Multimedia Tools and Applications
https://doi.org/10.1007/s11042-022-13484-w

1221: DEEP LEARNING FOR IMAGE/VIDEO COMPRESSION AND VISUAL QUALITY ASSESSMENT

End-to-end video compression for surveillance and conference videos

Shenhao Wang[1] · Yu Zhao[2] · Han Gao[2] · Mao Ye[2] · Shuai Li[3]

Received: 29 June 2021 / Revised: 17 September 2021 / Accepted: 13 July 2022

Abstract

The storage and transmission tasks of surveillance and conference videos are an important branch of video compression. Since surveillance and conference videos have strong inter-frame correlation, considerable continuity at the image level and motion level between the consecutive frames exists. However, traditional video codec networks cannot fully use the characteristics of surveillance and conference videos during compression. Therefore, based on the DVC video codec framework, we propose a "MV residual + MV optimization" coding strategy for surveillance and conference videos to further reduce the compression rate and improve the quality of compressed video frames. During the testing stage, the online update strategy is promoted, which adapts the network's parameters to different surveillance and conference videos. Our contribution is to propose an optical flow residual coding method for videos with strong inter-frame correlation, implement optical flow optimization at decoding end and online update strategy at the encoding end. Experiments show that our method can outperform DVC framework, especially on CUHK Square surveillance video with 1.2dB improvement.

✉ Yu Zhao
YuZhao10086@gmail.com

Shenhao Wang
shenhao2@andrew.cmu.edu

Han Gao
han.gao@std.uestc.edu.cn

Mao Ye
cvlab.uestc@gmail.com

Shuai Li
shuaili@sdu.edu.cn

[1] School of Physics, University of Electronic Science and Technology of China, Chengdu 611731, People's Republic of China

[2] School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 611731, People's Republic of China

[3] School of Information Communication, Shandong University, Jinan 250000, People's Republic of China

Published online: 05 August 2022

Keywords Deep learning · End-to-end video compression · Surveillance and conference videos · Online update

1 Introduction

In 2017, it was reported that video transmission caused 75% of Internet congestion, and it will reach 82% [11] in 2022. Therefore, the task of video compression is an area that urgently needs in-depth research to reduce the size of video bitstream and decrease the probability of data congestion. In addition, with increasing awareness in public and private safety, high need for epidemic prevention and control supervision, exploding demand for client access, and necessity of behavioral content analysis, the transmission and storage of surveillance and conference videos are facing great challenge. However, despite sacrificing video quality, these gigantic video data still need to occupy a large amount of memory space for storage and big bitstream for transmission. With similar characteristics to surveillance video content, conference videos also face the same transmission and storage problems. At present, most of surveillance and conference videos compression are still implemented by H.264 [19] and H.265 [28]. However, while these coding standards achieve high compression effects, they are often accompanied by higher complexity. Due to the rapid iterative improvement of graphics processing units (GPUs), deep learning has experienced unprecedented development. In 2012, Krizhevsky et al. proposed the AlexNet network structure [13], and won the championship in the ImageNet competition with a very low error rate, which caused a corresponding trend of deep learning. Therefore, in H.265 [28], many modules have absorbed different deep learning methods to a certain extent, and most new video compression methods are also merged from deep learning methods to traditional methods.

The method that researchers first proposed is the hybrid of classic compression algorithms and deep learning networks, which can be divided into five categories, corresponding to five important modules in video compression: intra-prediction, inter-prediction, quantization, entropy coding, and loop filtering. In 2018, Cui et al. proposed the IPCNN network to optimize the intra-prediction efficiency in HEVC [7]; Li et al. proposed the IPFCN network to replace the original prediction module [15] in HEVC, achieving average BD-rate savings of 3.0%. Besides, many deep learning networks have been proposed for inter-frame prediction to improve the overall coding effect. For example, the CNNMCR network proposed by Huo et al. [10] and the FRCNN network proposed by Yan et al. [38]. In classic video compression, quantization and entropy coding are two important stages, which correspond to lossy compression and lossless compression respectively. Alam et al. proposed a network structure optimized for these two stages based on VNet-2 [2], Song et al. [26] also put forward an improved method based on LeNet [14], and achieved considerable results. Although the aforementioned networks have achieved effective results in reducing the redundancy of the frame sequence by careful design, these individually designed networks require manual manipulation through the whole compression system and this undoubtedly results in low effectiveness. Thus, they still cannot convey end-to-end video encoding tasks.

To take advantages of the deep leaning method and ease the whole compression process while avoiding flaws that hybrid compression methods have, a series of end-to-end video codec schemes have been proposed recently. These work can be divided into two categories: frame interpolation and alignment. In the area of frame interpolation, Wu et al. [34] proposed a interpolation-based method based on RNNs, which uses information of near key frames to interpolate the intermediate frames repeatedly. While Djelouah et al. [8] reported an interpolation method by utilizing comprehensive motion information, which continuously interpolates to obtain former images. Different from the hierarchical interpolation

proposed by Wu et al., the method proposed by Djelouah et al. is carried on by sequence order. The motion information used by the above interpolation-based compression method is still the traditional block-based motion estimation. The interpolation operation mainly uses the forward frame and the backward frame as the key reference to compress the current frame. But for low-latency compression, it is more difficult to convey interpolation as the backward frame is not available. Therefore, video compression frameworks based on the alignment method are proposed. The most common alignment method is optical flow alignment. The DVC framework by Lu et al. [17] only relies on information of forward frames to compress the current frame, which is more feasible for low-latency transmission. In M-LVC, Lin et al. [16] combined multi-frame processing operation with the optical flow alignment method. In addition to optical flow alignment, deformable alignment methods have also been used in video compression in nearest papers. For example, in FVC, Hu et al. [9] replaced optical flow mode with deformable alignment to boost compression performance.

Due to the needs of security work, epidemic prevention and control, and behavior analysis, the transmission and storage tasks of surveillance and conference videos need to be further resolved to meet the increasing demand. However, the above-mentioned network frameworks are not developed for surveillance and conference videos compression, but focused on the universal video compression problem. They did not optimized for such a type of video with less variation in context. Thus, to meet the demand of surveillance and conference videos compression, the following works have been proposed. Both Sengar et al. [24] and Wu et al. [36] proposed foreground-background parallel compression approach, but the former is still a hybrid coding approach, while Wu et al. used an end-to-end network scheme. Zhao et al. [40], on the other hand, process the redundant information of foreground and background in different ways, thus achieving the reduction of video bit-rate. In contrast, [35] extracts skeleton information to optimize the coding of surveillance video through adversarial networks. Considering strong inter-frame correlation of surveillance and conference videos, we propose a more feasible framework for end-to-end surveillance and conference videos compression.

Our approach is still based on the DVC framework and the traditional video coding schemes, which consists of six parts, namely, motion estimation, motion compensation, quantization and entropy coding, inverse conversion, optical flow refinement, and frame reconstruction. In the motion estimation stage, we choose the pixel-level motion information, i.e. optical flow. And due to the robust inter-frame correlation of surveillance conference video, we introduce optical flow residuals to further remove the redundant information, as well as reduce compressed bit-rate. The details of this operation is as follows: Optical flow of the first and second frames in a GOP is kept intact, and the remaining optical flow in this GOP is obtained by continuously adding residuals to the previous optical flow. For the optical flow errors and missing details caused by the residual operation and lossy compression, we use the optical flow refinement network to compensate the optical flow at decoding side, so that the decoded motion information could be more similar to the original estimated motion information. What's more, we also introduce online update mechanism in the testing stage, so that our network can approach local optimal solution on specific video and thus improve the testing result.

The main contributions and novelty are summarized as follows:

1. We proposed an end-to-end video compression network particularly for surveillance and conference videos, by considering the strong inter-frame correlation of surveillance and conference videos, for which we adopt residuals of optical flows to cut down the bitrate while completing end-to-end video compression task.
2. An optical flow refinement module is developed to recover the lost details during lossy compression, making the reconstructed optical flows closer to the real optical flows.

3. We proposed to use an online update mechanism in the testing stage, which enables the network parameters to approach the local optimal solution on the current video, providing the network with adaptability and improving the testing results. For surveillance and conference videos, online updating could have significant performance improvement due to the strong inter-frame correlation.

Our method achieves better results in surveillance and conference videos than several traditional and end-to-end video codec methods such as H.264 [19] and DVC [17].

2 Related work

Recently, in addition to image codec methods that rely on manual operations (e.g., JPEG [32], JPEG2000 [25], and BPG [6]), deep learning based methods have been improved and employ different classes of neural networks, such as RNN [12, 30, 31], CNN [4, 5, 20, 29], etc. And in the field of video compression, most of the methods are based on image frames for redundant information rejection in time or spatial domain. Therefore image compression techniques have considerable importance for video compression, and many of these networks can be incorporated into video codec, such as the residual codec module in our framework. However, due to limitation of space, here we only briefly introduce the related work of hybrid video compression and end-to-end video compression.

2.1 Hybrid video compression

In recent years, a number of video coding standards have been settled, such as H.264, H.265, and the upcoming H.266, all of which follow the framework of predictive coding. In H.265, there are already many modules that incorporate deep neural network models. Hybrid video compression can be classified by following modules: intra-frame prediction, inter-frame prediction, quantization, entropy coding, and loop filtering.

In terms of intra-frame prediction, Cui et al. proposed an IPCN networks to optimize intra-frame prediction in HEVC [7], which became the first work to integrate CNN into intra prediction. They use current block and adjacent blocks in HEVC as the input of their network to get block-based residual information to refine the current block, and the efficiency of intra-frame prediction is thus optimized. Li et al. raised an IPCN network [15] to replace the original prediction module in HEVC, thus achieving an average 3.0% BD-rate savings. However, this network has a relative rise in computation time of about 3-265 times due to the complex structure. For inter-frame prediction, Huo et al. proposed a CNNMCR network [10]. This is a CNN-based motion compensation refinement network that can reduce errors during inter-frame prediction by using the motion compensation estimation of current block as well as adjacent reconstructed blocks to reduce the noise and artifacts caused by block operations. A CNN-based FRUC approach has also been adopted to generate direct visual reference frames as a way to improve inter-frame prediction.

In typical video compression, quantization and entropy coding are two very important stages. Alam et al. proposed a method based on VNet-2 for quantization optimization [2], which is divided into two steps: In the first step, VNet-2 is used to estimate the proximity visibility threshold CT, which will be used to guide quantization in the second step. In the second step(entropy encoding stage), Song et al [26] also proposed an improvement of CABAC based on LeNet [14], using CNN neural networks to directly predict the probability distribution.

For surveillance and conference videos, hybrid coding forms include [24, 40]. Sengar et al. [24] combines advantages of block-based and object-based coding techniques, using adaptive thresholding-based optical flow techniques to segment foreground moving objects

from background and determine foreground contours using Freeman chain code. Then, the block-based motion estimation and compensation are performed using variants of particle swarm optimization. Finally, an entropy coding method based on DCT and Huffman coding is utilized to compress the data representation. Zhao et al. [40], on the other hand, take different approaches to decrease redundant information of foreground and background. They proposed a block level background reference frame to reduce background redundancy and used a supervised predictive generative adversarial network (SP-GAN [27]) to generate foreground reference frames that help to deal with foreground redundancy.

2.2 End-to-end video compression

Although the aforementioned networks all used deep learning techniques to make improvement for different modules in HEVC and achieved significant results. However, these networks are only modified and enhanced for a single module, and for implementing a whole coding system, they still require manual manipulation and are not able to perform end-to-end coding. Therefore, end-to-end video coding schemes are increasingly studied to absorb effectiveness by deep learning method and decrease the difficulty during network's training. It can be mainly divided into: image interpolation and alignment.

The interpolation approach is widely utilized in end-to-end network schemes. Interpolation operation can recover the current frame by two or more key frames, which means that non-key frames do not need to occupy bitstreams during transmission and do not incur additional storage costs. Wu et al. [34] proposed a RNN-based hierarchical interpolation method, which first uses two key frames to generate intermediate frame, and then restores this intermediate frame as a key frame to built other sub-intermediate frames. Djelouah et al. [8], on the other hand, synthesize motion information and continuously interpolate backward to obtain interpolated images, while the generated images are then used as the basis for next interpolation operation. In other words, the interpolation operation actually uses information of former and later key frames to repeatedly obtain different level interpolation images.

Alignment is another implementation of end-to-end network scheme. Although interpolation approach has great significance for bit-rate reduction, it is difficult to be applied to low-latency transmission as interpolation operation requires forward and backward frames to support compression. Therefore, for low-latency transmission, optical flow alignment has become one of hot spots, because optical flow only needs information of previous frames to complete the inter-frame prediction task. DVC framework [17] proposed by Lu et al. is a typical motion prediction-based coding model that relies only on information of forward frames to compress the current frame. The optical flow information is compressed directly in DVC, and for image information, only the first frame is encoded, while residual information is passed to decoder to assist reconstruction. M-LVC [16] also uses optical flow alignment but works mainly on multiple frame prediction to assist video compression. In FVC [9], Hu et al. use deformable alignment, and abandon transmission from image level but instead implements the codec through feature-level alignment operations.

With regard to end-to-end network on surveillance and conference videos, Wu et al. [36] came up with the method of foreground-background parallel compression. Since the background of surveillance and conference videos is relatively fixed, the foreground can be extracted after establishing the background frame, which often occupies only a rather small part. So the spatial redundant information can be shrank by foreground-background separate compression. And they also created an update strategy for background, so that information in the background such as object moves and light changes could be well retained. Except foreground-background parallel compression, Wu et al. [35] obtained a global spatio-temporal feature and skeleton for each frame by an RNN networks, and reconstructed video frames using generative adversarial networks accordingly, thus achieving the

reduction of surveillance video bit-rate. But we do not compare to this two methods for they are not suitable for low-latency compression. Because, the background information and skeleton information required to help encoding should be generated before compression.

3 Proposed method

3.1 Overview of the proposed method

Figure 1 briefly illustrates our proposed network structure, which consists of six parts.

Step 1. **Motion estimation and compression.** In the motion estimation module, the current frame x_t will be input to the prediction network along with the reference frame $\hat{x}_{t-1}$, and this network will extract the motion information v_t from $t-1$ to t moments based on information of these two frames. Here, we use Spynet [22] as our optical flow prediction network, which is a relatively lighter-weight network. Since direct transmission of motion information can not effectively reduce bit-rate due to some redundant information in it, we use a auto-encoder here to encode optical flows for transmission and then recover it by decoder. Due to the robust inter-frame correlation of surveillance and conference videos, we also introduce optical flow residual $R_t = v_t - \hat{v}_{t-1}$. The encoding side of the auto-encoder would encode R_t into Y_t, and after quantization operation, we are able to obtain the quantized optical flow residuals $\hat{Y}_t$. When $\hat{Y}_t$ is entropy encoded, it will be restored to $\hat{R}_t$ after transmission to the decoding side.

Step 2. **Motion compensation and compression.** The predicted frame $\bar{x}_t$ of the current frame x_t can be obtained by applying the motion vector v_t estimated in the previous step with previous reconstructed frame $\hat{x}_{t-1}$. The above process is carried out by a motion compensation network, which works in two steps: in the first step, the predicted motion vector v_t is applied to $\hat{x}_{t-1}$ to obtain the transformed image by simple mathematical operations; in the second step, the motion vector v_t, $\hat{x}_{t-1}$ and the transformed image are fed together into the convolutional neural network

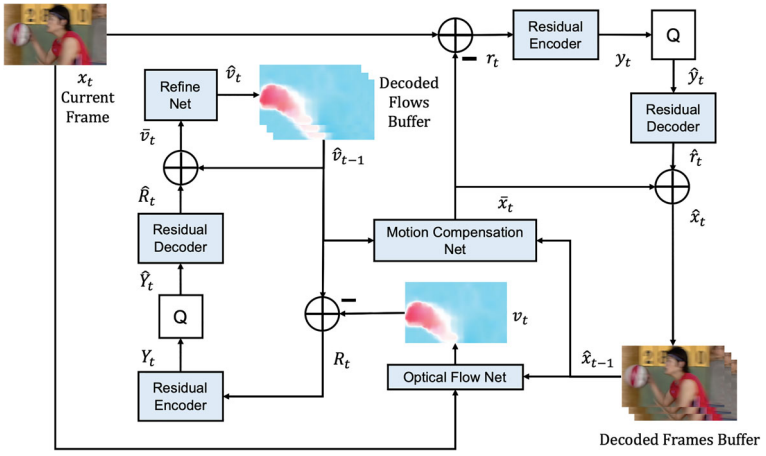

Fig. 1 Schematic diagram of end-to-end compression for surveillance and conference videos

to generate the predicted image $\bar{x}_t$. In this step, the image residual r_t could also be obtained by subtracting the current predicted frame $\bar{x}_t$ from the real x_t, and the residual information would be transmitted to decoder. The same as optical flow residual transmission, the encoder of image residual encodes r_t into y_t, which is then quantized to obtain latent presentation $\hat{y}_t$. When $\hat{y}_t$ go through entropy coding, the latent presentation will be restored to $\hat{r}_t$ at decoding side.

Step 3. **Optical flow refinement.** After decoding the optical flow residual $\hat{R}_t$, we add it to $\hat{v}_{t-1}$ to obtain the preliminary reconstructed optical flow $\bar{v}_t$ at t moment. Due to the errors caused by quantization, we adopt an optical flow refinement network to further improve and optimize $\bar{v}_t$ to obtain the final reconstructed optical flow $\hat{v}_t$.

Step 4. **Quantization.** The residual information r_t and R_t obtained in steps 1 and 2 will be quantized into y_t and Y_t. Here, since the quantization operation in traditional coding approach is not differentiable, it is not possible to back-propagate for deep neural networks, which causes difficulties during the training of end-to-end network. Many papers [1, 4, 30] have done work on this and proposed significant methods to resolve this problem. In framework proposed by us, the method of adding uniform noise [4] is used to perform quantization.

Step 5. **Entropy coding.** After the residual information r_t and R_t are encoded and quantized to obtain the latent presentations $\hat{y}_t$ and $\hat{Y}_t$. $\hat{y}_t$ and $\hat{Y}_t$ will be converted into bitstreams by entropy coding and transmitted to the decoding side. In order to realize end-to-end network compression, the entropy coding network used here was first utilized by [5]. This neural network could estimate distribution probability model of the residual information r_t and R_t, and further guide the entropy coding according to this estimated distribution probability model.

Step 6. **Frame reconstruction.** When the image residual presentation $\hat{y}_t$ is obtained at the decoding side as $\hat{r}_t$, it would be added to $\bar{x}_t$ acquired in step 2 to get the reconstructed current frame $\hat{x}_t$. $\hat{x}_t$ will be saved together with the reconstructed optical flow $\hat{v}_t$ in step 3 and used as references for motion prediction and compensation at $t+1$ moment.

3.2 MV encoder and decoder network

In order to minimize bit-rate, we adopt an auto-encoder to encode optical flow residuals, resulting in considerable bit-rate reduction before entropy coding. Different from motion residuals transmitted in M-LVC, which is the bias between the current optical flow v_t and the predicted optical flow $\bar{v}_t$, instead our method transmits the difference between the current optical flow v_t and the previous optical flow v_{t-1}, as follows:

$$R_t = v_t - \hat{v}_{t-1}, \tag{1}$$

where v_t represents the estimated motion vector, i.e. optical flow, of the t moment by Spynet, and $\hat{v}_{t-1}$ denotes the reconstructed motion vector of $t-1$ moment.

And as the datatype of optical flow is highly similar to that of image, thus we use the image auto-encoder in [4] to compress the optical residuals with little changes. Figure 2 shows overall structure of this network. The residual R_t will be the input of this network to get the latent presentation Y_t, which will later be quantized into $\hat{Y}_t$. After entropy encoding and transmission, de-convolution will be introduced to act on $\hat{Y}_t$, where the reconstructed optical flow residual $\hat{R}_t$ is generated as follows:

$$Y_t = E_{res}(R_t),\ \hat{R}_t = D_{res}(\hat{Y}_t), \tag{2}$$

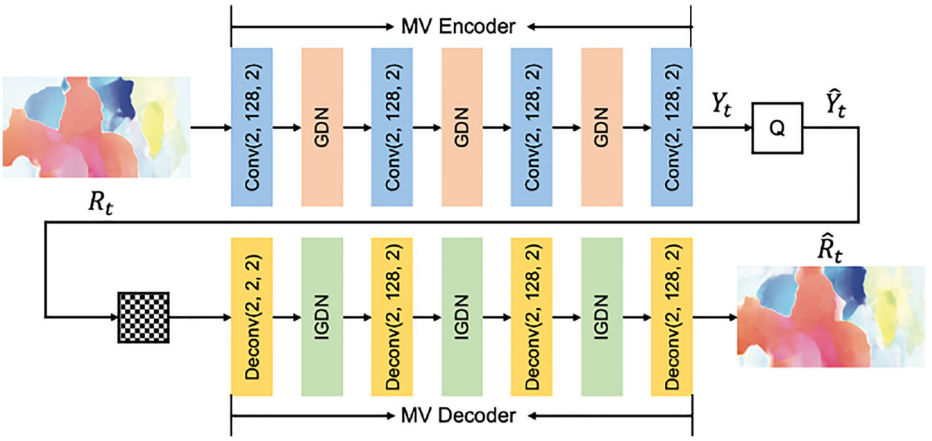

Fig. 2 MV residual auto-encoder network. Conv(3,128,2) represents that the kernel size, output channel and stride of convolution operation is 3x3, 128 and 2 respectively. GDN/IGDN [4] is a nonlinear transform function. Q means quantization

here, $E_{res}(\cdot)$ and $D_{res}(\cdot)$ represent encoder and decoder of our optical flow residual network. $\hat{Y}_t$ is the presentation of motion residual after decoder, and because of quantization operation, some information would be lost during the transmission.

3.3 MV refinement network

Optical flow residual approach has been adopt in motion compression module according to strong inter-frame correlation of surveillance and conference videos. However, the residual information may lead to error accumulation and propagation. What's more, the optical flow given by the motion estimation network is not the real optical information. A certain gap between predicted information and the truth exits. So, in the case that errors be generated during optical flow process, only transmitting optical flow of the first and second video frames are transmitted makes the motion information even more inaccurate. Quantization later would also cause discard of many fine data. For these reasons, we introduce an optical flow optimization module to reduce the errors and improve the overall optical flow quality. The procedure is as follows:

$$\hat{v}_t = Refine(\bar{v}_t) = Refine(\hat{R}_t + \hat{v}_{t-1}), \tag{3}$$

where $\hat{R}_t$ is decoded residual optical flow from previous step. $\bar{v}_t$ presents reconstructed optical flow, obtained by adding $\hat{R}_t$ to the reconstructed motion vector of previous moment $\hat{v}_{t-1}$. $Refine(\cdot)$ denotes the process through MV Refinement Network described in Fig. 3. Refine operation is performed to reduce the errors generated by lossy compression and build optical flow more close to the truth.

The framework of the whole refine network is shown in Fig. 3. This network was first used for optical flow optimization in [3]. Similar to Spynet, this network uses information at different resolutions to generate a high-quality optical flow. A three-level structure of the optical flow refine network is shown in the figure. In addition to the convolutional and upsampling layers, this network uses many residual blocks. After each convolutional layer, different feature will be extracted from $\bar{v}_t$, and each feature layer will be input to the residual blocks to further extract optical flow information at corresponding size. Finally the three

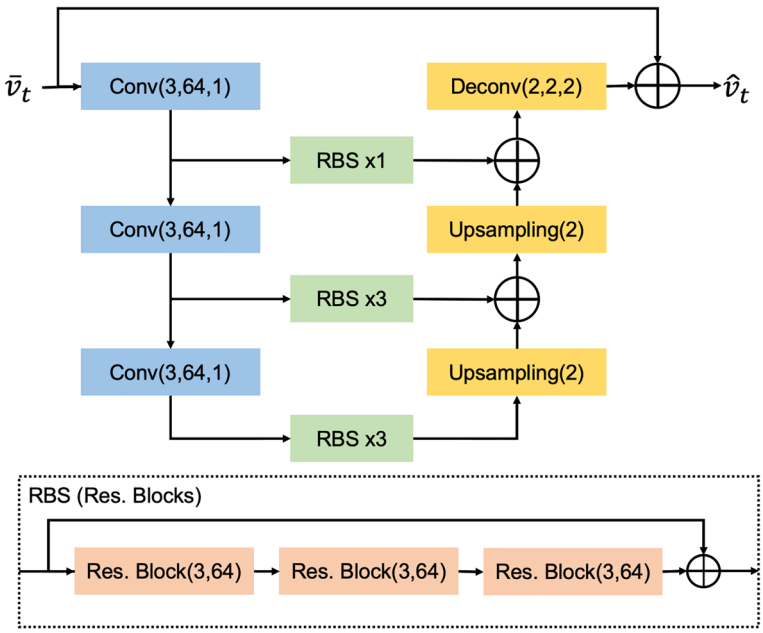

Fig. 3 Our optical flow refine net. (3,64,1) represents that the kernel size, output channel and stride of convolution operation is 3x3, 64 and 1 respectively. RBS x3 means that this net contains 3 Res. Blocks

feature information will be combined together for convolution, and then be summed to the original input, i.e., the unrefined optical flow, to get the optimized optical flow $\hat{v}_t$.

This network feed many fine information to the predicted MV $\bar{v}_t$ through extracting intra-frame information, thus leading to a more accurate optical flow. By using this refined MV, the distortion between the ground frame of next time and the predicted frame would be narrowed, which would also result in the decrease of bits of image residuals. The refine operation, in conclusion, could improve the quality of decoded frame, but also reduce the bit-rate of image residuals.

3.4 Motion compensation network

When v_t from the optical flow estimation network is input to the motion compensation network along with the reconstructed frame $\hat{x}_{t-1}$, the prediction $\bar{x}_t$ of the current frame will be acquired. Since optical flow information is pixel-level momentum information, we believe that $\bar{x}_t$ recovered by high-quality optical flow is closer enough to the original frame x_t, so it would be used as coding reference for image residuals.

In motion compensation, we use the compensation network of DVC. The process is as follows: first, a simple warp operation based on optical flow information is performed on the previous video frame $\hat{x}_{t-1}$ using the predicted optical flow v_t. However, this deformation operation inevitably leads to some artifacts, so we input v_t, $\hat{x}_{t-1}$ and the warped image together into a CNN. This network can reduce artifacts to a certain extent and can return a more detailed reconstruction $\bar{x}_t$:

$$\bar{x}_t = C(\hat{x}_{t-1} + v_t + W(\hat{x}_{t-1} + v_t)), \tag{4}$$

where, $C(\cdot)$ denotes the motion compensation operation and $W(\cdot)$ presents the warping process.

3.5 Residual encoder and decoder network

After motion compensation, $\bar{x}_t$ will be used as the reference for image residual and the real image x_t will be subtracted from it to obtain the image residual r_t. We then use the codec network proposed in [5] for image residual coding, which is a highly nonlinear network that can convert the image residuals into corresponding potential expressions before quantization and entropy coding. This network can also generate the corresponding probability distributions based on the input data. The procedures can be presented as follows:

$$y_t = E'_{res}|_{\hat{\sigma}}(r_t),\ \hat{r}_t = D'_{res}|_{\hat{\sigma}}(\hat{y}_t), \tag{5}$$

$$z_t = H(y_t),\ \hat{\sigma} = H'(\hat{z}_t), \tag{6}$$

similarly, $E'_{res}(\cdot)$, $D'_{res}(\cdot)$ denote the encoding and decoding ends of image residual encoder, and $\hat{y}_t$ is the potential expression of the residual. Here $H(\cdot)$, $H'(\cdot)$ denote the encoder and decoder in the entropy coding process. When the potential information is passed through this auto-encoder, $\hat{\sigma}$ will be output to guide the entropy coding of the image residuals.

The network of image residual auto-encoder is the same as the optical flow residual codec aforementioned, except that the number of channels is set differently due to different data types. Compared with optical residual codec, the biggest difference is the probability distribution generating network, whose main function is to generate the probability model for arithmetic entropy coding to get $\hat{y}_t$.

3.6 Online encoder updating scheme

To further improve the recovery quality of video frames, we also introduce an online update strategy. Since in practical cases, all truth information is available during the encoding process while the decoder is controlled by users and does not allow for real-time synchronous updates. Therefore, our online update strategy is carried out for encoder, which is similar to [18]. The specific process is as follows.

We first complete training of overall network based on training set to obtain a global optimal solution. The trained model will be imported into the network during the testing and all parameters except those of two auto-encoders are fixed. Before actually encoding frames of a GOP, we train the network using the test sequence, and do an update process to the parameters of encoders using backward propagation. This online-update method could decrease the value of loss function in this GOP. For each frame, we record the value of the loss function and judge whether to continue online updating based on the fluctuation of loss function's value. When the value of recent three or more consecutive frames does not fluctuate more than 3% of the previous frame's, the loss is considered to be stabilized and the online update would not be continued for this GOP. For each GOP, the number of frames updated is less than or equal to the size of this GOP. Until the update of encoders in this GOP is completed, the formal coding process of this GOP will start.

In the online update process, we pursue the local optimal solution in this GOP, and reduce the value of loss function. By updating the encoder networks' parameters through training on the test sequences, this local optimal solution can be approached, thus improving the test results on this video sequence.

3.7 Training strategy

Loss function In order to achieve the overall end-to-end compression result for surveillance and conference videos, we need to reduce the bitstreams, but also to simultaneously reduce the differences between the reconstructed frames and the real frames, improving overall image quality of the decoded video frames. Therefore, we adopt the following loss function:

$$L = \lambda D + R = \lambda d(x_t, \hat{x}_t) + (f(\hat{y}_t) + f(\hat{Y}_t)), \tag{7}$$

here, $d(x_t, \hat{x}_t)$ denotes the difference between the real frame x_t and the reconstructed frame $\hat{x}_t$, and for simple calculation of the loss function and subsequent PSNR values, we use the mean squared error (MSE) here. The $f(\hat{y}_t)$ and $f(\hat{Y}_t)$ present the number of bits required during the transmission of image residuals and optical flow residuals after entropy coding. λ, on the other hand, is a multiplier that determines the trade-off the number of bits and distortion.

Quantization The image residual representation y_t and the optical flow residual representation Y_t need to be quantized before entropy coding to further reduce bit-rates. However, traditional quantization methods cannot perform differentiation operations, which means that the entire network cannot be back-propagated if traditional quantization methods are used. To enable the network to be trained end-to-end, we adopt the quantization method in [4]: replacing traditional quantization operation by adding uniform noise. And in the testing stage, rounding operation is performed directly.

Step by step training strategy The training of neural networks plays a very important role for networks' effectiveness. Especially for deep neural networks such as end-to-end video codec, the training details can often determine whether a network success. Therefore, based on difficulties encountered in our experiments and training, we elaborate step-by-step training strategy for our network. Firstly, train the optical flow residual codec network, where the optical flow prediction network, i.e. the Spynet, uses a pre-trained model, and its parameters need to be fixed during training. After training the optical flow residual auto-encoder, we add the motion compensation network for training, and the parameters of the Spynet and optical flow residual codec should be fixed. When the above training and overall fine-tuning are completed, the image residual codec network would be then trained. Similarly, other parameters need to be immutable to prevent deterioration of whole network. Finally, the optical flow refinement module would be added to the overall training until the whole network is successfully trained. During the whole training process, we adopt the same strategy as M-LVC to use the same loss function, and do not vary the loss functions for different modules [23, 39].

4 Experiments

4.1 Experimental setup

Datasets The training set we use is Vimeo-90k dataset [37], and when loading the video frames we scale the image into 448x256 size. Since our network aims at surveillance and conference videos, the test sets we use are HEVC standard test sequences (Class A, Class E) [28], CUHK Square surveillance dataset [33] and Ewap surveillance dataset (Ewap eth and Ewap hotel) [21].

Evaluation method We use PSNR to measure degree of distortion between the reconstructed frames and the real frames. The number of bits is measured by the number of bits per pixel (bpp) to represent the required bits to encode each pixel.

Implementation details For the first frame information and the first optical flow, we encode them using H.265. And different encoding rates are adopted, we use different λ parameters of 256, 512, 1024, 2048 for our loss function. The optimizer is the Adam optimizer, and we set the initial learning rate lr to 0.0001, β_1 to 0.9, and β_2 to 0.999. During the training, the learning rate is reduced as the loss function's value stabilize. The Batch size is set to 4 and the whole system is implemented based on PyTorch.

4.2 Experimental results

We include H.265 and H.264 for a more comprehensive comparison. For H.265 and H.264 encoding settings, we use the very fast mode in FFmpeg. And for HEVC Class A, Class E, CUHK Square and Ewap eth, Ewap hotel, we adopt the same GOP value, which is set to 10.

Figure 4 shows the test results on test sequences. The line graph shows that our network achieves a significant improvement in surveillance conference video. On the CUHK

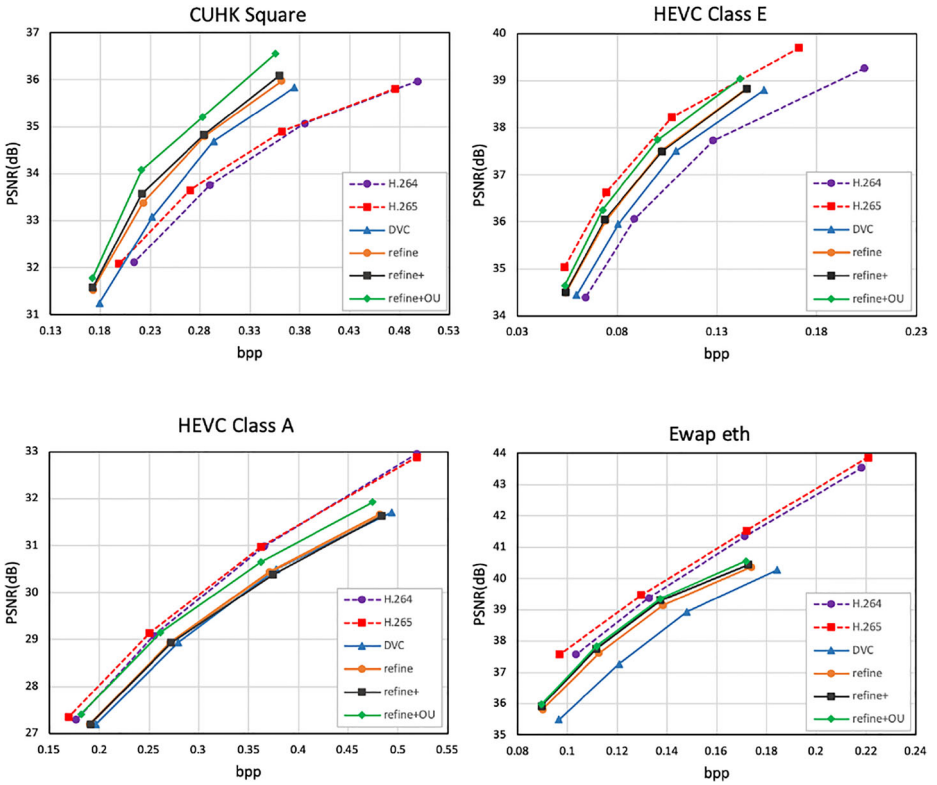

Fig. 4 Overall performance on CUHK Square, HEVC Class E, HEVC Class A and Ewap eth datasets. The compared methods are H.264, H.265, DVC, our proposed method without online updating strategy (refine), the method where the first optical flow frame is also refined by our network (refine+) and the method applying online updating strategy to "refine+"(refine+OU)

Square dataset, we achieved 1.2dB improvement compared to DVC at the same bpp value. On the HEVC Class E conference video sequence, an average improvement of 0.5dB was made. And for other videos, our framework could all outperform DVC. It is obvious that our strategies for surveillance and conference videos can better obtain the spatio-temporal information of video sequences, thus improving the coding quality and reducing the bit-rate.

We also carried out experiments on other datasets. The comparison is shown in Fig. 5. It can be seen that our network without online updating Strategy could also have the same performance as DVC. With the proposed online updating, the network could have adaptability to the compressed video frames, improving the compression performance of the following frames and outperform DVC.

In Ewap hotel, our network and DVC are both lightly worse than H.265 (as shown in Fig. 5). The reason is that DVC framework, i.e. alignment based end-to-end method, still has limit when dealing with extremely large movement. As shown in Fig. 6, a running train appears in the Ewap hotel dataset, resulting in massive MV information.

However, on most of the tested sequences, our method achieves better outcomes than DVC, while on some surveillance and conference videos, H.265 and H.264 can still achieve higher PSNR than ours.

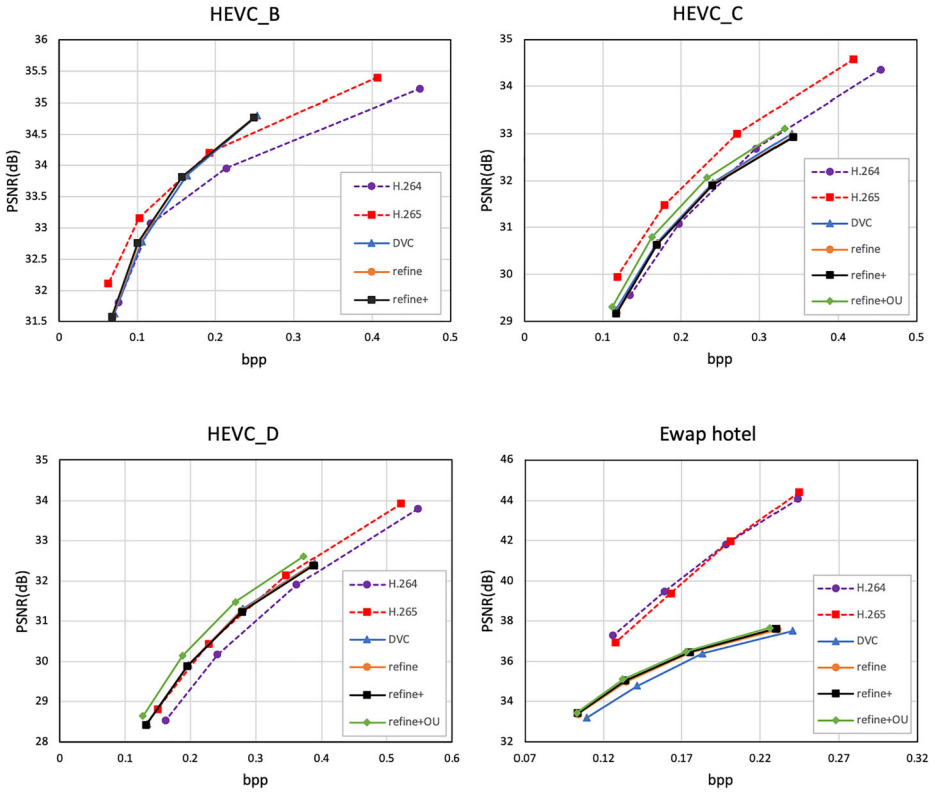

Fig. 5 Network performance on HEVC Class B, C, D and Ewap hotel datasets. The compared methods are H.264, H.265, DVC, refine (Our proposed method without online updating strategy), refine+ (the first optical flow frame is also refined by our network) and refine+OU (adding online updating strategy to "refine+")

Fig. 6 One frame of Ewap hotel datasets. Ewap hotel dataset has a running train in the top of these pictures, which resulting in tremendous movement, i.e., huge MV information

4.3 Ablation study and model analysis

MV of surveillance and conference videos Surveillance and conference videos own a high degree of separation between foreground and background, which leads to a weak connection between objects contained. This kind of videos also has strong inter-frame correlation, a feature that results in coherent motion information. In addition to the relatively fixed background and robust inter-frame correlation, the moving objects in surveillance and conference videos also have a more consistent motion pattern when divided by category. In our method, we adopt optical flow residuals to encode motion information in view of the strong inter-frame correlation. To further verify the effectiveness of our method, we do some extra experiments on CUHK Square surveillance video sequences and visualize the experimental process in Fig. 7, and the original frame is given together.

Figure 7a shows the visualization of original optical flow, and it can be observed that there are three moving people within the range. Figure 7b is the predicted optical flow image by Spynet, and it is obvious that in addition to some background noise there are three areas of blocks, which correspond to the motion information of these three people respectively. And in the visualized image of the optical flow residual, the background noise witnessed a

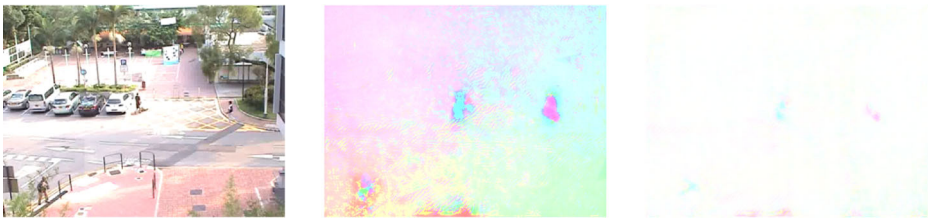

Fig. 7 Visualization of optical flow and optical flow residual. (a) The original frame x_t. (b) The predicted MV V_t. (c) The MV residual R_t

decrease. Besides, the three human regions sims to have a de-filling operation, the color in the middle of these regions is basically close to white, that is, the MV information in the middle of these region is close to 0, while the edges of the region preserve the original data. This change can effectively reduce the bit-rate of MV during transmission. Especially for larger moving objects, a considerable part of the MV can be offset by the subtraction, which could also eliminate repeated transmission, thus reducing the transmission bit-rate.

In addition to the visualization of optical flow and optical flow residual, we also processed the bpp required to transmit optical flow and optical flow residuals for different λ parameters. All the data is from CUHK Square surveillance video sequences, as shown in Table 1.

It can be concluded that the application of residual is effective in reducing the bit-rate and can achieve an average bpp reduction of 37.7%.

Analysis of MV refinement In our network, the image reconstruction depends on the previous frame and optical flow information, therefore, the accuracy of the MV plays a very critical role in subsequent image recovery. Although the application of MV residuals can effectively reduce the bit-rate of optical flow, errors would also be introduced, leading to a certain degree of deformation and distortion. So when the optical flow residuals introduce relatively large errors, the image reconstruction will deteriorate, which would in turn increase the bit-rate of the image residuals during transmission.

Thus, it is indispensable to finely optimize the optical flow using an refinement network after transmission. For the refinement network used adopted by us, we designed additional experiments to verify its effectiveness. The following experiments are still performed on the CUHK Square surveillance video dataset.

For transmission without MV refinement ("W/O refine"), we can see in Fig. 8 that the PSNR faces a reduction of about 0.5 dB at the same bpp compared to "refine+", and the overall effect is lower than that of DVC. In order to verify the necessity of the combination of optical flow residual and optical flow refinement, we added the optical flow optimization network to the DVC. As can be seen from the lines of "DVC" and "DVC+refine" in Fig. 8, the addition of MV refinement network does not improve the overall coding effect, but reduces the quality of the reconstructed images. This indicates that the MV refinement network does not work for frameworks that transmits complete MV information, and also confirms the effectiveness of "MV residual + MV refinement" for surveillance and conference videos.

Online update strategy Since the encoder has access to all the truth information, this makes the online update strategy possible. For users, if the parameters of the decoding side are also involved in the online update, it will make the testing process of the network more

Table 1 Comparison of MV and MV residuals

CRF	Bpp of optical flow	Bpp of residuals	Decrease
20	0.027883	0.018120	35.01%
23	0.024189	0.015300	36.75%
26	0.020322	0.012347	39.24%
29	0.016336	0.009841	39.76%
AVG	0.221825	0.013902	37.69%

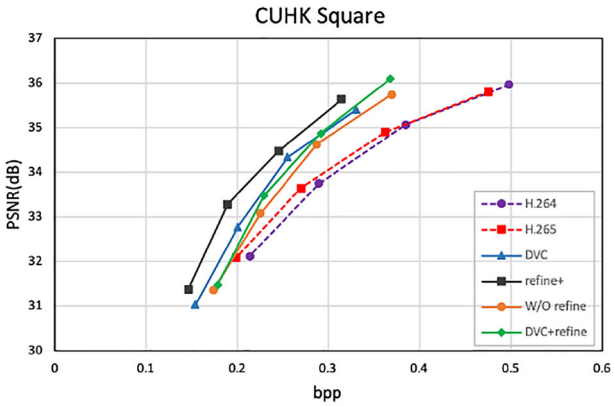

Fig. 8 Ablation experiments of MV refinement network. To verify the effectiveness of our method, comparisons are done, with H.264, H.265, DVC, Our method(refine+), W/O refine(MV refinement is not adopted) and DVC+refine(MV refine strategy is applied to DVC and the model used for MV Refinement Network is the same as "refine+")

tedious and require transmission of the updated parameters, which is contrary to the original purpose of reducing the bit-rate, so in this module we only perform the update for the encoding side.

The final results of the tests using the online update method have been shown in Fig. 4 ("refine+OU"), and it can be seen that the online update strategy achieved significant results on all these datasets. Compared to our network without online update("refine+"), the method with online update mechanism obtained about 0.5dB improvement on HEVC Class A and CUHK Square sequences. For conference video HEVC Class E, it also achieved about 0.25dB increase in PSNR at the same bit-rate.

5 Conclusion

We proposed an end-to-end video codec network for surveillance and conference videos. By utilizing the nonlinearity of convolutional neural networks and residual networks, our network is able to absorb the advantages of traditional video codecs while effectively reducing the coding bit-rate and enabling the implementation and training of end-to-end compression. We also introduce a "MV residuals + MV refinement" strategy to further reduce the number of bits for surveillance and conference videos while ensuring the image quality of decoded video frames. In the testing stage, the use of online update for encoders could guarantee local optimal solution for different videos. Experimentally, our approach proved to exceed the previously proposed DVC compression framework. In the field of surveillance and conference videos compression, we expect that the coding efficiency can be further improved by different alignment methods or by methods that avoid transmitting too much motion information.

Acknowledgements This work was supported in part by National Key R&D Program of China (2018YFE0203900), National Natural Science Foundation of China (61773093), Sichuan Science and Technology Program (2020YFG0476) and Important Science and Technology Innovation Projects in Chengdu (2018-YF08-00039-GX).

References

1. Agustsson E, Mentzer F, Tschannen M, Cavigelli L, Timofte R, Benini L, Gool LV (2017) Soft-to-hard vector quantization for end-to-end learning compressible representations. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R. (eds) Advances in neural information processing systems, vol 30. Curran Associates, Inc
2. Alam MM, Nguyen TD, Hagan MT, Chandler DM (2015) A perceptual quantization strategy for HEVC based on a convolutional neural network trained on natural images. In: Applications of digital image processing XXXVIII, vol 9599, p 959918. International Society for Optics and Photonics
3. Alexandre D, Hang HM (2020) Learned video codec with enriched reconstruction for clic p-frame coding. arXiv:2012.07462
4. Ballé J, Laparra V, Simoncelli EP (2017) End-to-end optimized image compression. In: International conference on learning representations
5. Ballé J, Minnen D, Singh S, Hwang SJ, Johnston N (2018) Variational image compression with a scale hyperprior. In: International conference on learning representations
6. Bellard F BPG image format (http://bellard.org/bpg/), Accessed 30 Jan 2017
7. Cui W, Zhang T, Zhang S, Jiang F, Zuo W, Zhao D (2018) Convolutional neural networks based intra prediction for HEVC, pp 436–436
8. Djelouah A, Campos J, Schaub-Meyer S, Schroers C (2019) Neural inter-frame compression for video coding. In: Proceedings of the IEEE/CVF international conference on computer vision, pp 6421–6429
9. Hu Z, Lu G, Xu D (2021) FVC: a new framework towards deep video compression in feature space. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 1502–1511
10. Huo S, Liu D, Wu F, Li H (2018) Convolutional neural network-based motion compensation refinement for video coding. In: 2018 IEEE International symposium on circuits and systems (ISCAS), pp 1–4
11. Index CVN (2016) Forecast and methodology, 2015–2020. White paper, 1–41
12. Johnston N, Vincent D, Minnen D, Covell M, Singh S, Chinen T, Hwang SJ, Shor J, Toderici G (2018) Improved lossy image compression with priming and spatially adaptive bit rates for recurrent networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4385–4393
13. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. Adv Neur Inform Process Syst 25:1097–1105
14. LeCun Y, Bottou L, Bengio Y, Haffner P (1998) Gradient-based learning applied to document recognition. Proc IEEE 86(11):2278–2324
15. Li J, Li B, Xu J, Xiong R, Gao W (2018) Fully connected network-based intra prediction for image coding. IEEE Trans Image Process 27(7):3236–3247
16. Lin J, Liu D, Li H, Wu F (2020) M-LVC: multiple frames prediction for learned video compression. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 3546–3554
17. Lu G, Ouyang W, Xu D, Zhang X, Cai C, Gao Z (2019) DVC: an end-to-end deep video compression framework. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, pp 11006–11015
18. Lu G, Cai C, Zhang X, Chen L, Ouyang W, Xu D, Gao Z (2020) Content adaptive and error propagation aware deep video compression. In: European conference on computer vision, pp 456–472. Springer
19. Marpe D, Schwarz H, Wiegand T (2003) Context-based adaptive binary arithmetic coding in the h. 264/avc video compression standard. IEEE Trans Circ Syst Video Technol 13(7):620–636
20. Minnen D, Ballé J, Toderici G (2018) Joint autoregressive and hierarchical priors for learned image compression. In: Bengio S, Wallach H, Larochelle H, Grauman K, Cesa-Bianchi N, Garnett R (eds) Advances in neural information processing systems, vol 31. Curran Associates, Inc
21. Pellegrini S, Ess A, Schindler K, Van Gool L (2009) You'll never walk alone: modeling social behavior for multi-target tracking. In: 2009 IEEE 12th International conference on computer vision, pp 261–268
22. Ranjan A, Black MJ (2017) Optical flow estimation using a spatial pyramid network. In: Proceedings of the IEEE Conference on computer vision and pattern recognition, pp 4161–4170
23. Reda FA, Liu G, Shih KJ, Kirby R, Barker J, Tarjan D, Tao A, Catanzaro B (2018) Sdc-net: video prediction using spatially-displaced convolution. In: Proceedings of the European conference on computer vision (ECCV), pp 718–733
24. Sengar SS, Mukhopadhyay S (2020) Motion segmentation-based surveillance video compression using adaptive particle swarm optimization. Neural Comput Applic 32(15):11443–11457
25. Skodras A, Christopoulos C, Ebrahimi T (2001) The jpeg 2000 still image compression standard. IEEE Signal Process Mag 18(5):36–58
26. Song R, Liu D, Li H, Wu F (2017) Neural network-based arithmetic coding of intra prediction modes in HEVC. In: 2017 IEEE Visual communications and image processing (VCIP), pp 1–4

27. Song X, Chen Y, Feng ZH, Hu G, Yu DJ, Wu XJ (2020) SP-GAN: self-growing and pruning generative adversarial networks. IEEE Trans Neural Netw Learn Syst 32(6):2458–2469
28. Sullivan GJ, Ohm JR, Han WJ, Wiegand T (2012) Overview of the high efficiency video coding (HEVC) standard. IEEE Trans Circ Syst Video Technol 22(12):1649–1668
29. Theis L, Shi W, Cunningham A, Huszár F (2017) Lossy image compression with compressive autoencoders. In: International conference on learning representations
30. Toderici G, O'Malley SM, Hwang SJ, Vincent D, Minnen D, Baluja S, Covell M, Sukthankar R (2016) Variable rate image compression with recurrent neural networks. In: International conference on learning representations
31. Toderici G, Vincent D, Johnston N, Jin Hwang S, Minnen D, Shor J, Covell M (2017) Full resolution image compression with recurrent neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 5306–5314
32. Wallace GK (1992) The jpeg still picture compression standard. IEEE Trans Consum Electron 38(1):xviii–xxxiv
33. Wang M, Li W, Wang X (2012) Transferring a generic pedestrian detector towards specific scenes. In: 2012 IEEE Conference on computer vision and pattern recognition, pp 3274–3281
34. Wu CY, Singhal N, Krahenbuhl P (2018) Video compression through image interpolation. In: Proceedings of the European conference on computer vision (ECCV), pp 416–431
35. Wu Y, He T, Chen Z (2020) Memorize, then recall: a generative framework for low bit-rate surveillance video compression. In: 2020 IEEE International symposium on circuits and systems (ISCAS), pp 1–5
36. Wu L, Huang K, Shen H, Gao L (2021) Foreground-background parallel compression with residual encoding for surveillance video. IEEE Trans Circuits Syst Video Technol 31(7):2711–2724
37. Xue T, Chen B, Wu J, Wei D, Freeman WT (2019) Video enhancement with task-oriented flow. Int J Comput Vis 127(8):1106–1125
38. Yan N, Liu D, Li H, Li B, Li L, Wu F (2018) Convolutional neural network-based fractional-pixel motion compensation. IEEE Trans Circuits Syst Video Technol 29(3):840–853
39. Yang R, Xu M, Wang Z, Li T (2018) Multi-frame quality enhancement for compressed video. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 6664–6673
40. Zhao L, Wang S, Wang S, Ye Y, Ma S, Gao W (2021) Enhanced surveillance video compression with dual reference frames generation. IEEE Trans Circuits Syst Video Technol, 1–1