6.1 [14]	0.9254 (0)	0.9104 (0)	0.7962 (0)	0.8723 (0)	0.8783 (0)	0.7766 (0)	0.9114 (0)	0.8442 (0)	0.8644	—	—
FR-RankDVQA [19]	0.9393 (0)	0.9184 (0)	0.8659 (0)	0.8991 (0)	0.9037 (0)	0.8391 (0)	0.9142 (0)	0.8979 (0)	0.8972	10.731	4.608
RankDVQA-mini	**0.8846**	**0.8748**	**0.8135**	**0.8479**	**0.8890**	**0.7592**	**0.8819**	**0.8661**	**0.8521**	**1.457**	**0.455**

the residuals between the predicted quality indices (after a non-linear regression) and the subjective ground truth [10, 45].

Table I summarises the quantitative results of all tested VQA methods in terms of SROCC values, F-test results and complexity figures (number of model parameters and Floating Point Operations (FLOPs)). Note that the model size figures presented for RankDVQA and RankDVQA-mini are for both PQANet and the STANet. It can be observed that, with only 9.87% of the parameters and 13.57% of FLOPs compared to the original RankDVQA, RankDVQA-mini still achieves competitive correlation performance, outperforming all other deep VQA methods, including DISTS, C3DVQA and LPIPS, and conventional quality metrics: PSNR, SSIM and MS-SSIM. It is noted that RankDVQA-mini does not outperform VMAF (although the overall SROCC is competitive). However, this is the first step towards reducing the complexity of deep VQA metrics (which are the state of the art in terms of performance) for their practical use and we show a promising trade-off between complexity and performance. It is our hope to inspire further work on developing compact and efficient deep VQA models that surpass VMAF. Finally, according to the F-test, RankDVQA-mini shows significant advantage over most compared methods on various test sets, and its differences from VMAF and the original RankDVQA are insignificant.

Figure 1 provides a more intuitive comparison between RankDVQA-mini and other deep VQA methods in terms of performance and complexity. It can be observed that the proposed method achieves an excellent trade off between correlation performance and complexity (model size and FLOPs) - it requires a similar level (in the same order of magnitude) of model size and FLOPs as DeepVQA, C3DVQA and DeepQA, but achieves evident performance improvement (confirmed by the F-test results in TABLE I).

IV. Conclusion

In this work, we present a new lightweight and effective deep video quality assessment method, RankDVQA-mini, by applying a two phase complexity reduction workflow to the state-of-the-art deep quality metric, RankDVQA. The resulting compact model retains the superior performance of its original counterpart, but with a reduction of 90% in terms of model parameters and 14% of FLOPs. Future work should focus on further runtime reductions and more effective knowledge distillation to improve model performance.

References

[1] D. R. Bull and F. Zhang, *Intelligent image and video compression: communicating pictures*. Academic Press, 2021.

[2] P. Ndjiki-Nya, D. Doshkov, H. Kaprykowsky, F. Zhang, D. Bull, and T. Wiegand, "Perception-oriented video coding based on image analysis and completion: A review," *Signal Processing: Image Communication*, vol. 27, no. 6, pp. 579–594, 2012.

[3] D. Ma, F. Zhang, and D. R. Bull, "CVEGAN: A perceptually-inspired gan for compressed video enhancement," 2020.

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

[5] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multi-scale structural similarity for image quality assessment," in *Proc. Asilomar Conf. on Signals, Systems and Computers*, vol. 2. IEEE, 2003, p. 1398.

[6] A. Rehman, K. Zeng, and Z. Wang, "Display device-adapted video quality-of-experience assessment," in *Human Vision and Electronic Imaging XX*, vol. 9394. International Society for Optics and Photonics, 2015, p. 939406.

[7] Z. Wang, L. Lu, and A. C. Bovik, "Video quality assessment based on structural distortion measurement," *Signal processing: Image communication*, vol. 19, no. 2, pp. 121–132, 2004.

[8] M. H. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Trans. on broadcasting*, vol. 50, no. 3, pp. 312–322, 2004.

[9] K. Seshadrinathan and A. C. Bovik, “Motion tuned spatio-temporal quality assessment of natural videos,” *IEEE Trans. on Image Processing*, vol. 19, no. 2, pp. 335–350, 2010.

[10] F. Zhang and D. R. Bull, “A perception-based hybrid model for video quality assessment,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 26, no. 6, pp. 1017–1028, 2016.

[11] P. V. Vu, C. T. Vu, and D. M. Chandler, “A spatiotemporal most-apparent-distortion model for video quality assessment,” in *2011 18th IEEE International Conf. on Image Processing*, 2011, pp. 2505–2508.

[12] J. Y. Lin, T.-J. Liu, E. C.-H. Wu, and C.-C. J. Kuo, “A fusion-based video quality assessment (FVQA) index,” in *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2014 Asia-Pacific*. IEEE, 2014, pp. 1–5.

[13] F. Zhang, A. Katsenou, C. Bampis, L. Krasula, Z. Li, and D. Bull, “Enhancing VMAF through new feature integration and model combination,” in *2021 Picture Coding Symposium*. IEEE, 2021, pp. 1–5.

[14] Z. Li, A. Aaron, I. Katsavounidis, A. Moorthy, and M. Manohara, “Toward a practical perceptual video quality metric,” *The Netflix Tech Blog*, 2016.

[15] W. Kim, J. Kim, S. Ahn, J. Kim, and S. Lee, “Deep video quality assessor: From spatio-temporal visual sensitivity to a convolutional neural aggregation network,” in *Proceedings of the European Conf. on Computer Vision*, 2018, pp. 219–234.

[16] M. Xu, J. Chen, H. Wang, S. Liu, G. Li, and Z. Bai, “C3DVQA: Full-reference video quality assessment with 3d convolutional neural network,” in *ICASSP 2020 - 2020 IEEE International Conf. on Acoustics, Speech and Signal Processing*, 2020, pp. 4447–4451.

[17] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, “The unreasonable effectiveness of deep features as a perceptual metric,” in *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition)*, June 2018.

[18] K. Ding, K. Ma, S. Wang, and E. P. Simoncelli, “Image quality assessment: Unifying structure and texture similarity,” *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 44, no. 5, pp. 2567–2581, 2020.

[19] C. Feng, D. Danier, F. Zhang, and D. R. Bull, “Rankdvqa: Deep vqa based on ranking-inspired hybrid training,” *arXiv preprint arXiv:2202.08595*, 2022.

[20] J. Gou, B. Yu, S. J. Maybank, and D. Tao, “Knowledge distillation: A survey,” *International Journal of Computer Vision*, vol. 129, pp. 1789–1819, 2021.

[21] R. Reed, “Pruning algorithms-a survey,” *IEEE Trans. on Neural Networks*, vol. 4, no. 5, pp. 740–747, 1993.

[22] L. Wang and K.-J. Yoon, “Knowledge distillation and student-teacher learning for visual intelligence: A review and new outlooks,” *IEEE Trans. on pattern analysis and machine intelligence*, vol. 44, no. 6, pp. 3048–3068, 2021.

[23] G. Hinton, O. Vinyals, J. Dean *et al.*, “Distilling the knowledge in a neural network,” *arXiv preprint arXiv:1503.02531*, vol. 2, no. 7, 2015.

[24] C. Morris, D. Danier, F. Zhang, N. Anantrasirichai, and D. R. Bull, “ST-MFNet Mini: Knowledge distillation-driven frame interpolation,” *arXiv preprint arXiv:2302.08455*, 2023.

[25] W. Park, D. Kim, Y. Lu, and M. Cho, “Relational knowledge distillation,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 3967–3976.

[26] A. Mishra and D. Marr, “Apprentice: Using knowledge distillation techniques to improve low-precision network accuracy,” *arXiv preprint arXiv:1711.05852*, 2017.

[27] T. Ding, L. Liang, Z. Zhu, and I. Zharkov, “Cdfi: Compression-driven network design for frame interpolation,” in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8001–8011.

[28] Y. Jin, J. Wang, and D. Lin, “Multi-level logit distillation,” in *Proceedings of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, 2023, pp. 24 276–24 285.

[29] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, “Swin transformer: Hierarchical vision transformer using shifted windows,” in *Proceedings of the IEEE/CVF International Conf. on Computer Vision*, 2021, pp. 10 012–10 022.

[30] T. Chen, T. Ding, B. Ji, G. Wang, Y. Shi, J. Tian, S. Yi, X. Tu, and Z. Zhu, “Orthant based proximal stochastic gradient method for L_1 $L1$-regularized optimization,” in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part III*. Springer, 2021, pp. 57–73.

[31] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, “Imagenet large scale visual recognition challenge,” *International journal of computer vision*, vol. 115, pp. 211–252, 2015.

[32] 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 Trans. on Image Processing*, vol. 27, no. 1, pp. 206–219, 2018.

[33] J. Kim and S. Lee, “Deep learning of human visual sensitivity in image quality assessment framework,” in *2017 IEEE Conf. on Computer Vision and Pattern Recognition*, 2017, pp. 1969–1977.

[34] P. C. Madhusudana, N. Birkbeck, Y. Wang, B. Adsumilli, and A. C. Bovik, “ST-GREED: Space-time generalized entropic differences for frame rate dependent video quality prediction,” *IEEE Trans. on Image Processing*, vol. 30, pp. 7446–7457, 2021.

[35] A. E. Guide, “the fifth workshop and challenge on learned image compression (video track),” *http://compression.cc/*, 2022.

[36] D. Ma, F. Zhang, and D. Bull, “BVI-DVC: a training database for deep video compression,” *IEEE Trans. on Multimedia*, 2021.

[37] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *arXiv preprint arXiv:1412.6980*, 2014.

[38] University of Bristol, “Bluecystal phase 4,” http://www.acrc.bris.ac.uk/acrc/phase4.htm, accessed: 1st May 2017.

[39] F. Zhang, F. Mercer Moss, R. Baddeley, and D. R. Bull, “BVI-HD: A video quality database for HEVC compressed and texture synthesised content,” *IEEE Trans. on Multimedia*, vol. 20, no. 10, pp. 2620–2630, October 2018.

[40] A. Katsenou, F. Zhang, M. Afonso, G. Dimitrov, and D. R. Bull, “BVI-CC: A dataset for research on video compression and quality assessment,” *Frontiers in Signal Processing*, vol. 2, p. 874200, 2022.

[41] A. V. Katsenou, F. Zhang, M. Afonso, and D. R. Bull, “A subjective comparison of AV1 and HEVC for adaptive video streaming,” in *Proc. IEEE Int Conf. on Image Processing*, 2019.

[42] J. Y. Lin, R. Song, C.-H. Wu, T. Liu, H. Wang, and C.-C. J. Kuo, “MCL-V: A streaming video quality assessment database,” *Journal of Visual Communication and Image Representation*, vol. 30, pp. 1–9, 2015.

[43] Y. He, Y. Ye, F. Hendry, Y.-K. Wang, and V. Baroncini, “SHVC verification test results,” in *the JCT-VC meeting, JCTVC-W0095*. ITU-T, ISO/IEC, 2016.

[44] Video Quality Experts Group, “Report on the validation of video quality models for high definition video content,” VQEG, Tech. Rep., 2010. [Online]. Available: http://www.its.bldrdoc.gov/vqeg/projects/hdtv/hdtv.aspx

[45] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack, “Study of subjective and objective quality assessment of video,” *IEEE Trans. on Image Processing*, vol. 19, pp. 335–350, 2010.