A study on the impact of training data in CNN-based super-resolution for low bitrate end-to-end video coding

Fatemeh Nasiri, Wassim Hamidouche, Luce Morin, Gildas Cocherel, Nicolas Dhollande

A STUDY ON THE IMPACT OF TRAINING DATA IN CNN-BASED SUPER-RESOLUTION FOR LOW BITRATE END-TO-END VIDEO CODING

Fatemeh Nasiri [⋆†+], *Wassim Hamidouche*[⋆†], *Luce Morin*[†⋆], *Gildas Cocherel*[+], *Nicolas Dhollande*[+]

[⋆] IRT b<>com, 35510 Cesson-Sévigné, France,
[†] Univ Rennes, INSA Rennes, CNRS, IETR - UMR 6164, 35000 Rennes, France
[+] AVIWEST, 35760, Saint-Gregoire, France

ABSTRACT

In this study, the effectiveness of Super Resolution (SR) methods based on Convolutional Neural Network (CNN) in low bitrate video coding, with a focus on the Versatile Video Coding Standard (VVC), is investigated. Video transmission over networks with limited bandwidth is a common challenge for different applications. One solution is to adopt SR methods where the main principle is to spatially down-sample the input sequence prior to the encoding, then up-sampling the decoded sequence before displaying it. For a fixed target bandwidth, a finer quantization is applied on the low-resolution sequence compared to high-resolution, so that the high quality reconstructed pixels help in retrieving the lost information. However, most CNN-based SR methods are designed for single images and merely focus on the original input signal. Therefore, their trained networks lack understanding of compression artifacts. In this study, we test a hypothesis that training CNN-based SR methods with compressed sequences outperforms training with uncompressed ones. The assumption is that such training allows the SR methods to learn compression artifacts and differentiate them from actual texture information. To this end, state-of-the-art CNN-based SR methods are tested with compressed and uncompressed training set. Experiments show that the use of compressed training data brings, on average, an additional bitrate saving of 6%, in terms of BD-Rate.

Index Terms— Convolutional Neural Networks, Super Resolution, Low Bitrate Video Coding

1. INTRODUCTION

With evergrowing applications of video transmission, the task of retaining a high quality displayed video under the network limitations has recently become more trendy. On the one hand, compressed videos are sent under stricter bandwidth constraints which limits the amount of transmitted information and makes the compression artifacts such as blurriness or blockiness more evident in received videos. On the other hand, receiver devices are usually powerful enough to afford complex post-processing steps to perform texture restoration. Therefore, there is a chance to achieve video quality levels that are currently inaccessible in specific low bitrate applications, such as telesurgery, monitoring systems [1, 2], *etc.*

A key to this goal is to adopt artificial intelligence techniques to learn compression loss patterns. In particular, the Convolutional Neural Network (CNN) based Super Resolution (SR) algorithms properly fit the requirements of the compression artifact restoration task [3]. The SR algorithms aim at generating a high-resolution signal from a given low-resolution one. In its basic form, the large amount of missing information in the low-resolution images makes generating the high-resolution image challenging. However, a variety of advanced SR methods are proposed to overcome this problem [4–8]. Particularly, CNN-based algorithms have shown impressive performance compared to traditional methods [9–12].

Despite the potentials, the use of SR methods on compressed videos is sparsely studied in the literature [13–15]. Particularly that, recently, video transmission methods involving sub-sampling input signal have become very popular. For instance, a standardization activity is currently ongoing to release a video codec, called Low Complexity Essential Video Coding (LCEVC), which addresses the same issue by down-sampling, coding and transmitting meta-data [16]. Also, AV1 codec also adopts resolution adaptation at both encoder and decoder with pre-defined up-sampling filters [17]. Finally, the concept of Reference Picture Resampling (RPR), which has recently been adopted for VVC, benefits from a similar methodology [18].

In this study, first a general framework for integrating SR methods within a coding system is described. Then, for a set of selected SR methods, the impact of training with compressed dataset is compared. In particular, the objective of this paper is to demonstrate how differently CNN-based SR methods perform on reconstructed video signals when they are trained with compressed or uncompressed datasets. The importance of this work is perceiving the image super resolution subject rather from a video coding point of view for exploring its potentials.

The rest of this paper is organized as follows. In Section 2, an SR-based video transmission framework is described to integrate the selected SR methods along with the VVC codec. Section 3 describes the characteristics of the experiments. Section 4 presents the details of performance evaluation with discussions and finally, Section 5 concludes this paper.

2. SR-BASED VIDEO CODING FRAMEWORK

A general framework for low bitrate video transmission using SR is described. This configurable framework is used by some broadcasters to adapt their content to low bandwidth and/or low complexity constraints. There are three main steps in the SR framework: first, the input sequence is down-sampled prior to the coding. Second, the down-sampled sequence is coded with VVC, using an adjusted Quantization Parameter (QP). Third, an up-sampling step is performed on the reconstructed image using an SR method. Once all the above three steps are performed, the output is comparable to the regular coding scheme, where no down/up-sampling steps are used. Fig. 1 compares the two coding schemes. In addition to possible coding efficiency gains in the low bitrate range, experiments of this study show that the use of the SR framework saves between 40% to 80% encoding time.

2.1. QP adjustment with respect to down-sampling factor

The Quantization Parameter (QP) is a mean to apply user-specified level of distortion to the compression and has two main functionalities in a codec: 1) determining the quantization step size of residual coefficients, and 2) making a trade-off between rate and distortion of different coding decisions. The combination of these two roles guarantees that under a given rate constraint, the distortion

978-1-7281-8750-1/20/$31.00

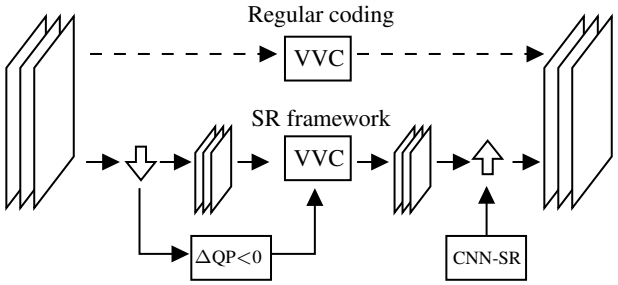

Fig. 1: Comparison of the regular coding scheme and CNN-based SR framework

per pixel will remain below a threshold [19]. The fact that the SR framework reduces the resolution of the input sequence should not impact this distortion. Therefore, to accommodate a lower resolution, a QP adjustment parameter $\Delta QP<0$ is added to the input QP value to apply a finer quantization. The principle of computing ΔQP as a function of the scale factor, described in [20], is adopted in the current work. Based on this method, a QP adjustment of ΔQP = -6 is applied for the used scale factor 2 on width and height.

2.2. Scope of the performance

The SR framework of Fig. 1 and its internal modules are flexible in terms of functionality. More precisely, one can adjust the following settings depending on target application:

- Scale factor
- SR method for up-sampling
- Training dataset in case of CNN-based SR

The use of the SR framework becomes justifiable when properly tuned. For more efficient deployment, one should first understand when this framework can be beneficial. In terms of rate-distortion-complexity measurement, the SR framework can potentially have the following impacts:

- Rate:
  - Rate per sample may increase, since the QP adjustment causes finer residual quantization.
  - Rate per frame may decrease, since the down-sampling step reduces the number of coded samples in each frame.
- Distortion:
  - Distortion per frame may increase, since the down-sampling step throws away majority of samples, and this information loss may not be fully retrieved by the up-sampling step.
  - Distortion per frame may decrease, since the up-sampling step, in particular the CNN-based ones, are supposedly smart and able to retrieve a high amount of the lost information.
  - Distortion per sample may decrease, since a finer quantizer is applied on the down-sampled input.
- Complexity:
  - May decrease at the encoder side, since the number of coded samples is reduced due to the down-sampling.
  - May increase at the decoder side, since the up sampling modules are added.

The overall trade-off between all above impacts determines whether or not a specific setting of the SR framework provides desired bandwidth saving and/or complexity reduction over the regular coding scheme. In other words, the use of the SR framework is preferable when the combination of the rate-distortion-complexity results in a better global performance.

2.3. Problem statement

In this paper, we investigate the impact of alternative training for SR methods. More precisely, we test a hypothesis: given that a CNN-based SR method is to be used for up-sampling the decoded sequences, involving compression artifacts in the training process of the SR method will improve its up-sampling performance.

The assumption is that observing compression artifacts during the training phase helps SR methods differentiate those artifacts from actual texture information during inference. Fig. 2 presents examples of actual texture information and compression artifacts.

Fig. 2: Compression artifacts (*e.g.* blockiness, blurriness) in textures coded at very low bitrate (*i.e.* 100-500kbps for 1920×1080p sequence).

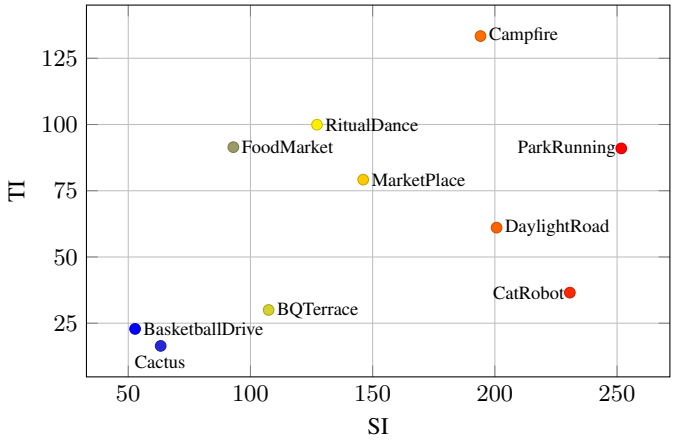

Fig. 3: Spatial Index (SI) and Temporal Index (TI) of test sequences.

2.4. Selected CNN-based SR methods

Two CNN-based SR methods are selected. These methods represent relatively simple and complex CNN architectures for SR. It is important to note that the selected methods are not supposed to be compared to each other in terms of performance. Instead, the goal is to compare each method with itself, under different training conditions.

Efficient Sub-Pel Convolutional Neural Network (ESPCN): ESPCN is composed of three CNN layers [10]. The two first layers are used for the feature maps extraction and the last layer, which is a sub-pixel convolutional layer, is responsible for aggregating the feature maps from low-resolution space and constructing the high-resolution image. Using the sub-pixel convolutional layer, as an up-sampler in the last layer, decreases the computational time and increases the network flexibility in learning different down-sampling kernels. For the

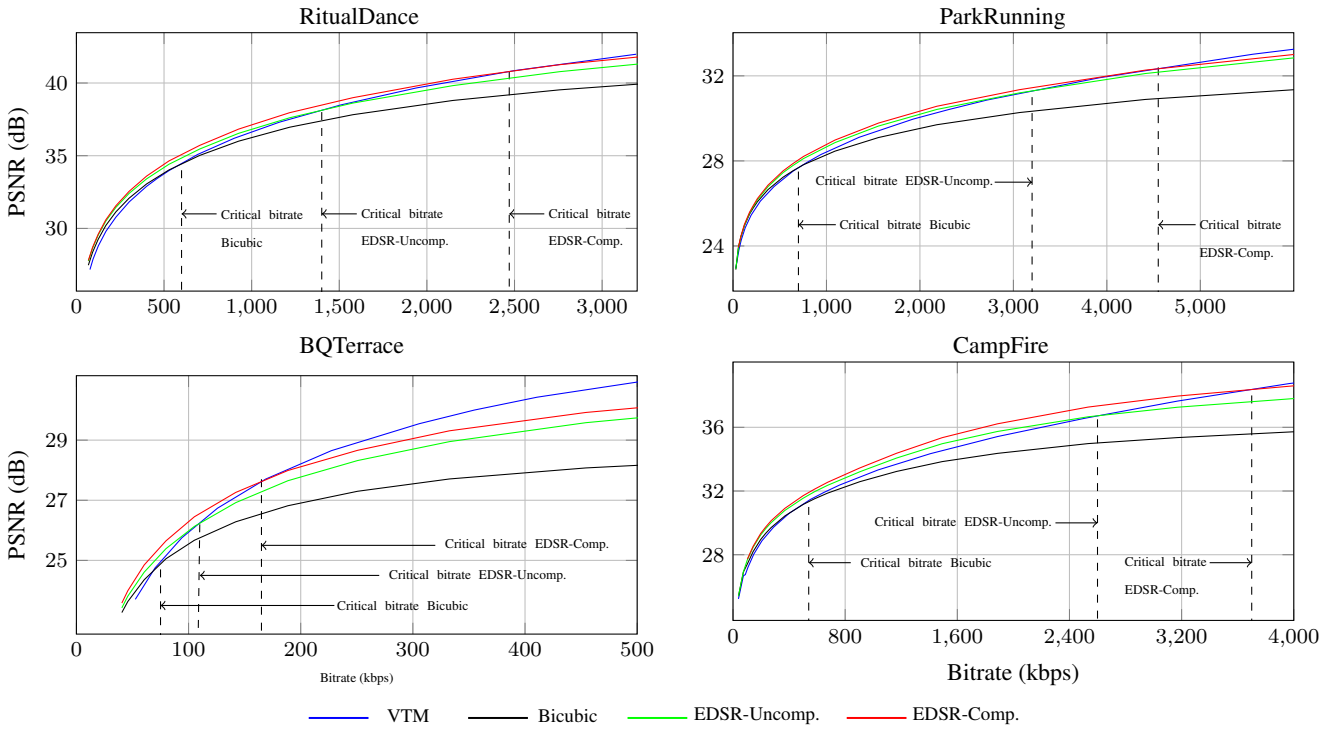

Fig. 4: Rate-PSNR curves corresponding to different coding schemes: EDSR in the uncompressed and compressed settings, the bicubic and the VTM. In order to clarify the improvement due to the use of compressed training set, the critical bitrates with respect to the VTM are shown with dashed lines.

training, ESPCN uses L2 loss which maximizes PSNR. The design of this method is considered as a relatively simple network architecture.

Enhanced Deep Super Resolution Network (EDSR): EDSR uses a residual network architecture for solving the SR problem [11]. In this network, the original architecture of the ResNet [21,22] has been modified to increase the performance for this specific task. The modifications make the network lighter to be trained and to capture the proper features for constructing the best super resolved image. Experimental results show that EDSR outperforms most state-of-the-art SR networks. Moreover, the use of residual-based building blocks enables EDSR to learn missing high frequency information in different scaling factors. EDSR uses L1 loss for training, which gives better convergence than L2 loss. Compared to the ESPCN network, the architecture of the EDSR network is noticeably more complex.

3. EXPERIMENTS DESCRIPTION

3.1. SR framework setting

Test-train sequences: The experiments of this paper are focused on full High Definition (HD) video sequences with sample resolution of 1920×1080. As training sequence set, we used the DIV2K [23] and Flickr2K datasets. The test sequence set is composed of 10 sequences from the Common Test Condition (CTC) of JVET and JCT-VC [24]. In order to further extend the list of test sequences, five Ultra HD (UHD) sequences from the CTC are also down-sampled into HD and used. Fig. 3 quantifies the motion and texture characteristics of these test sequences. For this purpose, Spatial Index (SI) and Temporal Index (TI) are used [25], where higher values indicates more complex texture and motion characteristics, respectively.

Down-sampling: The down-sampling step of the SR framework is shared between the training and test phases. In both cases, the bicubic filter, implemented in FFMPEG, has been used. As we only focus on 1920× 1080 resolution in the conducted experiments, only the scaling factor of 2 is used for down-sampling.

Coding schemes: Four coding schemes have been compared. The two CNN-based SR methods presented in Section 2.4; the non-CNN bicubic SR scheme; and the VVC test Model (VTM) coding scheme in the context of regular coding of Fig. 1. The two latter schemes, the bicubic and VTM, are served as anchors for the former CNN-based SR methods.

3.2. Training of CNN-based SR methods

Ground truth: In the SR framework, the ultimate goal is to be as similar as possible to the original high-resolution sequence. Therefore, this signal is used as the ground-truth for both training settings.

Down-sampled dataset: Two down-sampled training datasets are used: uncompressed and compressed. In the uncompressed setting, the training data is simply provided by down-sampling the original sequences. While, in the compressed setting, the same down-sampled sequences are compressed prior to the training.

Coding artifacts: To produce the training dataset of the compressed setting, the VVC Test Model (VTM-5) has been used. It was assessed that the QP-independent training of the CNN-based SR methods results in a poor performance. This is due to the fact that coding artifacts have various characteristics in different bitrates. Therefore, we divided the QP range of 22-63 into 6 equal intervals, as a compromise between the performance and number of networks. For each interval, the QP value in the middle has been selected to learn the compression artifacts of that interval. This design choice results in six trained networks for each CNN-based SR scheme.

4. RESULTS

4.1. Coding efficiency performance

Table 1 presents the performance of various coding schemes and their different settings. In order to conduct performance comparisons, two metrics are used:

Table 1: Performance of ESPCN and EDSR methods trained with compressed and uncompressed datasets. Bitrate saving values of the compressed setting, presented in terms of BD-Rate (%), are calculated against the bicubic SR method and the uncompressed setting. Moreover, the critical bitrates of compressed and uncompressed settings are computed against the VTM and presented in terms of "kbps".

Sequence	CNN-based SR method									
	ESPCN					EDSR				
	Uncompressed		Compressed		BD-Rate	Uncompressed		Compressed		BD-Rate
	BD-Rate vs. Bicubic	Critical bitrate	BD-Rate vs. Bicubic	Critical bitrate	Compr. vs. Uncompr.	BD-Rate vs. Bicubic	Critical bitrate	BD-Rate vs. Bicubic	Critical bitrate	Compr. vs. Uncompr.
BasketballDrive	-43%	150 kbps	-46%	175 kbps	-3%	-50%	240 kbps	-59%	380 kbps	-9%
BQTerrace	-19%	90 kbps	-24%	102 kbps	-5%	-26%	110 kbps	-33%	165 kbps	-7%
Cactus	-11%	180 kbps	-14%	200 kbps	-3%	-14%	180 kbps	-22%	320 kbps	-8%
CampFire	-2%	490 kbps	-5%	630 kbps	-3%	-19%	2650 kbps	-23%	3700 kbps	-4%
CatRobot	-8%	170 kbps	-11%	190 kbps	-3%	-16%	200 kbps	-24%	455 kbps	-8%
DayLight	-5%	240 kbps	-8%	290 kbps	-3%	-9%	260 kbps	-15%	371 kbps	-6%
FoodMarket	+6%	570 kbps	+4%	730 kbps	-2%	-2%	1850 kbps	-6%	2000 kbps	-4%
MarketPlace	0%	350 kbps	-3%	430 kbps	-3%	-7%	540 kbps	-12%	963 kbps	-5%
ParkRunning	-5%	1000 kbps	-8%	1250 kbps	-3%	-12%	3200 kbps	-16%	4550 kbps	-4%
RitualDance	+2%	450 kbps	-1%	600 kbps	-3%	-12%	1400 kbps	-17%	2471 kbps	-5%
Average	**-8.5%**	**293 kbps**	**-11.6%**	**460 kbps**	**-3.1%**	**-16.7%**	**1063 kbps**	**-22.7%**	**1538 kbps**	**-6%**

1) Bjøntegaard Delta Rate (BD-Rate): This metric is computed between different SR methods. The negative BD-Rate value is interpreted as the percentage of bitrate saving in the same level of quality based on Peak Signal-to-Noise Ratio (PSNR) [26].

2) Critical bitrate: This term is used to denote the maximum bitrate of a sequence where the use of the SR framework still outperforms the regular coding [27]. Obviously, the larger values of critical bitrate indicate that the use of the SR framework can be justified in a wider range of applications. For a better comparison, Fig. 4 shows the Rate-PSNR curves of the EDSR method for a selection of test sequences, with their critical bitrate.

4.2. Observations and discussions

Training set: With no exception, the use of compressed training set outperforms the uncompressed one. This is reflected in three aspects. First, there are coherent BD-Rate gains with the compressed setting compared to the uncompressed setting, which are -3.1% and -6% for the ESPCN and EDSR methods, respectively. Second, the critical bitrate of the SR framework significantly moves towards the higher bitrates, when the compressed setting is used instead. Third, it was assessed that the amount of BD-Rate gain due to the compressed setting seems to be consistent and content-independent.

The QP-dependent network training in the compressed setting is critical. As mentioned earlier, the preliminary experiments of this study showed that when the compressed setting was trained with a dataset composed of all range of QPs, the results are significantly worse than the QP specific compressed setting. This means that the statistics of coding artifacts vary in different ranges of low bitrate. Therefore, in order to be able to restore low bitrate artifacts, one should expose the CNN learning to the appropriate training samples, representing the right type of artifacts. In conclusion, all these evidences show that the proper use of compressed training set significantly improves the performance of the SR framework.

The SR framework: The performance of the SR framework degrades in higher bitrates. The critical bitrates metrics of Fig. 4 properly demonstrate this fact. This figure shows that after certain bitrate, the SR framework becomes significantly poorer than the regular coding with VTM. One possible reason for this behavior is the nature of artifacts that are specific for very low bitrates, (*e.g.* blockiness and blurriness). More precisely, restoring these artifacts might be suitable for neural-network based solutions, while avoiding them at these bitrates is very difficult for the VTM. It is also asserted that the performance of the SR framework is highly content-dependent. As can be seen in Table 1, in some sequences such as ParkRunning, CampFire and RitualDance, the use of compressed train data moves the critical bitrate about 1Mbps. According to Fig. 3, all these sequences have relatively complex spatial and temporal characteristics.

SR methods: The BD-Rate improvement of using compressed training set with EDSR is significantly larger than that of ESPCN. As mentioned earlier, the ESPCN architecture is relatively simpler than EDSR. Therefore, this result loosely concludes that simple network structures might not be powerful enough to differentiate between compression artifacts and actual texture information during training. Testing this hypothesis with more network examples is left as future work. Another observation is that the sequences with the highest improvement due to the use of compressed dataset, are BQTerrace, Cactus and CatRobot. According to Fig. 3, all these sequences have relatively low temporal complexity. Interestingly, in all three sequences, the performance of the SR framework against the VTM anchor is among the poorest ones. The interpretation can be that when the SR framework performs poorly compared to the VTM, the use of compressed dataset can make a bigger change. Finally, the results show that the CNN-based SR methods do not necessarily perform better than simple SR methods, in all sequences. Examples like FoodMarket, RitualDance and MarketPlace, where the bicubic method outperforms ESPCN, prove that a bad choice of CNN-based SR method can easily deteriorate the SR framework.

5. CONCLUSION

In this study, the impact of adding compressed videos to the training set for CNN-based SR methods has been investigated. A coding framework is introduced in which different SR methods can serve for up-sampling. It was assessed that training CNN-based SR methods compressed training set significantly outperforms uncompressed training sets. This impact improves the global coding efficiency of the SR framework and justifies its use in a wider range of bitrates. Furthermore, it was observed that to boost the performance increase of using compressed training set, complex network architectures are preferred over simple ones, since they are more capable of learning common coding artifacts in low bitrates.

6. REFERENCES

[1] Lucie Lévêque, Wei Zhang, Christine Cavaro-Ménard, Patrick Le Callet, and Hantao Liu, "Study of video quality assessment for telesurgery," *IEEE Access*, vol. 5, pp. 9990–9999, 2017.

[2] Iulia Mitrica, Eric Mercier, Christophe Ruellan, Attilio Fiandrotti, Marco Cagnazzo, and Béatrice Pesquet-Popescu, "Very low bitrate semantic compression of airplane cockpit screen content," *IEEE Transactions on Multimedia*, vol. 21, no. 9, pp. 2157–2170, 2019.

[3] Sung Cheol Park, Min Kyu Park, and Moon Gi Kang, "Super-resolution image reconstruction: a technical overview," *IEEE signal processing magazine*, vol. 20, no. 3, pp. 21–36, 2003.

[4] Shengyang Dai, Mei Han, Wei Xu, Ying Wu, Yihong Gong, and Aggelos K Katsaggelos, "Softcuts: a soft edge smoothness prior for color image super-resolution," *IEEE Transactions on Image Processing*, vol. 18, no. 5, pp. 969–981, 2009.

[5] Jian Sun, Zongben Xu, and Heung-Yeung Shum, "Image super-resolution using gradient profile prior," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–8.

[6] William T Freeman, Thouis R Jones, and Egon C Pasztor, "Example-based super-resolution," *IEEE Computer graphics and Applications*, , no. 2, pp. 56–65, 2002.

[7] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma, "Image super-resolution via sparse representation," *IEEE transactions on image processing*, vol. 19, no. 11, pp. 2861–2873, 2010.

[8] Roman Zeyde, Michael Elad, and Matan Protter, "On single image scale-up using sparse-representations," in *International conference on curves and surfaces*. Springer, 2010, pp. 711–730.

[9] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang, "Image super-resolution using deep convolutional networks," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 2, pp. 295–307, 2015.

[10] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 1874–1883.

[11] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, 2017, pp. 136–144.

[12] Zhen Li, Jinglei Yang, Zheng Liu, Xiaomin Yang, Gwanggil Jeon, and Wei Wu, "Feedback network for image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3867–3876.

[13] Armin Kappeler, Seunghwan Yoo, Qiqin Dai, and Aggelos K Katsaggelos, "Super-resolution of compressed videos using convolutional neural networks," in *2016 IEEE International Conference on Image Processing (ICIP)*. IEEE, 2016, pp. 1150–1154.

[14] Bahadir K Gunturk, Yucel Altunbasak, and Russell M Mersereau, "Super-resolution reconstruction of compressed video using transform-domain statistics," *IEEE Transactions on Image Processing*, vol. 13, no. 1, pp. 33–43, 2004.

[15] Hongwei Lin, Xiaohai He, Linbo Qing, Qizhi Teng, and Songfan Yang, "Improved low-bitrate HEVC video coding using deep learning based super-resolution and adaptive block patching," *IEEE Transactions on Multimedia*, 2019.

[16] B. Allan et al., "Text of ISO/IEC CD 23094-2, low complexity enhancement video coding (LCEVC)," *Motion Picture Expert Group (MPEG) meeting 129*, 2019.

[17] Urvang Joshi, Debargha Mukherjee, Yue Chen, Sarah Parker, and Adrian Grange, "In-loop frame super-resolution in av1," in *2019 Picture Coding Symposium (PCS)*. IEEE, 2019, pp. 1–5.

[18] Benjamin Bross and Ye-Kui Wang Jianle Chen, Shan Liu, "Versatile video coding draft text," *JVET-Q2001*, 2020.

[19] Gary J Sullivan and Thomas Wiegand, "Rate-distortion optimization for video compression," *IEEE signal processing magazine*, vol. 15, no. 6, pp. 74–90, 1998.

[20] Yue Li, Dong Liu, Houqiang Li, Li Li, Feng Wu, Hong Zhang, and Haitao Yang, "Convolutional neural network-based block up-sampling for intra frame coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 9, pp. 2316–2330, 2018.

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[22] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al., "Photo-realistic single image super-resolution using a generative adversarial network," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 4681–4690.

[23] Eirikur Agustsson and Radu Timofte, "Ntire 2017 challenge on single image super-resolution: Dataset and study," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2017, pp. 126–135.

[24] Karsten Suehring and Xiang Li, "JVET common test conditions and software reference configurations," *JVET-B1010*, 2016.

[25] TU-T P.910, "Subjective video quality assessment methods for multi- media applications,," *ITU Telecom. Standardization Sector of ITU*, April 2008.

[26] Gisle Bjontegaard, "Calculation of average PSNR differences between RD-curves," *VCEG-M33*, 2001.

[27] Weisi Lin and Li Dong, "Adaptive downsampling to improve image compression at low bit rates," *IEEE Transactions on Image Processing*, vol. 15, no. 9, pp. 2513–2521, 2006.