Multi-Frame Deformable Look-Up Table for Compressed Video Quality Enhancement

Gang He[1], Guancheng Quan[1,*], Chang Wu[1], Shihao Wang[2], Dajiang Zhou[2], Yunsong Li[1]

[1]Xidian University, Shaanxi, China
[2]Ant Group, Zhejiang, China
ghe@xidian.edu.cn, gcquan@stu.xidian.edu.cn, jackwu0630@gmail.com,
{shihao.wsh,dajiang.zdj}@antgroup.com,ysli@mail.xidian.edu.cn

Abstract

The rapid progress of multimedia technology has led to an increased focus on enhancing the quality of experience (QoE) for video. Specifically, the demand for low-latency and high-quality decoding has grown significantly. Compressed Video Quality Enhancement (CVQE) methods based on Deep Neural Networks (DNNs) have achieved remarkable success. However, most of the methods suffer from high computational complexity, thereby limiting their practicality in low-latency scenarios. Recently, Look-Up Table (LUT) methods have shown great efficiency, which makes them considerably promising in the field of low-latency CVQE. In this paper, we propose an efficient multi-frame deformable Look-Up Table structure for CVQE. Firstly, we design an efficient CNN to explore the inter-frame correlation and then predict the multi-scale convolution offsets. Secondly, we introduce a temporal feature extraction module and a multi-scale fusion module. We first exploit the predicted offsets to guide sampling for precise temporal alignment and extract multi-frame information. Then, higher quality frames are reconstructed from the fused multi-scale features. During the inference, we convert these two modules into LUTs to achieve a sound trade-off between model performance and computational complexity. Experiments demonstrate that our proposed method dramatically outperforms the state-of-the-art LUT-based methods, and obtains competitive performance compared to CNN-based methods with the capability to run in real-time(30fps) at 1080p resolution.

Introduction

In recent decades, the internet has developed by leaps and bounds, and the amount of videos has exploded. In order to deal with the considerable video data in the network, video compression is widely used in various practical application fields to reduce bandwidth and storage space. For example, H.264/AVC (Wiegand et al. 2003) and H.265/HEVC (Sullivan et al. 2012) are commonly used video coding standards. However, severe artifacts (e.g., blurring, ringing and blocking) are produced sometimes through video compression, especially in low bit-rate or low-latency scenarios. This may result in a degradation of the quality of experience (QoE).

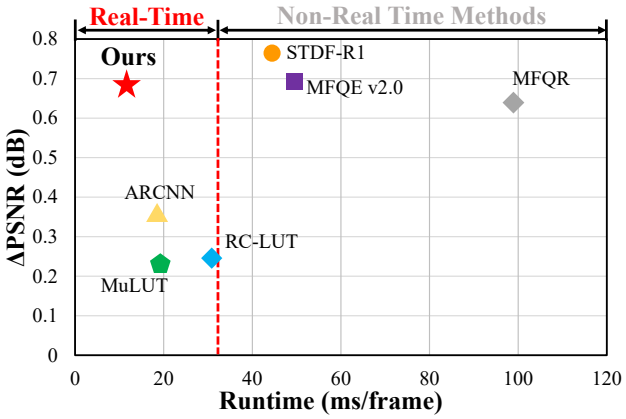

Figure 1: Illustration of different model performance at 1280×720 resolution at QP37. Real-time denotes the runtime faster than 30 fps.

In order to cope with these compression artifacts, some traditional methods, such as in-loop filtering (Norkin et al. 2012), SA-DCT (Foi, Katkovnik, and Egiazarian 2007), Sparse Coding (Jung et al. 2012), have been proposed. Nevertheless, most of these methods may be less flexible and have limited performance, which might make them being laborious to apply in the field of compressed video quality enhancement (CVQE) in low-latency scenarios.

With the rapid emergence of deep learning, plentiful works on CVQE are conducted based on deep neural networks. Early methods such as AR-CNN(Dong et al. 2015a), Deep Multi-scale CNN (Nah, Hyun Kim, and Mu Lee 2017), DnCNN (Zhang et al. 2017) and DS-CNN (Yang, Xu, and Wang 2017) were performed independently for single images. In order to utilize the temporal information, Yang et al. (Yang et al. 2018) first introduced multi-frame quality enhancement on compressed video (MFQE 1.0). Since then, extensive multi-frame based networks have been proposed. For example, Xing et al. (Guan et al. 2019) proposed MFQE 2.0 based on (Yang et al. 2018); Deng et al. (Deng et al. 2020) realized a multi-frame CVQE method based on Deformable Convolution (Dai et al. 2017). In addition, more and more image/video restoration methods (He et al. 2022a; Xu et al. 2023, 2024b,a; Wu et al. 2023, 2024) have been

*Corresponding author

proposed. However, these methods may have high computational complexity, resulting in a significant reliance on computing devices. Fast-MFQE (Chen et al. 2023) and MFQR (Liu and Jia 2024) were proposed to address low-latency scenarios, but they might still rely on high-performance devices.

Recently, Look-up Table(LUT)-based methods have shown significant efficiency and have resonated vigorously in the field of image/video super-resolution. Numerous works such as SR-LUT (Jo and Kim 2021), SPLUT (Ma et al. 2022), MuLUT (Li et al. 2022) and RC-LUT (Liu et al. 2023) have demonstrated the efficiency of converting the neural network to LUTs. For example, SR-LUT achieves an image super-resolution method by converting a network with a small receptive field to LUTs. MuLUT efficiently enlarges the receptive field of the model by cascading multiple LUTs. While RC-LUT decouples channel-wise and spatial calculation based on MuLUT, and further expands the receptive field to achieve greater performance. The utilization of multi-frame information for LUTs is proposed in ConvLUT (Yin et al. 2023), and the temporal information is used to generate weights for multiple expert LUTs. However, currently available LUT-based methods are not specific to the field of CVQE and most of them are based on single-frame images. The multi-frame method, on the other hand, might be difficult to fully utilize the temporal information by weighting the expert LUTs. Therefore, there is an imperative need to fill the technology gap of the low latency CVQE task.

To address the above issues, we propose an efficient multi-frame deformable Look-Up Table model for CVQE. Specifically, our proposed network contains a hybrid CNN-LUT structure. In the CNN part, we propose a Motion Feature Modulation Network (MFMN), a lightweight CNN structure introducing motion vector (MV) information. In the LUT part, we first utilize the predicted offsets from MFMN to guide the alignment of multi-frame information. Then we introduce a temporal feature extraction module and a multi-scale fusion module to extract multi-frame feature and reconstruct high-quality video frames. In the inference phase, these two modules are converted into a series of LUTs, leading to achieve a deformable LUT operation. In summary, the main contribution of this paper are as follows:

- We propose an efficient multi-frame deformable Look-Up Table structure for low-latency CVQE task, which can efficiently utilize multi-frame information to achieve more promising performance.
- We construct a Motion Feature Modulation Network, utilizing motion vectors for feature modulation to generate multi-scale convolution offsets. Intra-frame refinement and inter-frame alignment are implemented by means of the offset sampling operation.
- We propose a temporal feature extraction module and a multi-scale fusion module to reconstruct high quality video frames by means of feature extraction and fusion operation. Modules are converted to LUTs during inference phase to improve the model efficiency and thus achieve deformable LUT process.
- We quantitatively and qualitatively evaluate the performance of our proposed model on 18 HEVC Test Sequences, which outperforms existing LUT-based models and rivals multi-frame CNN models with less computational cost.

Related Work

Quality Enhancement

Early quality enhancement (QE) methods(Dabov et al. 2007; Foi, Katkovnik, and Egiazarian 2007; Jung et al. 2012; Norkin et al. 2012) were traditional models based on single image. Recently, extensive works (Dong et al. 2015b; Nah, Hyun Kim, and Mu Lee 2017; Zhang et al. 2017; Lai and Wang 2020; He et al. 2022b; Liu et al. 2024) based on DNN have been proposed for the image QE task. Specifically, (Lai and Wang 2020) utilized an improved Inception module and a self-attention mechanism for the QE task. (He et al. 2022b) proposed a deep dual-domain semi-blind network for image QE. (Liu et al. 2024) proposed a semantic-informed and two-phase QE approach designed for Martian images.

For the video quality enhancement (VQE) task, (Yang et al. 2018) proposed for the first time a multi-frame quality enhancement method for compressed video. Later, the optical flow estimation was utilized and improved in (Guan et al. 2019; Chan et al. 2022; Lin et al. 2022) for the VQE task. On the other hand, (Deng et al. 2020; Liu, Zhou, and Xiao 2022; Wang et al. 2019) adopted the deformable convolution to exploit multi-frame information with satisfactory results. Nevertheless, most of the methods are time-consuming and may not be applicable to the low-latency CVQE task.

LUT-based Network

Recently, LUT-based methods have shown significantly efficiency in many domains. For example, Zeng et al. (Zeng et al. 2020) proposed an adaptive 3D-LUT for real-time color enhancement of images. SRLUT (Jo and Kim 2021) was proposed by using the idea of the look-up table for image super-resolution. It achieves fast processing speed by converting complex computations to cheap in-memory lookups. MuLUT (Li et al. 2022) embraced the merits of SRLUT, and achieved greater results by cascading multiple LUTs. RCLUT (Liu et al. 2023) decoupled the channel computation from the spatial computation through the reconstructed convolution module, enlarging the receptive field by a factor of 9 compared to MuLUT.

Recently, ConvLUT (Yin et al. 2023) achieves compressed video super-resolution by utilizing multi-frame information. The temporal information is used to generate weights for multiple expert LUTs. However, this method may not be able to fully utilize the inter-frame information, while other LUT-based methods may be more inappropriate for compressed video quality enhancement with multiple frames.

Deformable Convolution

The concept of deformable convolution network (DCN) was first introduced by Dai et al. (Dai et al. 2017) to enhance

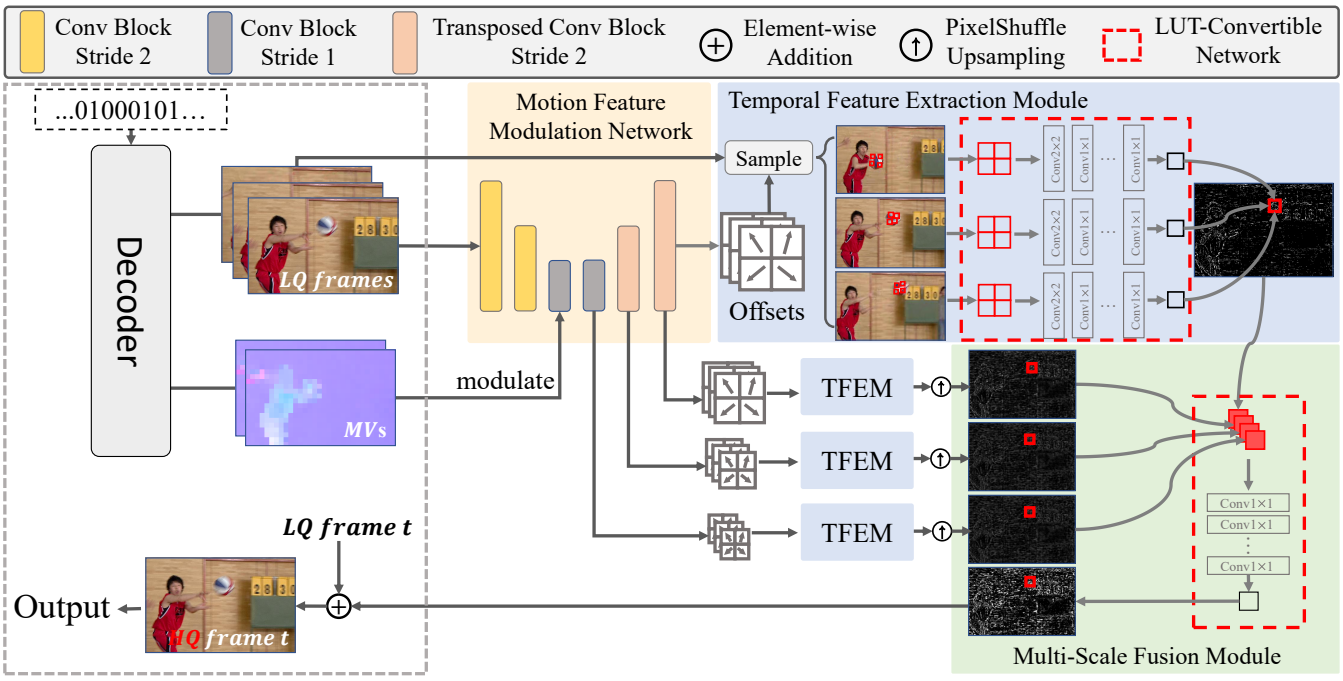

Figure 2: Overview of the proposed method for compressed video quality enhancement.

the ability of the regular convolution to cope with complex inputs. Later, several improved DCNs were proposed, such as DCNv2 (Zhu et al. 2019), DCNv3 (Wang et al. 2023), and DCNv4 (Xiong et al. 2024). By learning to obtain the offsets of each position of the convolution kernel, the method is able to realize improved results. On the basis of DCNs, several works (Bertasius, Torresani, and Shi 2018; Wang et al. 2019) have obtained superior multi-frame performance. Recently, deformable convolution was applied for compressed video enhancement task in (Deng et al. 2020; Liu, Zhou, and Xiao 2022). As a novel method for motion information extraction, they achieved better performance than previous ones.

Proposed Method

Overview

As shown in Fig.2, our proposed method receives three consecutive frames as input, as well as corresponding motion vectors for feature modulating. The goal of our method is to reconstruct enhanced video frames with the compression artifacts reduced. Specifically, the video frames $I^{lq}_{t-1,t,t+1} \in R^{H\times W\times 3}$ are considered as three related low quality frames (only Y channel). The motion vector from frame $t-1$ to frame t is denoted as $MV_{t-1\to t} \in R^{H\times W\times 2}$ (X and Y components, respectively). By learning the features of $I^{lq}_{t-1,t,t+1}$ through the network and modulating them by MVs, the final reconstruction high quality solution $\hat{I}^{hq}_t \in R^{H\times W\times 1}$ can be expressed as:

$$\hat{I}^{hq}_t = LUT[\mathcal{S}(I^{lq}_{t-1,t,t+1}, \mathcal{F}(I^{lq}_{t-1,t,t+1}, MVs))],$$

where $\mathcal{F}(\cdot)$ denotes the proposed CNN that extracts the kernel offsets, and $\mathcal{S}(\cdot)$ represents the sampling of the video frames through the offset to get the offset pixel values. $LUT[\cdot]$ is the proposed LUT-based implementation for feature extraction and multi-scale fusion.

Our proposed method consists of three parts: Motion Feature Modulation Network (MFMN), Temporal Feature Extraction Module (TFEM) and Multi-Scale Fusion Module (MSFM). MFMN is a lightweight CNN for extracting the offsets of the convolution kernel. TFEM and MSFM fuse temporal features and reconstruct high quality video frames through the LUT structure.

Motion Feature Modulation Network

In order to sufficiently utilize the intra- and inter-frame information, we propose a Motion Feature Modulation Network(MFMN) to predict the offset of the convolution kernel position. To be specific, frames $I^{lq}_{t-1,t,t+1}$ are fed to encoder blocks to obtain muti-frame information feature f^{encode}_t. Analogously, we feed the motion vectors to several convolution blocks for feature extraction:

$$f^{mv}_t = ReLU(C_{3\times3}(ReLU(C_{3\times3}(MV)))),$$

where $ReLU$ denotes Rectified Linear Unit, and $C_{3\times3}(\cdot)$ represent the convolution process with 3×3 kernel size. Afterwards, the two features are fused through a spatial fusion module. The modulated features can be obtained as:

$$f^{fuse}_t = \mathcal{F}_{fusion}([f^{encode}_t, f^{mv}_t]),$$

where $\mathcal{F}_{fusion}(\cdot)$ denotes a combination of 3×3 convolution and GELU activation, and $[\cdot,\cdot]$ indicates concatenation operation.

After the fused feature is obtained, it is up-sampled by three transposed convolution. At the same time, the feature

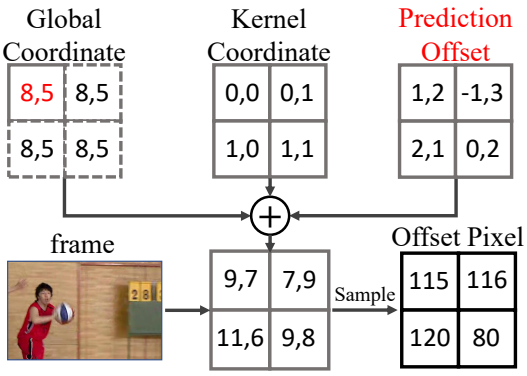

Figure 3: The coordinates of the four kernel positions are obtained by summing the predicted offsets with the global and local coordinates. Then the aligned pixel points can be obtained by indexing the corresponding coordinates.

is concatenated with the corresponding one in the encoding part. In this manner, the multi-scale features generated from the up-sampling process are preserved and then encoded to generate the offset prediction results. The process can be described as:

$$\delta^d_{t-1,t,t+1}, \theta^d_{t-1,t,t+1} = \mathcal{F}_{decoder}(f_t^{fuse}),$$

where $d \in [1, 2, 4, 8]$, which indicates the resolution of the offsets and masks as $\frac{1}{d}$ of the original video frame. $\delta^d_{t-1,t,t+1} \in R^{\frac{H}{d} \times \frac{W}{d} \times 3*2*k^2}$ denotes the predicted offset, and $\theta^d_{t-1,t,t+1} \in R^{\frac{H}{d} \times \frac{W}{d} \times 3*1*k^2}$ denotes the confidence for each offset, where k represents the kernel size of deformable convolution. For example, when $k = 2$ and $d = 1$, there are 24 channels in $\delta^1_{t-1,t,t+1}$, where the first, middle and last 8 channels are the convolution kernel offsets for frame $t-1$, t and $t+1$, respectively. For a 2×2 convolution kernel with single channel input, there are four weights in total. Each weight has offsets in the X and Y directions, making a total of 8 channels. The arrangement within the channels can be expressed as $[X_1, Y_1, X_2, Y_2, ...]$.

Temporal Feature Extraction Module

Upon the multi-scale offsets and masks are obtained, they need to be utilized in the deformable LUT structure. For regular deformable convolution, given the reference feature f_{ref}, offset δ and mask θ, the features after performing the alignment can be represented as:

$$f_{alig}(p) = DConv(I, \delta, \theta) = \sum_{k=1}^{k^2} \omega_k \cdot f_{ref}(p + p_k + \delta_k) \cdot \theta_k,$$

where k is the kernel size of deformable convolution, and p denotes the coordinate of each pixel point in the feature map. In addition, p_k represents the local coordinates of each weight in the convolution kernel, while δ_k is the offset of the corresponding parameter. Consequently, $p_k + \delta_k$ provides the local coordinate after the offset operation. Then it is added to p to obtain the new global coordinate of the current pixel in the whole feature map.

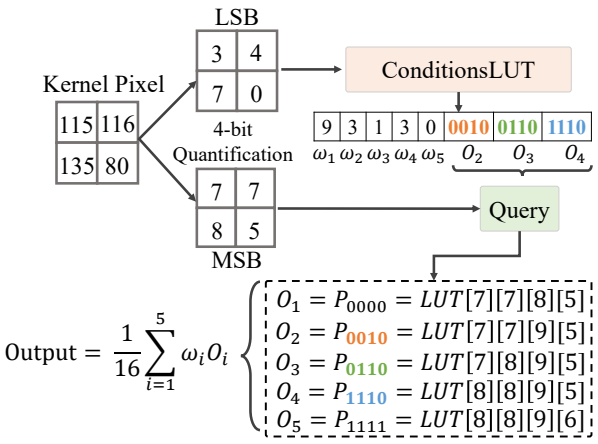

Figure 4: Illustration of the 4-bit quantization look-up table method based on tetrahedral interpolation. Convert frequent conditional judgments to a ConditionsLUT to accelerate the interpolation process.

Deformable LUT implementation. In this work, we adopt a channel-aware LUT-based method instead of the forward inference process of the neural network. Specifically, in order to convert the convolutions to LUTs, we adopt a deformable convolution with the kernel size of 2. Therefore, for the $t-1$ frame $I^{lq,d}_{t-1} \in R^{\frac{H}{d} \times \frac{W}{d} \times 1}$ with a downsampling scale of d, a corresponding offset δ^d_{t-1} and mask θ^d_{t-1} will be obtained from the preordered MFMN.

Thereafter, we obtain the global offset coordinates by generating grids. Firstly, grids of shape $\frac{H}{d} \times \frac{W}{d}$ are initialised, representing the x- and y-component of the global coordinates of each pixel in the video frame $I^{lq,d}_{t-1}$, respectively. Afterwards, grids indicating the x- and y-components of the local kernel coordinates are generated. These grids represent the 4 positions in the 2×2 convolution kernel. As shown in Fig.3, the global offset coordinates OC^d_{t-1} are obtained by combining these grids as:

$$OC^d_{t-1} = Grid_p + Grid_{p_k} + \delta^d_{t-1}.$$

Once the coordinates are obtained, the pixel values of each position after the offset can be looked up in the video frame and finally get the result $f^d_{t-1} \in R^{\frac{H}{d} \times \frac{W}{d} \times 4}$, where the 4 channels represent the 4 pixel values of the convolution kernel at the corresponding position. In this way, the offset pixel values of frame $t-1$ are sampled as:

$$f^d_{t-1} = Sample(I^{lq,d}_{t-1}, OC^d_{t-1}) \cdot \theta^d_{t-1}.$$

Afterwards, the computed results of the deformable convolution are converted to multiple LUTs. The table lookup and interpolation operations are performed on the obtained results:

$$\hat{f}^d_{t-1} = LUT_{TFEM}[f^d_{t-1}(0)][f^d_{t-1}(1)][f^d_{t-1}(2)][f^d_{t-1}(3)],$$

where $0, 1, 2, 3$ denotes the channel index, and LUT_{TFEM} represents both table lookup and interpolation processes in the LUT retrieval.

QP	Test Videos		High-Latency Methods			Low-Latency Methods			
			MFQE v2.0	STDF-R1	MFQR	AR-CNN	MuLUT	RCLUT	Ours
37	Class A	PeopleOnStreet	0.92/1.57	1.05/1.66	0.76/-	0.35/0.75	0.30/0.82	0.34/0.81	**0.80/1.60**
		Traffic	0.59/1.02	0.56/0.92	0.43/-	0.24/0.47	0.19/0.49	0.20/0.43	**0.50/1.04**
	Class B	BQTerrace	0.40/0.67	0.55/0.89	0.23/-	0.20/0.28	0.06/0.10	0.05/0.04	**0.28/0.58**
		BasketballDrive	0.47/0.83	0.60/0.99	0.26/-	0.23/0.55	0.10/0.46	0.14/0.39	**0.46/1.03**
		Cactus	0.50/1.00	0.59/1.06	0.40/-	0.19/0.38	0.13/0.37	0.15/0.37	**0.47/0.99**
		Kimono	0.55/1.18	0.66/1.32	0.12/-	0.22/0.65	0.21/0.86	0.23/0.75	**0.60/1.47**
		ParkScene	0.46/1.23	0.41/1.05	0.29/-	0.14/0.38	0.12/0.39	0.11/0.34	**0.36/1.07**
	Class C	RaceHorsesC	0.39/0.80	0.41/0.98	0.44/-	0.22/0.43	0.11/0.24	0.12/0.33	**0.33/1.05**
		BQMall	0.62/1.20	0.75/1.44	0.50/-	0.28/0.68	0.03/0.19	0.09/0.35	**0.43/1.15**
		PartyScene	0.36/1.18	0.52/1.49	0.40/-	0.11/0.38	0.02/0.16	0.04/0.34	**0.17/0.66**
		BasketballDrill	0.58/1.20	0.64/1.19	0.22/-	0.25/0.58	0.05/0.22	0.06/0.21	**0.38/1.15**
	Class D	RaceHorses	0.59/1.43	0.63/1.51	0.57/-	0.27/0.55	0.18/0.50	0.17/0.51	**0.40/1.24**
		BQSquare	0.34/0.65	0.75/1.03	0.58/-	0.08/0.08	-0.11/0.14	-0.08/0.15	**0.06/0.21**
		BlowingBubbles	0.53/1.70	0.53/1.69	0.45/-	0.16/0.35	0.10/0.62	0.13/0.64	**0.24/0.85**
		BasketballPass	0.73/1.55	0.80/1.54	0.39/-	0.26/0.58	0.09/0.24	0.11/0.32	**0.50/1.35**
	Class E	FourPeople	0.73/0.95	0.83/1.01	0.69/-	0.37/0.50	0.26/0.56	0.28/0.50	**0.68/1.05**
		Johnny	0.60/0.68	0.65/0.71	0.58/-	0.25/0.10	0.13/0.05	0.15/0.06	**0.54/0.69**
		KristenAndSara	0.75/0.85	0.84/0.83	0.64/-	0.41/0.50	0.29/0.57	0.32/0.47	**0.71/0.98**
	Average		0.56/1.09	0.65/1.18	0.44/-	0.23/0.45	0.13/0.39	0.15/0.39	**0.44/1.01**
32	Average		0.52/0.68	0.73/0.87	-	0.18/0.19	0.10/0.29	0.14/0.30	**0.39/0.66**
27	Average		0.49/0.42	0.67/0.53	-	0.18/0.14	0.09/0.15	0.11/0.24	**0.35/0.45**
22	Average		0.46/0.27	0.57/0.30	-	0.14/0.08	0.07/0.15	0.09/0.18	**0.26/0.25**

Table 1: Quantitative results of ΔPSNR (dB) / ΔSSIM ($\times 10^{-2}$) on 18 test videos at 4 different QPs.

Method	Runtime(fps)				
	A	B	C	D	E
MFQE2.0	2.8	5.6	31.5	76.8	13.1
STDF-R1	5.2	10.3	52.2	197.0	23.5
MFQR	6.7	12.6	15.9	58.9	10.1
AR-CNN	13.3	24.2	118.5	153.7	54.6
MuLUT	13.0	24.2	67.1	137.6	42.2
RC-LUT	8.5	17.7	44.0	55.8	30.9
Ours	**19.3**	**36.1**	**175.4**	**243.0**	**83.6**

Table 2: Runtime of compared methods at different resolutions.

LUT conversion. There are 256^4 combinations of inputs to the network of 2×2 receptive field. Therefore, a LUT of size 256^4 is stored after calculating all possible cases, which will take up a lot of storage space. Therefore, as shown in Fig.4, we sampled the LUT at equal intervals and quantized it to size 17^4 to save more space. When the quantized LUT needs to be looked up, the data near the lookup point needs to be interpolated to get the correct results. Thereafter, the least significant bit (LSB) values of kernel pixel are looked up in ConditionLUT to output the interpolation weights and interpolation points. This approach significantly reduces the time wasted due to high-frequency if-else judgements. Then, the five weights and five interpolation points are accumulated to generate the result of tetrahedral interpolation. Similarly, the result of frame t and $t+1$ can be obtained by the above process. Thus, summing the results from frame $t-1$ to $t+1$ gives the multi-scale results of the enhanced residuals:

$$\hat{R}_t^d = \hat{f}_{t-1}^d + \hat{f}_t^d + \hat{f}_{t+1}^d.$$

Multi-Scale Fusion Module

The previous module performs feature extraction of offset video frames from four different scales, resulting in 4-channel residual results. Therefore, in MSFM, these results are fused and enhanced by a series of 1×1 convolutions to end up with final enhanced residual results. A series of 4-channel 1×1 convolutions can be also converted to a 4D LUT. The conversion of this convolution module to LUTs can be expressed as:

$$\hat{R}_t^{lq} = LUT_{MSFM}[\hat{R}_t^{d=1}][\hat{R}_t^{d=2}][\hat{R}_t^{d=4}][\hat{R}_t^{d=8}],$$

where $\hat{R}_t$ denotes the residual result after quality enhancement. In order to reduce the size of LUTs, we also implemented a 4-bit quantization for 4D LUTs. The final enhancement solution at frame t is:

$$\hat{I}_t^{hq} = I_t^{lq} + \hat{R}_t^{lq}.$$

Experiments

Datasets

For model training, we selected CVCP (Chen et al. 2021) dataset. There are 589 video clips with 32 frames of 1080p

Figure 5: Qualitative results of different methods at QP37. The average ΔPSNR(dB) and ΔSSIM($\times 10^{-2}$) are indicated at the bottom of the images.

resolution, totalling 18848 8-bit grey scale video frames. For model testing, 18 standard test sequences (Ohm et al. 2012) with 150 to 600 frames per video from JCT-VC are utilized for evaluation. All mentioned videos were compressed in Low Delay P (LDP) mode using H.265/HEVC reference software HM16.20, as in previous works (Yang et al. 2018; Guan et al. 2019; Deng et al. 2020; Liu, Zhou, and Xiao 2022). In addition, we simultaneously extracted the motion vectors during the decoding process and stored them in the dataset. In order to evaluate the model at different compression levels, we performed video compression on 4 different Quantization Parameters (QPs), i.e., 22, 27, 32, 37.

Training Strategies

We constructed our method on Pytorch (Paszke et al. 2019) and the deformable convolution is based on DCNv2 (Zhu et al. 2019). We apply random crop of 256×256 to the input and set the batchsize to 8. We divide the training of the model into three stages: complete CNN training, sampling finetuning, and LUT-Convert finetuning. In each stage, we use Mean Square Error (MSE) loss to constrain the difference between the enhanced and original frames. All experiments are conducted in PyTorch2.1 with GPU V100s.

CNN training phase. Firstly, we utilize Adam (Kingma and Ba 2014) optimiser for CNN training on QP37 dataset, and set the initial learning rate as 10^{-4}. At the end of training, we utilize the trained model weights to finetune on datasets with QP of 22, 27 and 32, respectively.

Sampling reconstruction finetuning. Since the data types of the inputs and outputs of the Deformable LUTs are integers, directly converting the network (Float32) to LUTs for finetune may cause performance degradation. Therefore, we design an additional sampling finetuning stage to accomplish offset sampling and perceptual quantization in preparation for subsequent deformable LUT conversion. We first reconstruct the offset sampling process independently. Then, the features aligned by offset sampling are extracted by vanilla convolutions.

LUT-Convert finetuning. In the final stage, we convert the vanilla convolutions in the network into multiple LUTs. As in the previous work (Jo and Kim 2021; Ma et al. 2022; Li et al. 2022; Liu et al. 2023), 4-bit quantisation is applied at the time of conversion to reduce the size of the LUTs. In order to improve the stability of the converted LUTs, we adopt the LUT-aware Finetuning Strategy proposed by MuLUT. During the evaluation phase, we improve the efficient 4D-LUT interpolation method (Yin et al. 2023) based on CUDA to obtain high efficiency.

Comparison with State-of-the-Art

We compare our proposed method with state-of-the-art LUT-based methods and image/video quality enhancement methods, such as MFQE 2.0, STDF(STDF-R3), MFQR, AR-CNN, MuLUT(SDY x2) and RC-LUT(RC-3-5-7 x2). For all LUT-based methods, we performed the same CUDA acceleration for all 4D LUTs to guarantee fairness.

Quantitative results. Table 1 exhibit the performance comparison of the methods. As can be observed that the LUT-based methods achieve faster inference. Our proposed method also inherits the advantages of them, being able to

Training Phase	ΔPSNR(dB)	ΔSSIM($\times 10^{-2}$)	720p		1080p		1600p	
			Runtime(fps)	FLOPs(G)	Runtime(fps)	FLOPs(G)	Runtime(fps)	FLOPs(G)
CNN Training	0.456	1.076	8.3	235.4	3.7	529.6	1.9	1046.1
Sampling Finetuning	0.446	1.042	8.2	236.3	3.7	531.7	1.9	1050.2
LUT-Convert Finetuning	0.439	1.009	**83.6**	**58.3**	**36.1**	**131.2**	**19.3**	**259.1**

Table 3: Comparison of model performance at different training stages under QP37.

Model	MFMN			TFEM		MSFM	ΔP/ΔS
	MV	Single-frame	Multi-frame	Single-scale	Multi-scale		
(A)		✓			✓	✓	0.33/0.76
(B)			✓	✓			0.36/0.83
(C)			✓		✓		0.38/0.89
(D)			✓		✓	✓	0.41/0.98
Ours	✓		✓		✓	✓	**0.44/1.01**

Table 4: Ablation study on different module structures.

Methods	Query Times		Query Runtime(ms)		ΔPSNR/ΔSSIM
	1D-LUT	4D-LUT	720p	1080p	
SR-LUT	0	4	3.57	6.99	0.06/0.19
MuLUT	0	24	23.70	41.32	0.13/0.39
RC-LUT	12	24	32.36	56.50	0.15/0.39
Ours	0	13	4.4	7.9	**0.44/1.01**

Table 5: Comparison of query times and performance of different LUT-based methods.

run in real-time(36fps) on 1080p and outperforming all compared LUT-based methods. As for high-latency methods, our method obtain competitive results compared with the video quality enhancement method MFQE 2.0 and MFQR.

Qualitative results. Fig.5 shows the qualitative results of test videos between different methods. It can be noticed that there are considerable artifacts in compressed videos such as blurring, ringing and blocking. The CNN-based multi-frame method can reduce these artifacts decently, while the enhanced results produced by LUT-based method still have obvious ringing (*FourPeople*), bluring (*Kimono*) and blocking effects (*BasketballDrive*). In comparison, our proposed method is able to retrieve more details and robustly reconstruct video frames.

Ablation Study

In this section, we conduct ablation study on both model structure and training strategy, and also analyzed and discuss the experiment results. Specifically, we measure the performance and efficiency of the model at QP37, including average ΔPSNR (dB), average ΔSSIM (10^{-2}), inference speed (fps) and float-point operations (FLOPs). Numerous experiments demonstrate the effectiveness of our proposed model.

The effectiveness of LUT conversion. We conduct two model finetunings to cope with the loss of performance during the LUT conversion. Table 3 exhibits the performance of the model in different training phases. It can be observed that the quantity of computation of full CNN is extremely high. After the sampling finetuning, there is a slight degradation in the performance of the model. However, the computational amount and time consumption are improved exponentially after the LUT-convert stage. Specifically, to achieve the same feature extraction for a frame of 1080p, the CNN module takes 217.8ms, while TFEM takes only 7.4ms. In addition, MSFM takes only 0.5ms to process a frame of a 1080p video. It is extremely faster compared to the corresponding CNN module, which takes 32.8ms.

The effectiveness of proposed modules. Table 4 demonstrates the performance under different modules. Firstly, the comparison between (A) and our model illustrates the significance of introducing multi-frame information. Secondly, model (B) and (C) demonstrate the performance improvement of the multi-scale TFEM. In addition, (D) illustrating the effectiveness of the enhancement of the multi-scale aligned information. Finally, the introduction of the MV leads to further improvements in model performance.

The effectiveness of deformable LUTs. For video frames with a single channel input, previous methods need to query four times (multiple queries during interpolation are counted as one) to produce each output pixel point due to the rotation strategy (rotation of 0, 90, 180, and 270 degrees). Since our method does not take the rotation strategy, the number of LUT queries is much smaller. In addition, the multi-scale structure allows our method to complete queries faster. Table 5 shows that our method possesses higher efficiency than the compared LUT-based methods.

Conclusion

In this work, we propose an efficient multi-frame deformable Look-Up Table structure for low-latency CVQE task. We propose MFMN to exploit multi-frame information and predict multi-scale offsets, which are utilized to perform temporal alignment. Then, we introduce a temporal feature extraction module and a multi-scale fusion module to fuse and enhance aligned information. Finally, the conversion of the two modules to LUTs are performed to obtain significant efficiency improvement. Experiments demonstrate that our method outperforms the existing state-of-the-art LUT-based method, and has a satisfying trade-off between time consumption and model performance. Moreover, we believe that our proposed method can be extended for other low-level video tasks.

Acknowledgments

This work was supported by Ant Group Research Fund, the 2023 Research Project of Shaanxi Provincial Department of Transportation, No. 23-58X.

References

Bertasius, G.; Torresani, L.; and Shi, J. 2018. Object detection in video with spatiotemporal sampling networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 331–346.

Chan, K. C.; Zhou, S.; Xu, X.; and Loy, C. C. 2022. Basicvsr++: Improving video super-resolution with enhanced propagation and alignment. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 5972–5981.

Chen, K.; Chen, J.; Zeng, H.; and Shen, X. 2023. Fast-MFQE: A Fast Approach for Multi-Frame Quality Enhancement on Compressed Video. *Sensors*, 23(16): 7227.

Chen, P.; Yang, W.; Wang, M.; Sun, L.; Hu, K.; and Wang, S. 2021. Compressed domain deep video super-resolution. *IEEE Transactions on Image Processing*, 30: 7156–7169.

Dabov, K.; Foi, A.; Katkovnik, V.; and Egiazarian, K. 2007. Image denoising by sparse 3-D transform-domain collaborative filtering. *IEEE Transactions on image processing*, 16(8): 2080–2095.

Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; and Wei, Y. 2017. Deformable convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, 764–773.

Deng, J.; Wang, L.; Pu, S.; and Zhuo, C. 2020. Spatio-temporal deformable convolution for compressed video quality enhancement. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 10696–10703.

Dong, C.; Deng, Y.; Loy, C. C.; and Tang, X. 2015a. Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE international conference on computer vision*, 576–584.

Dong, C.; Deng, Y.; Loy, C. C.; and Tang, X. 2015b. Compression artifacts reduction by a deep convolutional network. In *Proceedings of the IEEE international conference on computer vision*, 576–584.

Foi, A.; Katkovnik, V.; and Egiazarian, K. 2007. Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images. *IEEE transactions on image processing*, 16(5): 1395–1411.

Guan, Z.; Xing, Q.; Xu, M.; Yang, R.; Liu, T.; and Wang, Z. 2019. MFQE 2.0: A new approach for multi-frame quality enhancement on compressed video. *IEEE transactions on pattern analysis and machine intelligence*, 43(3): 949–963.

He, G.; Xu, K.; Xu, L.; Wu, C.; Sun, M.; Wen, X.; and Tai, Y.-W. 2022a. SDRTV-to-HDRTV via hierarchical dynamic context feature mapping. In *Proceedings of the 30th ACM International Conference on Multimedia*, 2890–2898.

He, J.; He, X.; Zhang, M.; Xiong, S.; and Chen, H. 2022b. Deep dual-domain semi-blind network for compressed image quality enhancement. *Knowledge-Based Systems*, 238: 107870.

Jo, Y.; and Kim, S. J. 2021. Practical single-image super-resolution using look-up table. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 691–700.

Jung, C.; Jiao, L.; Qi, H.; and Sun, T. 2012. Image deblocking via sparse representation. *Signal Processing: Image Communication*, 27(6): 663–677.

Kingma, D. P.; and Ba, J. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Lai, P.-R.; and Wang, J.-S. 2020. Multi-stage attention convolutional neural networks for HEVC in-loop filtering. In *2020 2nd IEEE International Conference on Artificial Intelligence Circuits and Systems (AICAS)*, 173–177. IEEE.

Li, J.; Chen, C.; Cheng, Z.; and Xiong, Z. 2022. Mulut: Cooperating multiple look-up tables for efficient image super-resolution. In *European conference on computer vision*, 238–256. Springer.

Lin, J.; Hu, X.; Cai, Y.; Wang, H.; Yan, Y.; Zou, X.; Zhang, Y.; and Van Gool, L. 2022. Unsupervised flow-aligned sequence-to-sequence learning for video restoration. In *International Conference on Machine Learning*, 13394–13404. PMLR.

Liu, C.; and Jia, K. 2024. Multi-Frame Quality Recovery Model for Compressed Video Enhancement. *IEEE Transactions on Consumer Electronics*.

Liu, C.; Xu, M.; Xing, Q.; and Zou, X. 2024. MarsQE: Semantic-Informed Quality Enhancement for Compressed Martian Image. *arXiv preprint arXiv:2404.09433*.

Liu, G.; Ding, Y.; Li, M.; Sun, M.; Wen, X.; and Wang, B. 2023. Reconstructed convolution module based look-up tables for efficient image super-resolution. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 12217–12226.

Liu, J.; Zhou, M.; and Xiao, M. 2022. Deformable convolution dense network for compressed video quality enhancement. In *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1930–1934. IEEE.

Ma, C.; Zhang, J.; Zhou, J.; and Lu, J. 2022. Learning series-parallel lookup tables for efficient image super-resolution. In *European Conference on Computer Vision*, 305–321. Springer.

Nah, S.; Hyun Kim, T.; and Mu Lee, K. 2017. Deep multi-scale convolutional neural network for dynamic scene deblurring. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 3883–3891.

Norkin, A.; Bjontegaard, G.; Fuldseth, A.; Narroschke, M.; Ikeda, M.; Andersson, K.; Zhou, M.; and Van der Auwera, G. 2012. HEVC deblocking filter. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(12): 1746–1754.

Ohm, J.-R.; Sullivan, G. J.; Schwarz, H.; Tan, T. K.; and Wiegand, T. 2012. Comparison of the coding efficiency of video coding standards—including high efficiency video coding (HEVC). *IEEE Transactions on circuits and systems for video technology*, 22(12): 1669–1684.

Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Sullivan, G. J.; Ohm, J.-R.; Han, W.-J.; and Wiegand, T. 2012. Overview of the high efficiency video coding (HEVC) standard. *IEEE Transactions on circuits and systems for video technology*, 22(12): 1649–1668.

Wang, W.; Dai, J.; Chen, Z.; Huang, Z.; Li, Z.; Zhu, X.; Hu, X.; Lu, T.; Lu, L.; Li, H.; et al. 2023. Internimage: Exploring large-scale vision foundation models with deformable convolutions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14408–14419.

Wang, X.; Chan, K. C.; Yu, K.; Dong, C.; and Change Loy, C. 2019. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 0–0.

Wiegand, T.; Sullivan, G. J.; Bjontegaard, G.; and Luthra, A. 2003. Overview of the H. 264/AVC video coding standard. *IEEE Transactions on circuits and systems for video technology*, 13(7): 560–576.

Wu, C.; He, G.; Lai, X.; and Li, Y. 2023. MPCNet: Compressed multi-view video restoration via motion-parallax complementation network. *Neural Networks*, 167: 601–614.

Wu, C.; Quan, G.; He, G.; Lai, X.-Q.; Li, Y.; Yu, W.; Lin, X.; and Yang, C. 2024. QS-NeRV: Real-Time Quality-Scalable Decoding with Neural Representation for Videos. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 2584–2592.

Xiong, Y.; Li, Z.; Chen, Y.; Wang, F.; Zhu, X.; Luo, J.; Wang, W.; Lu, T.; Li, H.; Qiao, Y.; et al. 2024. Efficient Deformable ConvNets: Rethinking Dynamic and Sparse Operator for Vision Applications. *arXiv preprint arXiv:2401.06197*.

Xu, K.; He, G.; Xu, L.; Yang, X.; Sun, M.; Wang, Y.; Ma, Z.; Fan, H.; and Wen, X. 2023. Towards robust sdrtv-to-hdrtv via dual inverse degradation network. *arXiv preprint arXiv:2307.03394*.

Xu, K.; Ma, Z.; Xu, L.; He, G.; Li, Y.; Yu, W.; Han, T.; and Yang, C. 2024a. An End-to-End Real-World Camera Imaging Pipeline. In *ACM Multimedia 2024*.

Xu, K.; Xu, L.; He, G.; Yu, W.; and Li, Y. 2024b. Beyond Alignment: Blind Video Face Restoration via Parsing-Guided Temporal-Coherent Transformer. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence,* {*IJCAI-24*}, 1489–1497.

Yang, R.; Xu, M.; and Wang, Z. 2017. Decoder-side HEVC quality enhancement with scalable convolutional neural network. In *2017 IEEE International Conference on Multimedia and Expo (ICME)*, 817–822. IEEE.

Yang, R.; Xu, M.; Wang, Z.; and Li, T. 2018. Multi-frame quality enhancement for compressed video. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 6664–6673.

Yin, G.; Qu, Z.; Jiang, X.; Jiang, S.; Han, Z.; Zheng, N.; Liu, X.; Yang, H.; Yang, Y.; Li, D.; et al. 2023. Online Streaming Video Super-Resolution with Convolutional Look-Up Table. *arXiv preprint arXiv:2303.00334*.

Zeng, H.; Cai, J.; Li, L.; Cao, Z.; and Zhang, L. 2020. Learning image-adaptive 3d lookup tables for high performance photo enhancement in real-time. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(4): 2058–2073.

Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; and Zhang, L. 2017. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26(7): 3142–3155.

Zhu, X.; Hu, H.; Lin, S.; and Dai, J. 2019. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9308–9316.