HAL open science

Predictive Coding for Animation-Based Video Compression

Goluck Konuko, Stéphane Lathuilière, Giuseppe Valenzise

To cite this version:

Goluck Konuko, Stéphane Lathuilière, Giuseppe Valenzise. Predictive Coding for Animation-Based Video Compression. International Conference on Image Processing (ICIP'2023), Oct 2023, Kuala Lumpur (Malaysia), Malaysia. 10.1109/icip49359.2023.10222205 . hal-04243836

HAL Id: hal-04243836

https://hal.science/hal-04243836v1

Submitted on 16 Oct 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

PREDICTIVE CODING FOR ANIMATION-BASED VIDEO COMPRESSION

Goluck Konuko[†], *Stéphane Lathuilière*[‡], *Giuseppe Valenzise*[†]

[†] Université Paris-Saclay, CentraleSupélec, Laboratoire des signaux et systèmes
[‡] LTCI, Télécom Paris, Institut Polytechnique de Paris, France

ABSTRACT

We address the problem of efficiently compressing video for conferencing-type applications. We build on recent approaches based on image animation, which can achieve good reconstruction quality at very low bitrate by representing face motions with a compact set of sparse keypoints. However, these methods encode video in a frame-by-frame fashion, i.e., each frame is reconstructed from a reference frame, which limits the reconstruction quality when the bandwidth is larger. Instead, we propose a predictive coding scheme which uses image animation as a predictor, and codes the residual with respect to the actual target frame. The residuals can be in turn coded in a predictive manner, thus removing efficiently temporal dependencies. Our experiments indicate a significant bitrate gain, in excess of 70% compared to the HEVC video standard and over 30% compared to VVC, on a dataset of talking-head videos. Our code is available at github.com/animation-based-codecs.

***Index Terms*—** Video compression, image animation, generative models, video conferencing, predictive coding

1. INTRODUCTION

Recent work on learning-based video coding for videoconferencing applications has shown that it is possible to compress videos of talking heads with extremely low bitrate, without significant losses in visual quality [1, 2, 3, 4, 5, 6]. The basic tenet of these methods is that face motion can be represented through a compact set of sparse keypoints [7], which can be transmitted and used at the decoder side to animate a reference video frame.

However, despite the impressive coding performance of these methods at very low bitrates, existing animation-based codecs for videoconferencing still have several bottlenecks. Firstly, when the available bitrate increases, the reconstruction quality quickly reaches saturation, and conventional coding tools such as HEVC or VVC perform better. Secondly, bitrate variability in current schemes is complex, unlike conventional coding methods where a simple quantization parameter can be used to regulate bitrate. Finally, animation-based codecs operate on a frame-by-frame basis, which is inefficient for eliminating temporal redundancy in the video.

This paper addresses these limitations by proposing a *predictive coding* scheme for videoconferencing applications. Specifically, we interpret the keypoint-based image animation used in previous codecs [1] as a *spatial predictor* of the current (target) frame, as depicted in Figure 1. The residual between the animated and the target frame is then coded and used at the decoder side to correct the animated target frame. Since animation residuals exhibit *temporal* correlation, we also encode them in a predictive manner, i.e., we predict the current animation residual based on the previously decoded residual and encode the prediction difference. It is worth noting that this approach is similar in principle to the classic video coding prediction loop, with the important distinction that residual coding and animation are *jointly* learned in an end-to-end fashion. We name our method **RDAC**, for Residual Deep Animation Codec. Our results demonstrate significant rate-distortion improvements compared to standard codecs such as HEVC and VVC, as measured by several classical and learning-based perceptual quality metrics. Furthermore, the proposed technique has the additional advantage of reducing temporal drift compared to previous frame-by-frame approaches.

2. RELATED WORK

Image animation models have been applied to compress talking head videos at ultra-low bitrates in conferencing-type applications [1, 2, 3, 4, 5, 6]. Different from other learning-based compression frameworks [8, 9, 10, 11, 12, 13, 14], the animation-based codecs in [3] and [4] propose architectures that use a variable number of motion keypoints to change the reconstruction quality within a small range of low bitrates. The deep animation codec (DAC) in our previous work [1] offers the possibility to vary the bitrate by creating a list of reference frames from which the best reconstruction is computed. Specifically, a new reference frame is added to the decoder buffer if all the available frames give reconstruction below a predefined threshold. However, this approach may introduce temporal jittering when adjacent animated frames are predicted from different reference frames. Using second-order motion coherence [6] introduces spatio-temporal stability in the decoded video, hence reducing the jittering. However, this architecture is still limited in terms of quality vari-

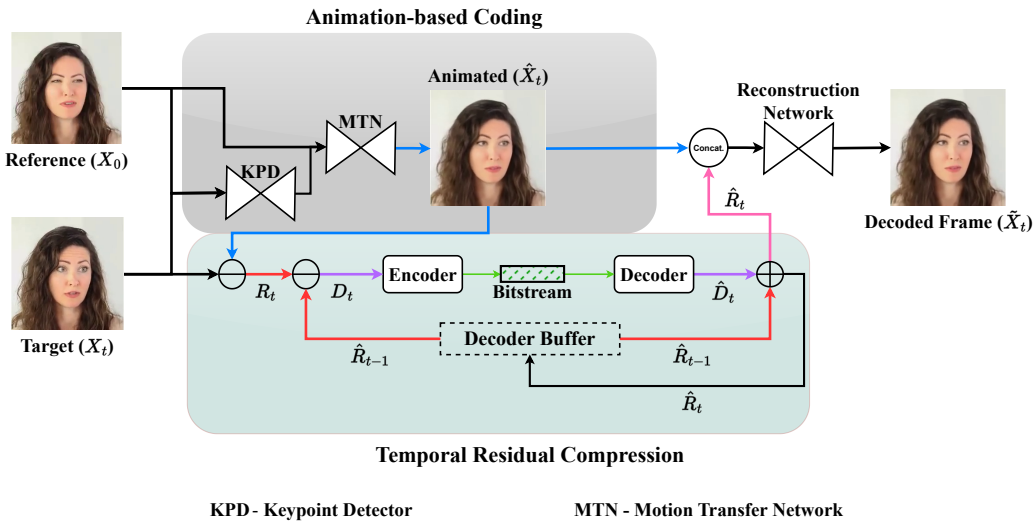

Fig. 1: **Proposed RDAC framework**. From a single reference frame, motion keypoints and a compact residual layer, our framework reconstructs a video with high perceptual and pixel fidelity. Note that when previously decoded frame information is available, it is used to improve the prediction accuracy of the next target residual through temporal correlation learning.

ability since it relies only on face animation. In our recent work [2], we proposed a hybrid coding architecture (HDAC) that uses a low-quality HEVC bitstream as side information to enhance the final result of the animation codec. While improving on previous methods, the use of this low-quality auxiliary stream limits in practice the possibility to reconstruct high-frequency details.

In this work, we propose a residual deep animation codec (RDAC) that learns a compact representation of the residual between a frame and its animation-based prediction, and encodes this residual using temporal prediction.

3. PROPOSED METHOD

A general scheme of the proposed residual deep animation codec is depicted in Fig. 1. The components of the proposed system are detailed as follows: Section 3.1 introduces the frame prediction and residual coding and Section 3.2 presents temporal learning in the residual space.

3.1. Deep Image Animation Prediction and Residual Coding

We leverage the principles developed in the First Order Model [7] for image animation and our prior works [1, 2] for animation-based prediction. The image animation process works by estimating a sparse set of motion landmarks using a keypoint detector (KPD) which is a UNet-like architecture from [7]. The keypoints are used by a motion transfer network (MTN) that generates the optical flow between a decoded reference image $\tilde{\mathbf{X}}_0$ and the desired target $\mathbf{X}_t$. Subsequently, the optical-flow map is applied to the feature space representation of the reference frame derived by the encoder of an autoencoder network. The deformed source features are assumed to be a close approximation of the target frame's feature representation and are used by a decoder network to produce the final animation $\hat{\mathbf{X}}_t$.

We build on this animation framework by including an encoder network that learns a latent representation of $\mathbf{R}_t = \mathbf{X}_t - \hat{\mathbf{X}}_t$ *i.e.* the residual after animation as illustrated in Fig. 1. We start with the architecture of the variational autoencoder network [9] used for learned image compression frameworks. However, since the residual images have very sparse features we mitigate the potential encoding of a noisy latent representation by increasing the number of downsampling convolutional layers from 3 to 5 and symmetrically increase the number of upsampling layers.

3.2. Using Temporal Correlation in the Residual Layer

For a sequence of target frames $\mathbf{X}_1 \rightarrow \mathbf{X}_T$ animated from a single reference frame, $\mathbf{X}_0$, we observe that the residual differences $\mathbf{R}_1 \rightarrow \mathbf{R}_T$ have a high temporal correlation. In this paper, we use a simple differential coding scheme to exploit this temporal correlation. Specifically, we compute the temporal difference signal between consecutive frame residuals, $\mathbf{D}_t = \mathbf{R}_t - \hat{\mathbf{R}}_{t-1}$, as shown in Fig. 1. Note that, in general, more sophisticated prediction schemes are possible, that could bring additional temporal decorrelation, e.g., any dense or block-based motion compensated scheme. In this work, we demonstrated coding gains even with a suboptimal zero-motion temporal predictor, leaving the study of more advanced prediction schemes to future work.

The difference signal $\mathbf{D}_t$ is coded using an additional

autoencoder network, which is trained together with the animation-based predictor and the reconstruction network. The decoding process consists in reconstructing the residual $\tilde{\mathbf{R}}_t = \tilde{\mathbf{D}}_t + \tilde{\mathbf{R}}_{t-1}$. The reconstructed residual is then concatenated to the animation-based predictor $\hat{\mathbf{X}}_t$ and passed as input to a reconstruction network that produces the final decoded frame $\tilde{\mathbf{X}}_t$. The reconstruction network consists of 2 convolution layers and 3 ResNet blocks.

3.3. Model Training

We initialize the animation module with pre-trained models from [1]. The loss terms for image animation are the same as in [1, 7], while the rate-distortion loss $\mathcal{L}_{RD}$ is derived as described in [9]:

$$\mathcal{L}_{RD} = \lambda \cdot \text{MSE}(\mathbf{R_t}, \hat{\mathbf{R}}_\mathbf{t}) + \text{Rate} \tag{1}$$

where the bitrate cost in bits-per-pixel (bpp) is computed from the entropy estimate of the residual latent representation.

4. EXPERIMENTS AND RESULTS

4.1. Evaluation Protocol

We randomly select 30 video sequences from the VoxCeleb test set with minimum lengths of 128 frames. We note that chaning the GOP size affects the average reconstruction quality of the video sequences. Therefore, we encode the sequences with GOP sizes 16, 32, 64, and 128 and select the best reconstruction point at each bitrate from a union of the computed metrics *i.e.* the convex hull of all the GOP configurations. The reference frame is encoded with QP 30 using the BPG codec (HEVC intra) and the motion keypoints as well as the compressed residuals are entropy coded using a context-adaptive arithmetic coder with a Prediction by Partial Match (PPM) model [15]. HEVC and VVC (VTM-11) metrics are computed under low-delay configurations with high QP values to minimize bitrate. We also compare against the LPIPS-VGG metrics reported for BeyondKP [5] and Face-Vid2Vid [3] since they use comparable test conditions. Notice that for these last two methods, we only have a single bitrate point, since they do not support bitrate variability beyond 10 kbps. MSE loss is used at training time for residual learning. However, the other loss terms used in training the network optimize for perceptual quality. Therefore, we restrict our evaluation to use only perceptual metrics and multi-scale pixel fidelity metrics.

4.2. RD Evaluation

In Tab. 1, we note over 70% bitrate savings for perceptual-based metrics *i.e.* LPIPS [16], msVGG [17] and DISTS [18] as well as over 40% bitrate savings for pixel-based metrics over HEVC. In Fig. 2 we make a visual comparison of our proposed framework with HEVC and VVC in the low bitrate range.

Table 1: **Bitrate savings (% BD-BR)** computed over 30 video sequences with 128 frames from VoxCeleb test set. The bitrate savings are measured with HDAC [2], HEVC and VVC codec as anchors

Metrics	*HDAC*	*VVC*	*HEVC*
msVGG	-55.10	-66.65	-74.99
DISTS	-48.46	-63.01	-82.62
LPIPS-VGG	-38.43	-48.73	-78.96
LPIPS	-6.45	-33.11	-75.96
FSIM	-42.89	-16.02	-63.10
MS-SSIM	-56.97	-20.11	-52.05

Fig. 3 illustrates the rate-distortion performance using the LPIPS metric. RDAC significantly improves performance of conventional video codecs over a wide range of bitrates, and it outperforms previous animation-based codecs which do not employ predictive coding.

4.3. Ablation study and temporal drift

An advantage of using a closed-loop prediction scheme for temporal coding of residuals is that it avoids the temporal drifting affecting previous open-loop schemes such as DAC. This is supported by Fig. 4, where we show the temporal reconstruction quality (measured with MS-SSIM) of our framework and DAC.

We also investigate to which extent the temporal prediction contributes to the RD gains, over a frame-by-frame scheme to code the prediction residuals $\mathbf{R}_t$. To this end, we remove the temporal feedback loop in Fig. 1, encoding the residuals as all Intra. Tab. 2 reports the gains of our proposed RDAC (with temporal prediction) over this simpler solution, demonstrating that reducing temporal correlation has a significant impact on coding performance.

Table 2: **Bitrate Savings (% BD-BR)** from our framework with temporal residual learning versus no temporal residual (10 sequences/64 frames)

DISTS	*LPIPS*	*msVGG*	*FSIM*	*MS-SSIM*
-9.35	-14.32	-5.65	-9.20	-6.86

4.4. Computational complexity

In Tab. 3, we make a complexity evaluation by comparing the coding or decoding time for a single interframe. The animation-based models DAC, HDAC, and our framework are evaluated on a CPU and GPU while the HEVC and VVC codecs are only evaluated on a CPU since they do not have

Fig. 2: **Visual Comparison of coding results.** A qualitative comparison of our proposed coding framework shows significant quality improvement over HEVC and VVC at low bitrates. We observe fewer smoothing and blocking artifacts as well as better color and style preservation in the reconstructed frames with our framework.

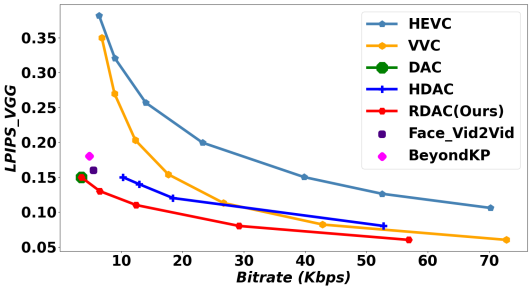

Fig. 3: **RD Performance:** Our method achieves considerable bitrate gains over a wider range of bitrates relative to both state-of-the-art video compression tools and the previously proposed frameworks.

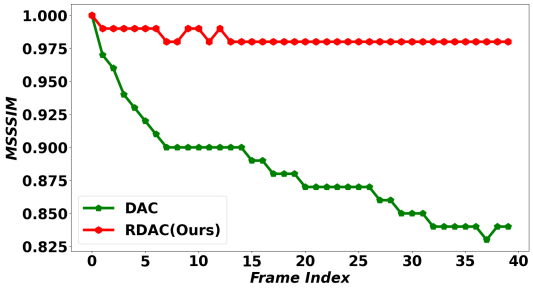

Fig. 4: **Reconstruction quality as a function of time:** RDAC temporal prediction avoids temporal drift, in contrast with open-loop schemes such as DAC, where quality degrades as the target frames get farther from the reference.

GPU acceleration capability. We note that our proposal adds only a moderate level of complexity relative to HEVC. However since we achieve bitrate savings greater than VVC, we consider this additional complexity as an acceptable tradeoff for the target application.

Table 3: **Computational Complexity:** Time to encode/decode 1 frame (in seconds). The computation time is estimated for the highest RD point of our framework.

	CPU (Intel Corei7)		**GPU (RTX 3090)**	
	Enc.	*Dec.*	*Enc.*	*Dec.*
HEVC	0.09	0.005	-	-
VVC	13.5	0.01	-	-
DAC	0.04	0.35	0.03	0.02
HDAC	0.10	0.35	0.12	0.02
RDAC (Ours)	0.52	0.50	0.21	0.02

5. CONCLUSIONS

Animation-based compression offers the possibility to transmit videos with very low bitrate. However, it is often limited to reconstructing the outputs at a fixed quality level, cannot scale efficiently when higher bandwidth is available, and does not compress efficiently temporal redundancies in the signal. In this paper, we propose a coding scheme that integrates image animation (re-interpreted as a frame predictor) with classical predictive coding principles, where we exploit both spatial and temporal dependencies to achieve a coding gain. Our RDAC codec outperforms previous methods and standard codecs by a large margin on a dataset of talking head videos, despite the very simple temporal prediction approach employed.

Acknowledgement: This work was funded by Labex DigiCosme - Université Paris-Saclay. This work was performed using HPC resources from GENCI-IDRIS

6. REFERENCES

[1] Goluck Konuko, Giuseppe Valenzise, and Stéphane Lathuilière, "Ultra-low bitrate video conferencing using deep image animation," in *IEEE ICASSP*, 2020.

[2] Goluck Konuko, Stéphane Lathuilière, and Giuseppe Valenzise, "A hybrid animation codec for low bitrate video conferencing," in *IEEE ICIP*, 2022.

[3] Ting-Chun Wang, Arun Mallya, and Ming-Yu Liu, "One-shot free-view neural talking-head synthesis for video conferencing," in *CVPR*, 2021.

[4] Maxime Oquab, Pierre Stock, Oran Gafni, Daniel Haziza, Tao Xu, Peizhao Zhang, Onur Celebi, Yana Hasson, Patrick Labatut, Bobo Bose-Kolanu, et al., "Low bandwidth video-chat compression using deep generative models," *arXiv preprint arXiv:2012.00328*, 2020.

[5] Bolin Chen, Zhao Wang, Binzhe Li, Rongqun Lin, Shiqi Wang, and Yan Ye, "Beyond keypoint coding: Temporal evolution inference with compact feature representation for talking face video compression," in *DCC*, 2022.

[6] Zhehao Chen, Ming Lu, Hao Chen, and Zhan Ma, "Robust ultralow bitrate video conferencing with second order motion coherency," in *2022 IEEE 24th International Workshop on Multimedia Signal Processing (MMSP)*, 2022, pp. 1–6.

[7] Aliaksandr Siarohin, Stéphane Lathuilière, Sergey Tulyakov, Elisa Ricci, and Nicu Sebe, "First order motion model for image animation," in *Neurips*, 2019.

[8] Oren Rippel and Lubomir Bourdev, "Real-time adaptive image compression," *arXiv preprint arXiv:1705.05823*, 2017.

[9] Johannes Ballé, David Minnen, Saurabh Singh, Sung Jin Hwang, and Nick Johnston, "Variational image compression with a scale hyperprior," *arXiv preprint arXiv:1802.01436*, 2018.

[10] Eirikur Agustsson, Michael Tschannen, Fabian Mentzer, Radu Timofte, and Luc Van Gool, "Generative adversarial networks for extreme learned image compression," in *IEEE ICCV*, 2019.

[11] Zhihao Hu, Guo Lu, and Dong Xu, "Fvc: A new framework towards deep video compression in feature space," in *IEEE Conference on Computer Vision and Pattern Recognition(CVPR)*, 2021.

[12] Johannes Ballé, Valero Laparra, and Eero P Simoncelli, "End-to-end optimized image compression," *Neurips*, 2016.

[13] Guo Lu, Wanli Ouyang, Dong Xu, Xiaoyun Zhang, Chunlei Cai, and Zhiyong Gao, "Dvc: An end-to-end deep video compression framework," in *IEEE CVPR*, 2019.

[14] Fabian Mentzer, George Toderici, David Minnen, Sung-Jin Hwang, Sergi Caelles, Mario Lucic, and Eirikur Agustsson, "Vct: A video compression transformer," in *NeurIPS*, 2022.

[15] J. Cleary and I. Witten, "Data compression using adaptive coding and partial string matching," *IEEE Transactions on Communications*, vol. 32, no. 4, pp. 396–402, 1984.

[16] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018.

[17] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large scale image recognition," in *ICLR*, 2015.

[18] Keyan Ding, Kede Ma, Shiqi Wang, and Eero P. Simoncelli, "Image quality assessment: Unifying structure and texture similarity," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022, vol. 44, pp. 2567–2581.