Predicting local distortions introduced by AV1 using Deep Features

Andréas Pastor, Lukáš Krasula, Xiaoqing Zhu, Zhi Li, Patrick Le Callet

Predicting local distortions introduced by AV1 using Deep Features

Andréas Pastor[1], Lukáš Krasula[2], Xiaoqing Zhu[2], Zhi Li[2], Patrick Le Callet[1, 3]
[1]*Nantes Université, Ecole Centrale Nantes, CNRS, LS2N, UMR 6004, F-44000 Nantes, France*
[2]*Netflix Inc., Los Gatos, CA, USA* [3]*Institut universitaire de France (IUF)*
[1]{andreas.pastor, patrick.lecallet}@univ-nantes.fr, [2]{lkrasula, xzhu, zli}@netflix.com

***Abstract*—Semantics extracted by filters in deep learning networks correlate well with how human eyes perceive distortions. These methods (e.g., LPIPS, PieAPP, etc.) rely on the relative difference in activation between feature maps in pairs of references and distorted patches. However, Deep Feature extraction can be expensive to compute as a difference of latent code between reference and distorted frames. Therefore, it is challenging to integrate them into the decision process of modern video codecs like AV1, making thousands of encoding trials during exhaustive Rate-Distortion Optimization (RDO) searches. In this study, we present a method using deep features to predict the distortion perceived locally by human eyes in AV1-encoded videos. The prediction relies on Deep Features extracted from the reference frame only to weigh the Mean Squared Error (MSE) introduced during encoding. This approach will make integration into video codecs easier as a pre-processing step before starting encoding. We show the superiority of the proposed metric against other Reference-Only metrics on a dataset of local distortions in videos. We achieve comparable performance as state-of-the-art Full-Reference video quality metrics.**

***Index Terms*—video quality, machine learning, local perception, dimensionality reduction, open video codec**

I. Introduction

A part of research in Computer Vision has focused on building Neural Networks to classify, locate, and track object instances. These networks with complex architectures rely successively on extracted, aggregated, and curated features. Traditional operations used for this process are convolutions, pooling, non-linear layers, and attention mechanisms. The outcomes of applying these operations are feature maps that can be classified into different categories. Low-level features, the first features extracted from pixels, are simple edge and texture detection filters. With the succession of layers and operations, features are spatially grouped into larger and larger regions to produce features more semantically representative of class concepts. These semantics are the final characteristics used to classify or detect objects.

Recent works [1]–[3] have shown that these Deep Features better correlate with how our eyes perceive distortions than existing image quality metrics. These metrics were benchmarked on various distortion types in static patches. In Image Quality Assessment, objective quality metrics, like PSNR and SSIM [4], rely only on pixel statistics and differences. These metrics are computationally efficient but have no semantic information to make decisions, resulting in lower performances. However, due to their simplicity and computational efficiency, these metrics have been favored to integrate into video codecs to tune and improve video encoding quality.

WPSNR [5] and improved version XPSNR [6] are block-based perceptually weighted PSNR metrics. Weights are derived from efficient spatial and temporal filtering in a reference frame to capture the Human Visual System (HVS) sensitivity to local distortions. These methods are computationally inexpensive to run and serve, for example, in HEVC and VVC video codecs, to perform subjectively optimized bit allocation.

Video Encoding research involves working on the tri-paradigm of compressing video data while maintaining acceptable visual quality and reasonable computation cost: crucial points for cloud encoding computation management, efficient storage, and transmission of videos to diverse client platforms.

To tune video quality locally in video coding algorithms like AV1, the optimizer selects at the Coding Unit (CU) level between different coded proposals of a reference block while accounting for the cost (i.e., bit rate) and the distortion introduced by signal compression. The distortion is estimated using the Sum of Squared Error (SSE) or equivalent in the transform domain (i.e., SATD). This fidelity paradigm guides the encoding between a reference block and its coded version. In *tune=WPSNR* mode of HEVC, the SSEs are weighted by the factors derived from WPSNR. Similarly, in AV1, *tune=ssim* flag enables a weighting of SSEs with factors proportional to reference blocks pixels variance.

The work in [7] proposes a benchmark of image and video quality metrics on small video tubes. These tubes were subjectively annotated with how humans perceived the distortion introduced by AV1 encoding. This work demonstrates how video quality metric VMAF [8], trained initially to predict video quality at a global scale, still performs relatively well on this dataset of tubes with small spatio-temporal horizons.

In this research work, we present a new metric to improve on WPSNR and XPSNR. Our metric derives weights to scale MSE distortions values, using information extracted by Neural Network to consider content semantics in predicting HVS local perception of distortions.

The following sections present, in section II, the dataset and what we call *tubes* for local distortions evaluation in videos. In section III, we explain how we extract features and the pooling techniques we consider to represent information extracted by Neural Networks efficiently. In section IV, we show the performances of our metric and compare it to other state-of-

the-art video quality metrics. Lastly, section V concludes the paper.

II. Subjective dataset

This section introduces our subjective dataset. From the reference sources (SRCs) of the VideoSet database [9], we extract *tubes*. A tube is a short video sequence of size 64×64 pixels and lasts 400ms, 12 frames at 30fps.

Motivations behind these *tubes* dimensions are the following. Human perception and gaze mechanisms inspire our spatio-temporal design choices. We incorporate how our eyes perform fixations to explore scenes and objects. These fixation events can last between a hundred and a few hundred milliseconds, hence the duration of our *tubes*. For the spatial resolution, we utilize the size of the fovea, which covers around 1° at the center of our field of view, translating to 60 pixels under standard viewing distance. Our tubes are aligned on the motion in the video to mimic our eyes' *Smooth Pursuit* on moving objects.

Before extracting the *tubes*, we encode all SRCs of the VideoSet database using AV1 encoder at fixed Quantization Parameter (QP) values, using *–cq-level* flag in libaom[1]. We generate 31 Processed Video Sequences (PVS) for each SRC using QP 3 to 63 with a step of 2.

We define a *tube-content* as a set ($Tube_{ref}$, $Tube_{D1}$, ..., $Tube_{Dn}$): with a first tube extracted in the SRC, $Tube_{ref}$, and N distorted version of it extracted from PVS. In the current dataset, N equals 5. Check Fig. 1 for *tube-contents* examples.

We define a Perceptual Difference curve (*PD-curve*) as the relation in a *tube-content* between its subjectively estimated perceived distortions (*PD-scores*) and the Mean Squared Error in Luma channel MSE_Y between the $Tube_{ref}$ and its corresponding $Tube_{Di}$. In fig. 2, we provide as an example the 54 *PD-curves* contained in the test set, 20% of the dataset size. The total dataset subjectively annotated contains 268 *PD-curves*.

The *tube-contents* in our subjective dataset are selected out of 100K following the clustering approach suggested in [7] for content selection. This clustering approach relies on the responses of various Full-Reference quality metrics on the degraded tubes.

The subjective data is collected in crowdsourcing by recruiting 1130 participants on Prolific[2]. Each participant is asked to perform 40 *quadruplets* comparisons, which took 6 minutes to complete on average. Subjective evaluations were made using a quadruplet preference-based scenario. For subjective quality assessment of image/video, it has been shown that Two-Alternative Forced Choice (2AFC) methods are more precise and sensitive than direct rating methods while reducing the cognitive load of participants. In our work, we used the Maximum Likelihood Difference Scaling (MLDS) [10] method. It efficiently selects stimuli to compare in a subjective study. In [11], a solution[3] is proposed to improve MLDS to select quadruplets for inter-content scaling efficiently. Inter-content scaling enables human observers to compare and rate the relative distortions introduced by codec across regions in a video frame [12]. The method has been compared with triplet-based and pairwise comparison subjective methodologies in [13] and validated for a crowdsourcing scenario in [14] where noisy annotations from outliers/spammers need to be handled.

[1]AV1 encoder v3.1.2, from AOM Alliance Open Media: https://aomedia.googlesource.com/aom/

[2]Prolific: https://www.prolific.co/

III. Proposed Model

This section details our model and how we extract Deep Features from popular Neural Networks (NN) architectures.

A. Feature extractors

We evaluate AlexNet [15], VGG16 [16], ResNet18-152 [17], EfficientNetB0-2 [18], SqueezeNet [19], ShuffleNetV2 [20] and MobileNetV3 [21] architectures. We include AlextNet, VGG16, and ResNet networks since they are popular choices for feature extraction and fine-tuning of quality assessment models [1]–[3], [22]. We choose SqueezeNet, ShuffleNetV2, and MobileNetV3 architectures, designed to be highly efficient and lightweight for mobile inference use cases. We pick the three most lightweight versions of EfficientNet for their fast inference speeds while having high classification performances. We fix the weights to their pre-trained versions on the ImageNet dataset [23] classification task.

Additionally, we remove the last activation layer and keep the remaining intermediate layers as feature extractors. We use the five *conv* layers for AlexNet and VGG16. In SqueezeNet, the first *conv* and the 6 *fire* modules. For ResNets, the first *conv1* and the 4 *conv2_x–conv5_x* modules. For EfficientNet networks, the first *ConvNormActivation* and the 7 *MBConv* modules. In ShuffleNet, the first *conv* and the 11 *InvertedResidual* modules. In MobileNetV3, the first *conv* and the 3 *stage* modules.

B. Features extraction and pooling from a reference tube

From a reference tube, a $(12, 3, 64, 64)$–tensor, we extract the feature maps from the K selected modules in the feature extractor, K $(12, C_k, H_i, W_i)$–tensors. C_i is the number of filters in module i. (H_i, W_i) are the feature map height and width.

After this operation, the feature maps are first averaged on the two spatial dimensions: we obtain K $(12, C_i)$–tensors. Then, to pool temporal activation, we apply two methods: a temporal *mean* averaging to get K $(1, C_i)$–tensors and a temporal *variance* pooling: another K $(1, C_i)$–tensors.

In summary, after flatting empty dimensions, a reference tube is represented by 2 (C)–tensors, where C is the sum of the K C_i. In the AlexNet context, C is equal to 1152. These two tensors, $MeanSem$ and $VarSem$, represent the semantics in a tube from low textural information to high-level semantics and its temporal variation, respectively.

[3]https://github.com/andreaspastor/MLDS_inter_content_scaling

Fig. 1. Example of *tube-contents* in the subjectively evaluated dataset. Each column represents a *tube-content* and its five distortion levels in increasing order from the top with the reference *tubes* to the bottom with the most distorted levels.

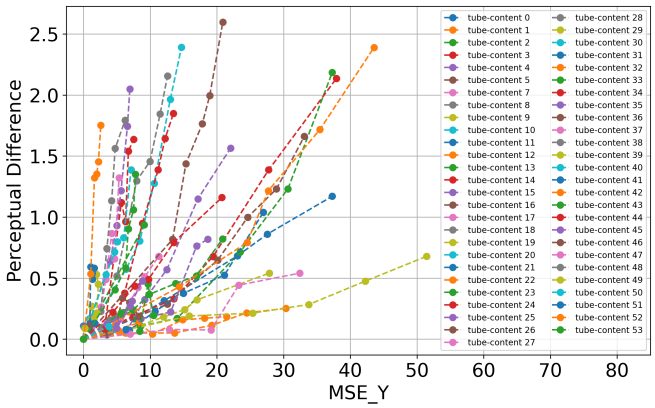

Fig. 2. The 54 *PD-curves* contained in the test set of the dataset. On the X-axis are the MSE in the Luma Channel between each reference tube and its corresponding distorted tubes. The perceptual differences participants estimated during the subjective study are on the Y-axis.

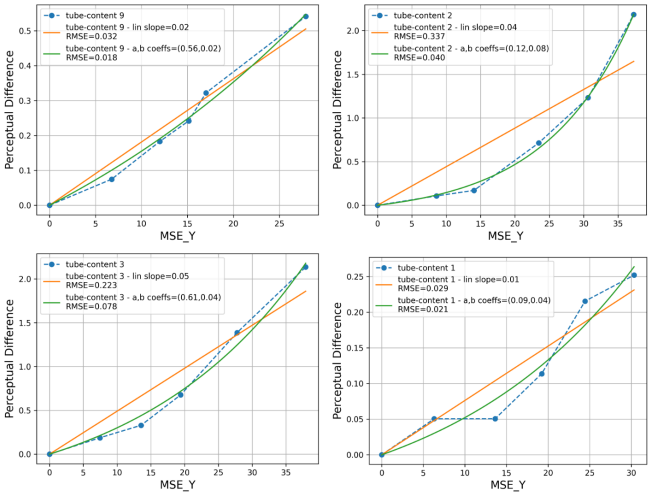

Fig. 3. Example of PD-curves fitting, in orange best linear fitting, in green best exp fitting. RMSE between individual PD-curves and fitted functions are provided.

C. Dimensionality reduction

To cope with the high dimensional space of the latent representation obtained per reference tube, we use Principal Components Analysis (PCA) to reduce the latent space size. We apply PCA separately on $MeanSem$ and $VarSem$ to easily track their contributions during training. The PCA projections are learned from our database of tubes. This database contains around 100K tube contents. This dataset is populated with tubes extracted from the large set of videos we encoded with libaom; see section II for more details.

D. Model Training and feature selection

Popular video quality metrics like VMAF [8] operate simple Machine Learning techniques to learn good aggregation of atoms features. We used a Support Vector Machine Regressor (SVR) to learn from the top K Principal Components (PC) of PCA projections.

We consider three variants to train the SVR model with our PD-curves: namely ***raw***, ***lin***, and ***exp***. For the ***raw*** configuration, we train an SVR to predict the raw subjective Perceptual Difference estimated between a reference and distorted tube. The model inputs are the PCA projected features extracted from the reference tube and the Mean Squared Error in the Luma Channel with the distorted tube. The model is responsible for learning the complete relationship inside a *PD-curve* in this configuration.

In the ***lin*** configuration, first, we extracted on each *PD-curve* of the dataset their best linear fitting slope. In figure 3, examples of slopes and fitted linear functions are provided in orange. The fitting function is:

$$PD'_{score} = A \times MSE_Y \tag{1}$$

The SVR model inputs in this training configuration are only the projected features from PCAs. The model outputs a linear slope, a weighting factor to transform MSE_Y values to perceptual scores.

In the ***exp*** configuration, a two parameters exponential fitting is applied on the *PD-curves*, in figure 3 (green). The exponential function is:

$$PD'_{score} = A \times (e^{B \times MSE_Y} - 1) \tag{2}$$

Similarly to ***lin*** configuration, the inputs are only the projected features from PCA. Two SVR models are employed

TABLE I
THE PROPOSED MODELS. WE REPORT THE NUMBER OF PRINCIPAL COMPONENTS (PCS) OF PCA PROJECTIONS AND THE BACKBONE TO EXTRACT THE FEATURES FOR EACH LEARNING CONFIGURATION.

Learning conf.	Backbone	MeanSem features	VarSem features
raw	SqueezeNet	first 4 PCs	0 PC
lin	resnet101	first 6 PCs	first 2 PCs
exp	resnet101	first 8 PCs	first 2 PCs

TABLE II
FULL-REFERENCE AND REFERENCE-ONLY METRICS SCORES ON DATASET TEST SET. * INDICATE PERFORMANCES OF RETRAINED METRICS.

Type	Metrics	PLCC	SRCC	KRCC	RMSE
Full-Reference IQA/VQA no semantic	PSNR$_{CB}$	0.472	0.594	0.428	0.535
	PSNR$_{CR}$	0.447	0.539	0.376	0.539
	PSNR$_{Y}$	0.517	0.685	0.507	0.526
	SSIM [4]	0.629	0.763	0.586	0.481
	VIF [24]	0.693	0.780	0.603	0.431
	DLM [25]	0.846	0.869	0.696	0.321
	VMAF [8]	0.833	0.867	0.694	0.335
	VMAF*	***0.875***	***0.900***	***0.747***	***0.291***
DL Full-Reference IQA semantic	LPIPS-vgg [1]	0.711	0.795	0.631	0.420
	LPIPS-squeeze	0.674	0.785	0.622	0.445
	LPIPS-alex	0.628	0.754	0.588	0.470
	DISTS [3]	0.787	0.851	0.671	0.369
Reference-Only no semantic	WPSNR [5]	0.618	0.819	0.642	0.483
	XPSNR [6]	0.665	0.828	0.652	0.461
	libaom tune=ssim	0.653	0.795	0.614	0.476
DL Reference-Only VQA	our model (raw)	**0.844**	0.878	0.714	0.336
	our model (lin)	**0.843**	**0.888**	**0.721**	0.328
	our model (exp)	**0.852**	**0.888**	**0.728**	**0.316**

here to regress the two factors (A, B). These two factors scale MSE_Y values to the perceptual continuum.

During each configuration training, we perform a Grid Search selection over (1) the number of PCA components inputted to the SVR model, (2) the Neural Network employed to extract the deep features, and (3) SVR hyperparameters. In table I, we reported the best combinations of features.

We conducted the Grid Search on the dataset train set with a 25-fold cross-validation. We reported performances on the test set only. The best-performing set of hyper-parameters for each learning configuration is reported in table I. For example, the best combination of attributes for the ***lin*** configuration is based on features extracted from RestNet101 backbone, using the six Principal Components of $MeanSem$ features vector and the two Principal Components of $VarSem$.

IV. RESULTS

In this section, we present the results. We consider metrics performances in terms of Pearson correlation coefficient (PLCC), Spearman correlation (SRCC), Kendall tau correlation (KRCC), and Root Mean Squared Error (RMSE). To report existing objective metrics performances, we fit a 4-parameter cubic polynomial to map the objective scores to the subjective scores. This fitting affects PLCC and RMSE scores and allows to compensate for range gaps and nonlinear relationships. To be fair to trained and retrained metrics, fitting coefficients are optimized on the train set and reported on the test set.

First, we evaluate the performances of Full-Reference image and video quality metrics in table II. These traditional quality metrics extract statistics from both reference and distorted frames. Moreover, these statistics contain no semantic information. We can see that VMAF and DLM have the highest correlation with the subjective scores. With VMAF retrained on our dataset, the performances are improving as the best performances overall.

Second, we evaluated the performances of Deep Learning Full-Reference Image Quality Assessment metrics. These metrics perform better than PSNR, SSIM, or VIF but remain lower than VMAF and DLM. Retraining of LPIPS and DISTS on our dataset was unsuccessful due to its limited training data and large sets of weights in these metrics.

Third, we explored the performances of Reference-Only metrics. We can see that their performances are lower than VMAF and DLM but on par with other IQA/VQA metrics and Deep learning Full-Reference quality metrics. We also included results of the metric performing bit allocation in libaom available under the *tune=ssim* flag. This metric is not the exact implementation of SSIM. This version only uses the pixel variance intra-block $block_{var}$ in reference frames to derive scaling factors for computed MSE distortions between reference and coded blocks. The equation of this block-variance scaling is the following:

$$ssim_{var} = 67.0354 \times (1 - e^{-0.00214 \times block_{var}}) + 17.4922 \quad (3)$$

Finally, our models achieve the second-best overall performances for SRCC and KRCC indicators after the VMAF retrained version. Regarding PLCC and RMSE, performances are similar to VMAF and DLM but significantly improve over WPSNR, XPSNSR, and *tune=ssim*. Our model fpr ***lin*** and ***exp*** training configurations outperforms ***raw*** training configuration, hinting at the gain brought by added prior knowledge of the shape of the *PD-curve*.

V. CONCLUSION

This work presents a new model to predict the perception of local distortions over *tubes*. The model relies on Deep Learning features extracted only from reference frames. We pool these features spatially and temporally to represent the semantic information along reference tubes. With PCA projections to better represent essential features, we reduced the complexity of this semantic latent space and eased the training while avoiding overfitting. Relying on the prediction of *PD-curves* slopes also improved performances.

The proposed method outperforms other methods based on reference frame-only pixel statistics. These metrics are still the primarily used objective metrics for optimizing video encoding algorithms. Our method could improve bit allocation strategies by replacing these metrics in future research work.

Compared to Full-Reference metrics, the proposed methods outperform existing Deep Learning Image quality metrics by efficiently learning from limited data. Evaluating the metric on high-resolution video remains a subject for future research.

REFERENCES

[1] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," *CoRR*, vol. abs/1801.03924, 2018. [Online]. Available: http://arxiv.org/abs/1801.03924

[2] E. Prashnani, H. Cai, Y. Mostofi, and P. Sen, "Pieapp: Perceptual image-error assessment through pairwise preference," *CoRR*, vol. abs/1806.02067, 2018. [Online]. Available: http://arxiv.org/abs/1806.02067

[3] K. Ding, K. Ma, S. Wang, and E. P. Simoncelli, "Image quality assessment: Unifying structure and texture similarity," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 5, pp. 2567–2581, 2020.

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

[5] J. Erfurt, C. R. Helmrich, S. Bosse, H. Schwarz, D. Marpe, and T. Wiegand, "A study of the perceptually weighted peak signal-to-noise ratio (wpsnr) for image compression," in *2019 IEEE International Conference on Image Processing (ICIP)*, 2019, pp. 2339–2343.

[6] C. R. Helmrich, M. Siekmann, S. Becker, S. Bosse, D. Marpe, and T. Wiegand, "Xpsnr: A low-complexity extension of the perceptually weighted peak signal-to-noise ratio for high-resolution video quality assessment," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 2727–2731.

[7] A. Pastor, L. Krasula, X. Zhu, Z. Li, and P. L. Callet, "On the accuracy of open video quality metrics for local decision in av1 video codec," in *2022 IEEE International Conference on Image Processing (ICIP)*, 2022, pp. 4013–4017.

[8] Netflix, "Vmaf v0.6.1 model," https://github.com/Netflix/vmaf.

[9] H. Wang, I. Katsavounidis, J. Zhou, J. Park, S. Lei, X. Zhou, M.-O. Pun, X. Jin, R. Wang, X. Wang, Y. Zhang, J. Huang, S. Kwong, and C.-C. J. Kuo, "VideoSet: A large-scale compressed video quality dataset based on JND measurement," *Journal of Visual Communication and Image Representation*, vol. 46, pp. 292–302, Jul. 2017. [Online]. Available: https://doi.org/10.1016/j.jvcir.2017.04.009

[10] K. Knoblauch, L. T. Maloney *et al.*, "Mlds: Maximum likelihood difference scaling in r," *Journal of Statistical Software*, vol. 25, no. 2, pp. 1–26, 2008.

[11] A. Pastor, L. Krasula, X. Zhu, Z. Li, and P. Le Callet, "Improving maximum likelihood difference scaling method to measure inter content scale," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 2045–2049.

[12] A. Pastor and P. L. Callet, "Perception of video quality at a local spatio-temporal horizon: Research proposal," in *Proceedings of the 13th ACM Multimedia Systems Conference*, ser. MMSys '22. New York, NY, USA: Association for Computing Machinery, 2022, p. 378–382. [Online]. Available: https://doi.org/10.1145/3524273.3533931

[13] A. Pastor and P. Le Callet, "Perceptual annotation of local distortions in videos: Tools and datasets," in *Proceedings of the 14th Conference on ACM Multimedia Systems*, ser. MMSys '23. New York, NY, USA: Association for Computing Machinery, 2023, p. 458–462. [Online]. Available: https://doi.org/10.1145/3587819.3592559

[14] A. Pastor, L. Krasula, X. Zhu, Z. Li, and P. L. Callet, "Recovering quality scores in noisy pairwise subjective experiments using negative log-likelihood," in *2023 IEEE International Conference on Image Processing (ICIP)*, 2023, pp. 2635–2639.

[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[17] 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.

[18] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning*. PMLR, 2019, pp. 6105–6114.

[19] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and¡ 0.5 mb model size," *arXiv preprint arXiv:1602.07360*, 2016.

[20] N. Ma, X. Zhang, H.-T. Zheng, and J. Sun, "Shufflenet v2: Practical guidelines for efficient cnn architecture design," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 116–131.

[21] A. Howard, M. Sandler, G. Chu, L.-C. Chen, B. Chen, M. Tan, W. Wang, Y. Zhu, R. Pang, V. Vasudevan *et al.*, "Searching for mobilenetv3," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 1314–1324.

[22] 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, 2018.

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

[24] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Transactions on image processing*, vol. 15, no. 2, pp. 430–444, 2006.

[25] S. Li, F. Zhang, L. Ma, and K. N. Ngan, "Image quality assessment by separately evaluating detail losses and additive impairments," *IEEE Transactions on Multimedia*, vol. 13, no. 5, pp. 935–949, 2011.