Deep Learning-Based Chroma Prediction for Intra Versatile Video Coding

Linwei Zhu, Yun Zhang, *Senior Member, IEEE,* Shiqi Wang, *Member, IEEE,* Sam Kwong, *Fellow, IEEE,*
Xin Jin, *Senior Member, IEEE,* and Yu Qiao, *Senior Member, IEEE*

Abstract—**Color images always exhibit a high correlation between luma and chroma components. Cross component linear model (CCLM) has been introduced to exploit such correlation for removing redundancy in the on-going video coding standard, i.e., versatile video coding (VVC). To further improve the coding performance, this paper presents a deep learning based intra chroma prediction method, termed as convolutional neural network based chroma prediction (CNNCP). More specifically, the process of chroma prediction is formulated to produce the colorful version from available information input. CNNCP includes two sub-networks for luma down-sampling and chroma prediction, which are jointly optimized to fully exploit spatial and cross component information. In addition, the outputs of CCLM are adopted as chroma initialization for performance enhancement, and the coding distortion level characterized by quantization parameter is fed into the network to release the negative affect from compression artifacts. To further improve the coding performance, the competition is performed between the conventional chroma prediction and CNNCP in terms of rate-distortion cost with a binary flag signalled. The learned CNNCP is incorporated into both video encoder and decoder. Extensive experimental results demonstrate that the proposed scheme can achieve 4.283%, 3.343%, and 4.634% bit rate savings for luma and two chroma components, compared with the VVC test model version 4.0 (VTM 4.0).**

Index Terms—**Deep learning, convolutional neural network, chroma prediction, versatile video coding.**

I. Introduction

RECENT years have witnessed wide applications of videos in various fields, such as entertainment, security surveillance, and Virtual Reality (VR). The evolution of videos in the past three decades can be summarized from the following aspects, i.e., High Definition (HD), High Frame Rate (HFR), Multi-View Video (MVD) [1], High Dynamic Range (HDR) [2] and Wide Color Gamut (WCG) [2]. The explosively increasing video data, which pose a great challenge to the data storage and transmission, require advanced video coding algorithms to reduce the data volume and maintain the video quality.

Although a couple of video coding standards have been issued in the past 30 years, such as H.264/Advanced Video Coding (AVC) [3] and High Efficiency Video Coding (HEVC) [4], the compression ratio still cannot catch the increasing of video data. In April 2018, the new standard was formally named as Versatile Video Coding (VVC) [5] by Joint Video Experts Team (JVET), which aims to adapt to new applications, such as HDR and VR, and improve the performance of its predecessor, i.e., HEVC.

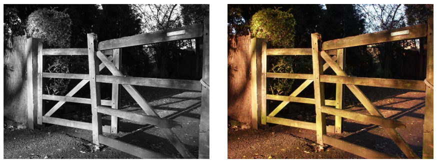
Regarding VVC, the coding performance has been significantly improved when compared to HEVC. Almost all the modules in VVC have been improved with new coding tools, including the block partition, intra prediction, motion estimation/componsetion, transform. In particular, for the block partition, the default Coding Tree Unit (CTU) has been enlarged from 64×64 to 128×128 [5]. Besides the quad-tree partition, the binary and ternary tree partitions are included as well [5]. More specifically, the quad-tree partition is firstly performed, and then each node of quad-tree can be further partitioned by binary tree with two identical units or by ternary tree with three units. Moreover, horizontal and vertical directions are performed for the binary and ternary tree partitions. For the intra prediction, 32 more angular modes are included for luma component, which are used to adapt to the diverse contents and different block sizes. In addition, multi-line neighboring pixels are utilized as reference for prediction [6]. For the motion estimation/compensation, many techniques are employed. The coding tool of Intra Block Copy (IBC) [7], which is only used for HEVC Screen Content Coding (SCC) extension, has been adopted in VVC. Besides the traditional motion compensation, Affine Motion Compensation (AMC) methods with 4-parameter and 6-parameter [8] are utilized to handle the motion of rotation, zooming, and shearing. In addition, Combined Inter/Intra Prediction (CIIP) [9] and Triangle Prediction Mode (TPM) [9] are proposed to improve the prediction. Regarding the transform, the maximum transform size has been upgraded to 128×128 for luma component as the increased size of CTU. With multiple transform types of DCT/DST, three transform

Manuscript received June 20, 2020; revised September 03, 2020, and October 10, 2020; accepted October 23, 2020. Date of publication XX XX, XXXX; date of current version XX XX, XXXX. This work was supported in part by Shenzhen Science and Technology Program under Grant JCYJ20180507183823045, in part by the Natural Science Foundation of China under Grant 61901459 and Grant 61672443, in part by China Postdoctoral Science Foundation under Grant 2019M653127, in part by Guangdong International Science and Technology Cooperative Research Project under Grant 2018A050506063, in part by Membership of Youth Innovation Promotion Association, Chinese Academy of Sciences under Grant 2018392, in part by Hong Kong RGC General Research Funds under Grant 9042816 (CityU 11209819), and in part by Hong Kong RGC Early Career Scheme under Grant 9048122 (CityU 21211018). This article was recommended by Associate Editor Prof. Lu Yu. *(Corresponding author: Yun Zhang.)*

L. Zhu, Y. Zhang and Y. Qiao are with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China (e-mail: lw.zhu@siat.ac.cn; yun.zhang@siat.ac.cn; yu.qiao@siat.ac.cn).

S. Wang and S. Kwong are with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong and also with the City University of Hong Kong Shenzhen Institute, Shenzhen 518057, China (e-mail: shiqwang@cityu.edu.hk; cssamk@cityu.edu.hk).

X. Jin is with the Shenzhen Key Laboratory of Broadband Network and Multimedia, Graduate School at Shenzhen, Tsinghua University, Shenzhen 518055, China (e-mail: jin.xin@sz.tsinghua.edu.cn).



(a) Gray image (b) Color image

Fig. 1. Gray and color images.

methods are designed, i.e., intra sub-partitioning [10], sub-block transform [10] and shape adaptive implicit transform selection [10].

In recent years, deep learning based video coding has become more and more popular because of its promising performance. The modules in video coding, including inter/intra prediction, motion compensation, and in-loop filtering/post-processing, have been improved with different neural networks [11][12]. For the module of inter/intra prediction, an enhanced bi-prediction scheme [13] was presented with Convolutional Neural Network (CNN) to improve the coding performance, which can directly infer the predictive signals in a data driven manner. An efficient inter prediction scheme by introducing the deep virtual reference frame was proposed in [14], which serves as the reference in the temporal redundancy removal process. Li *et al*. [15] proposed a deep learning method for intra prediction, where a fully connected network was learned for an end-to-end mapping from neighboring reconstructed pixels to the to-be-coded block. The process of intra prediction was modeled as an inpainting task in [16], where the Generative Adversarial Network (GAN) was adopted to intelligently remove the spatial redundancy. In [17], the chroma prediction was improved in HEVC with the hybrid neural network of CNN and fully connected network. For the module of motion compensation, the fractional-pixel motion compensation was formulated as an inter-picture regression problem for both uni-directional and bi-directional motion compensations in video coding [18], where the CNN model was adopted. In [19], a one-for-all fractional interpolation method was presented with a Grouped Variation CNN (GVCNN), which can handle different Quantization Parameter (QP) settings and generate all sub-pixel positions at one sub-pixel level. For the module of in-loop filtering/post-processing, a novel quality enhancement method was presented by using a Multi-reconstruction Recurrent Residual Network (MRRN) [20], where a modified recursive residual structure was designed to capture the multi-scale similarity of compression artifacts. Jia *et al*. [21] designed a structure of CNN model from multiple dimensions for loop filtering. Each CTU was treated as an independent region, such that the content-aware multi-model filtering mechanism was performed with different CNN models for different regions.

In this paper, we focus on the chroma prediction with neural network to further remove the redundancy in the YCbCr color space. The main contributions of this paper are listed as follows.

1) The procedure of chroma prediction is formulated to transfer from gray version to colorful version with available information input, in which the CNN model termed as CNN based Chroma Prediction (CNNCP) is utilized, such that the current to-be-coded chroma block can be better predicted.
2) Two sub-networks are equipped in CNNCP, i.e., luma down-sampling and chroma prediction, and they are jointly performed for fully exploiting the spatial and cross component information. In addition, the results of CCLM are adopted as chroma initialization, and the coding distortion level characterized by QP is fed into the network to eliminate the negative affect from compression artifacts.
3) Combining with the conventional and CNNCP, the video encoder and decoder are redesigned. The Rate-Distortion Optimization (RDO) is performed to select the better one between the conventional chroma prediction and CNNCP with one additional flag signalled to the decoder.

The remainder of this paper is organized as follows. Section II introduces the related works. The motivation and problem formulation are described in Section III. Section IV presents the proposed CNNCP for intra coding. The experimental results and analyses are discussed in Section V. Section VI concludes this paper.

II. Related Works

A. Image Colorization

Image colorization is a classical problem in the field of computer vision, which aims to make it colorful from the gray version, as shown in Figs. 1(a) and 1(b). In [22], a colorization based coding method was proposed for image compression, where at the decoder side the chroma pixels could be directly reconstructed by the colorization method. Bugeau *et al*. [23] modeled the problem of image colorization as selecting the best color from a set of color candidates and a variational approach was proposed. In [24], an example based image colorization method was presented by exploiting a new locality consistent sparse representation. A luma-guided diffusion based colorization framework in the YCbCr space was proposed [25], making it a valuable tool for compression. Given a reference color image and a destination grayscale image, an automatic colorization algorithm [26] was proposed to transfer color information from the reference image to the destination image.

The performance of image colorization has been significantly improved with deep learning. Cheng *et al*. [27] ensembled multiple neural networks to obtain better performance than an individual one. In [28], a fully automatic image colorization method using deep neural networks was presented, which aims to minimize users' effort and the dependence on the reference color images. Combining global priors and local priors, the CNN based image colorization architecture [29] was designed, where the training loss was jointly represented by colorization loss and classification loss, aiming to adapt to the contents. In [30], a recurrent framework was presented to guide the colorization of every frame in a video from a given reference image.

Most of these image colorization methods are only performed with grayscale information, leading to the fact that

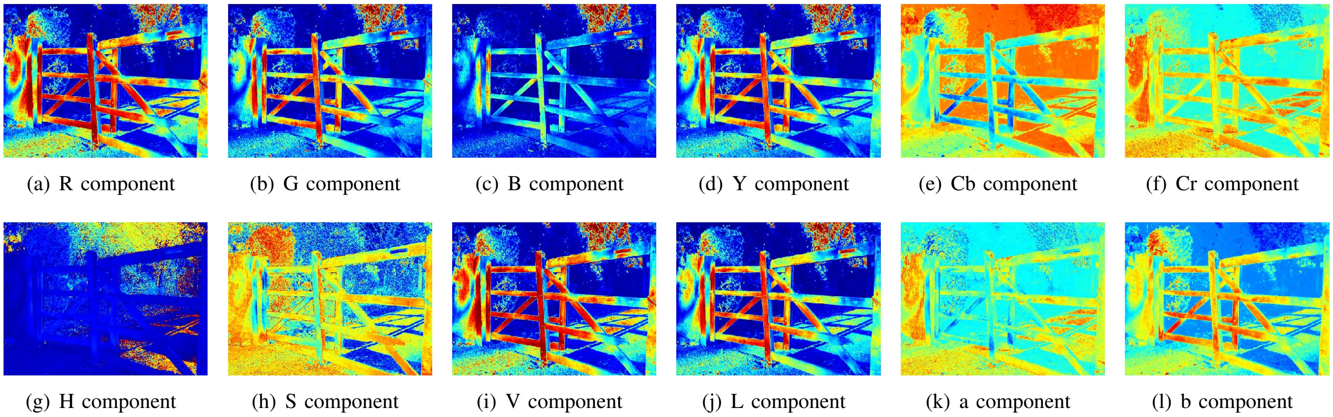

(a) R component (b) G component (c) B component (d) Y component (e) Cb component (f) Cr component
(g) H component (h) S component (i) V component (j) L component (k) a component (l) b component

Fig. 2. Fig. 1(b) represented in RGB, YCbCr, HSV, and Lab color spaces. The individual components of each color space are represented in jet colormap.

any visually pleasing outputs could be acceptable. However, it is challenging for video coding which aims to deliver high fidelity visual signals. As such, it is desirable to train an appropriate colorization model in the video coding domain of applications.

B. Chroma Prediction

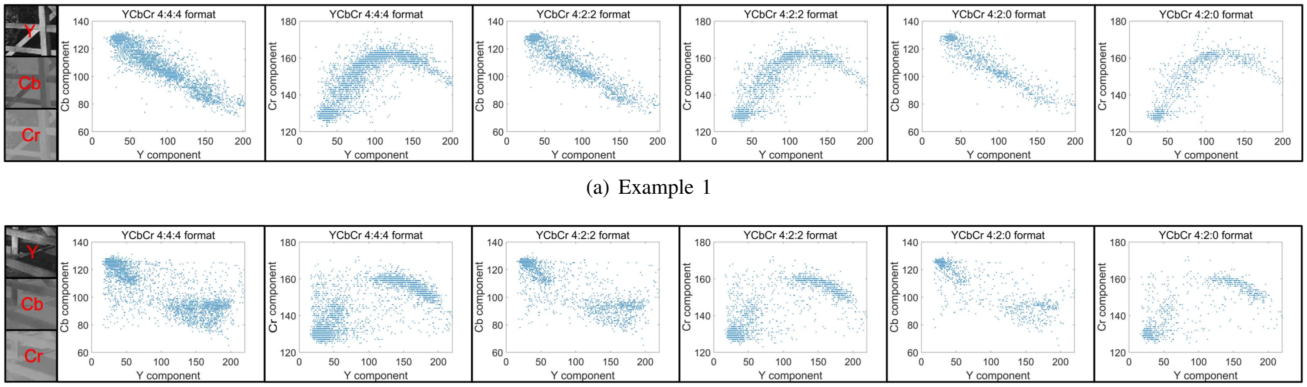
As shown in Figs. 2(a) - 2(l), the correlations in RGB, YCbCr, HSV, and Lab color spaces are illustrated. The individual components of each color space are represented in jet colormap for better visualization. We can easily observe that these three components in every color space are highly correlated. Such redundancy can be further exploited for performance improvement in the field of data compression.

To remove the redundancy in the YCbCr color space, the Cross Component Linear Model (CCLM) [31] was derived with the hypothesis that the pixels of luma and chroma in a coding block can be represented by a simple linear function. Furthermore, several methods are presented to improve the performance of CCLM, including Multi-Model Linear Model (MMLM) [32], Multi-Filter Linear Model (MFLM) [32], and Multi-Directional Linear Model (MDLM) [33]. They all achieve significant coding gains. Additionally, two chroma components (Cb and Cr) can also be represented by another linear model. In VVC, the CCLM has been adopted for chroma prediction. The two parameters of CCLM model are derived from the neighboring reconstructed luma and chroma pixels. To further improve the performance of CCLM, MDLM has been proposed, which consists of left (MDLM_L) and top (MDLM_T) versions. In analogous to the luma component, the angular prediction is also performed in chroma component. The difference lies in that only limited angular modes are adopted, i.e., Planar, DC, Vertical, and Horizontal. Moreover, Derived Mode (DM) has been used to share the mode of luma component. As a result, there are 8 modes in total for chroma prediction in VVC, including Planar, DC, Vertical, Horizontal, DM, CCLM, MDLM_L, and MDLM_T. In addition, dual tree has been used in VVC for intra luma and chroma prediction, which means that the luma and chroma components can have different partition sizes in a CTU.

Most of the existing works concentrate on the cross component redundancy removal with a linear model in case of residual or pixel values. Kim *et al.* [34] presented a Cross Component Prediction (CCP) method, in which the chroma residual signal was predicted from the luma residual signal. This work was extended in [35], such that Cb residual can be employed to predict Cr residual. Khairat *et al.* [36] exploited the correlation between residual components in 4:4:4 format with CCP, and predicted the second and third components from the first component in RGB and YCbCr color spaces. In [37], the template matching was performed for chroma prediction by using the reconstructed luma block. Zhang *et al.* [38] investigated three linear models for the representation of luma and chroma components, which significantly improved the coding performance of HEVC. In [39], chroma from luma prediction was performed in AV1.

Although these linear models, CCLM, MFLM, MMLM, and MDLM, have achieved coding gain to some extent, they are all manually designed, which limit the performance improvement for diverse videos. Sophisticated algorithms are desired to be developed for the representation from luma to chroma.

To further exploit the efficient representation from luma to chroma, the neural networks have been adopted. In [40], the separate networks were utilized to perform luma and chroma prediction, where the neighboring information and cross component information were involved. Blanch et al. [41] presented a neural network architecture for cross component intra prediction, in which an attention module was employed for learning spatial relations. In the literature [17], it presented a hybrid neural network for chroma prediction in HEVC, but it still can be further improved. Different from literature [17], this work has been applied to the next generation video coding standard, i.e., VVC. In particular, the luma down-sampling method with neural network has been considered instead of the traditional method used in [17] to improve the performance of chroma prediction. In addition, the result of CCLM is utilized as chroma initialization and the coding distortion level characterized by QP is included as the network input to eliminate the negative affect of compression artifacts. To fully exploit the spatial and cross component information, the joint luma down-sampling and chroma prediction networks



(a) Example 1

(b) Example 2

Fig. 3. Illustrations of correlations between luma and chroma components in case of YCbCr 4:4:4, 4:2:2, and 4:2:0 formats.

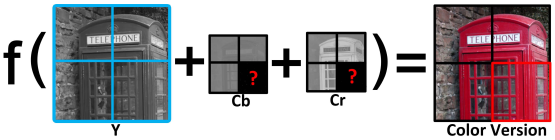


Fig. 4. Problem formulation for two chroma components prediction in case of YCbCr 4:2:0 format. f(·) is a function mapping the available information to the chroma information. The chroma component in the bottom-right of color version is predicted.

are performed.

III. Motivation and Problem Formulation

In video coding, the visual signals are mainly represented in the YCbCr color space. With the hypothesis of linear correlation between luma and chroma components in a coding block, several linear models for chroma prediction have been incorporated into the VVC. Due to the diverse contents, one linear model cannot handle all the cases. Two examples with different YCbCr formats are illustrated in Fig. 3, which indicate that the relationship between luma and chroma components is too complicated to be characterized with a linear model. Two linear models are developed in MMLM [32],

$$C' = \begin{cases} a_1 Y + b_1 & Y \leq T \\ a_2 Y + b_2 & Y > T \end{cases}, \tag{1}$$

where C' is the predicted chroma pixel, Y is the luma pixel, and T is a pre-defined threshold. a_1, b_1 and a_2, b_2 are model parameters. However, for the case of Fig. 3(b), it is still unable to represent this correlation with two linear models. This is a limitation.

Inspired by MMLM, we aim to predict the two chroma components simultaneously with a neural network, where the models are learned in a data-driven manner instead of a hand-crafted way. It is also expected that the networks are equipped with the capacity to learn the natural scene statistics regarding the color information, such that the coding information can be further improved by incorporating the prior information. The spatial information, denoted as local prior, has been adopted. The coding distortion is unavoidable in the lossy encoder, also brings obstacles to perform chroma prediction. Therefore, the impact of compression artifact is supposed to be considered. As shown in Fig. 4, the problem of chroma prediction in case of YCbCr 4:2:0 format is formulated. The to-be-predicted chroma blocks are located at the bottom-right, while the available information are higher resolution of reconstructed luma blocks (cross component information), and the neighboring reconstructed chroma blocks (spatial information). The predicted chroma component, $\mathbf{C}'$, can be formulated by,

$$\mathbf{C}' = f(\mathbf{Y}, \mathbf{C}, \mathbf{D}), \tag{2}$$

where $f(\cdot)$ is a function mapping the available information to the chroma information, $\mathbf{Y}$ is the reconstructed luma component, $\mathbf{C}$ is the chroma component, the above-left, above, and the left sub-blocks are the neighboring reconstructed chroma information, the sub-blocks located at the bottom-right are required to be predicted. Since the available luma and chroma components are distorted, the information of coding distortion level $\mathbf{D}$ is supposed to be included to eliminate the negative affect from compression artifacts. It is worth mentioning this formulation can be employed to other color spaces for data compression.

IV. Proposed CNN based Chroma Prediction for Intra Versatile Video Coding

A. Framework of CNN based Chroma Prediction

The framework of CNNCP is illustrated in Fig. 5. It consists of two neural networks, i.e., luma down-sampling network and chroma prediction network. Since the human visual system is more sensitive to luma than chroma, the YCbCr 4:2:0 format is frequently used in video coding, such that the luma component with size of $4N \times 4N$ is firstly down-sampled by the luma down-sampling network to the size of $2N \times 2N$ as same as the chroma component in this framework. Combining with the coding distortion level and neighboring chroma blocks, the down-sampled luma component will be fed into the chroma prediction network. To make the prediction more accurate, the results of CCLM are generated as the chroma initialization. The outputs of chroma prediction network are two chroma components with size of $2N \times 2N$. The blocks located at the bottom-right will be cropped as the final chroma prediction results with size of $N \times N$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCSVT.2020.3035356, IEEE Transactions on Circuits and Systems for Video Technology

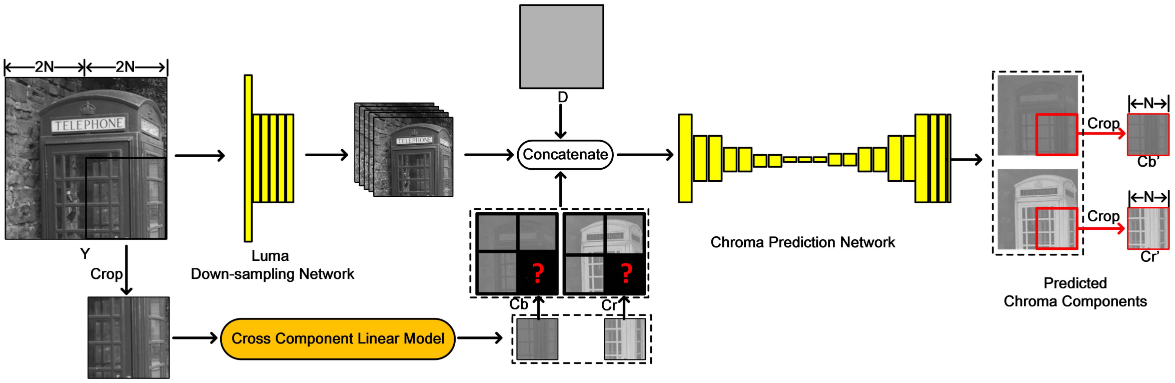


Fig. 5. Framework of proposed Convolutional Neural Network based Chroma Prediction (CNNCP) for YCbCr 4:2:0 format. This framework can also be applied to other YCbCr formats and even other color spaces with associated hyper-parameters.

In this framework, the functions of luma down-sampling and chroma prediction networks are similar to MFLM and MMLM, which are used to produce more down-sampled luma components and mapping models. The difference lies in that these two neural networks are learned from large-scale data, while the MMLM and MFLM are manually designed. To some extent, the chroma initialization with CCLM results will accelerate the training and enhance the prediction performance. The distortion information is characterized by QP, which indicates the level of compression artifacts injected into the available information. It means that the pixel values in the **D** block are filled with the identical QP value. As such, it will eliminate the negative affect of compression artifacts for performing the chroma prediction.

The hyper-parameters of luma down-sampling and chroma prediction networks are illustrated in Tables I and II. For the luma down-sampling network, there are 6 convolutional layers, and their kernel sizes are all 3 × 3. The stride of the second layer is 2 and that of others is 1. Except for the last layer, the outputs of convolutional layers are 16 feature maps and the activation function is ReLU. The output and activation function of last layer are 16 down-sampled luma components and Tanh, respectively. For the chroma prediction network, there are 20 convolutional layers in total. The kernel sizes of them are all 3×3. From 1^{st} to 17^{th} convolutional layers, they output 128 feature maps. For the last layer, the outputs are Cb and Cr components. In summary, we employ the encoder-decoder structure and parameter settings from [42] and [43]. Then, we increase the number of feature maps to 128 to improve learning ability and set batch size as 128 to improve the robustness of model training. Finally, to solve the memory overflow problem while training this model in our GPU, the high level feature maps (18^{th} and 19^{th} layers) are reduced to 16 since they are relatively less important.

TABLE I
HYPER-PARAMETERS OF THE LUMA DOWN-SAMPLING NETWORK.

#	Type	Kernel	Stride	Outputs	Activation
01	Conv.	3 × 3	1	16	ReLU
02			2		
03			1		
04					
05					
06				16	Tanh

TABLE II
HYPER-PARAMETERS OF THE CHROMA PREDICTION NETWORK.

#	Type	Kernel	Stride	Outputs	Activation
01	Conv.	3 × 3	1	128	ReLU
02			2		
03			1		
04			2		
05			1		
06			2		
07			1		
08			2		
09			1		
10					
11	DeConv.		1/2		
12	Conv.		1		
13	DeConv.		1/2		
14	Conv.		1		
15	DeConv.		1/2		
16	Conv.		1		
17	DeConv.		1/2		
18	Conv.		1	16	
19					
20				2	Tanh

In addition, this framework can also be applied to other YCbCr formats and even other color spaces. The hyper-parameters of network are supposed to be changed according to the associated YCbCr format and color space. For example, the stride of the second layer in luma down-sampling network should be set as 1 for YCbCr 4:4:4 format. As such, the luma down-sampling network is able to provide more luma information. For the case of YCbCr 4:2:2 format, the luma down-sampling can be only performed in the vertical or horizontal direction.

B. Adaptation to Variable Block Size in VVC

As mentioned in the Section I, the CTU size is 128 × 128 and the quad-tree plus binary- and ternary-tree partitions have been adopted in VVC. As a result, there are symmetric and asymmetric coding units with different sizes. How to design an efficient scheme to embrace the framework of chroma prediction for video coding becomes a challenging problem.

TABLE III
CANDIDATE SCHEME COMPARISONS.

Item	Proposed Scheme	1^{st} Alternative Scheme	2^{nd} Alternative Scheme
Model for Block Parition	128×128	128×128, 64×64, 32×32, 16×16, 8×8	One block partition, one model
Model Storage (about 9.17 MB/model)	1 CNNCP model	5 CNNCP models	25 CNNCP models
Operation at the Encoder Side	1 operation/CTU	341 operations/CTU	961 operations/CTU
Operation at the Decoder Side (Upper Bound)	1 operation/CTU	256 operations/CTU	256 operations/CTU

In this paper, N is fixed as 64 in CNNCP, which means that one CNNCP model is only applied for 128×128 block, such that the blocks smaller than 128×128 will copy the co-located prediction. The advantages of proposed scheme are that VVC is performed with the unit of CTU (128×128 block), such that the neighboring reconstructed CTUs can be easily collected. Moreover, the technique of dual tree provides the cross component information before chroma prediction in a CTU. Only one CNNCP model will be stored in the memory. For every CTU, the CNNCP is only performed once. The disadvantage is that if only a small block selects CNNCP in a CTU, the CNNCP will be performed with the unit of CTU at the decoder side.

In addition, there are two alternative schemes. For the first alternative scheme, five CNNCP models would be only applied for symmetric blocks, 128×128, 64×64, 32×32, 16×16 and 8×8 blocks, which means $N = 64, 32, 16, 8, 4$ in CNNCP, the blocks smaller than the block of 128×128, 64×64, 32×32, 16×16, or 8×8 will copy the co-located prediction from the associated block. Regarding the second alternative scheme, for every type of block partition, there is a CNNCP model, including the asymmetric partitions, 128×128, 128×64, 128×32, . . . , 16×8, 8×8, with 25 types of block partitions in total.

In the first alternative scheme, the spatial information cannot be easily collected for a block smaller than CTU because of the recursive partition manner. The final luma and chroma reconstructions will be determined after all blocks comparisons according to the RD cost. The number of the CNNCP models depends on the number of type of symmetric block partitions. Here, five CNNCP models are required for storage, i.e., 128×128, 64×64, 32×32, 16×16 and 8×8, which are designed for symmetric blocks. It is time consuming to train 5 CNN models. For every symmetric block size, the CNNCP will be performed to achieve the minimum cost at the encoder side. The number of CNNCP operation at the encoder side equals to the total number of block partitions, i.e., $1 + 4 + 16 + 64 + 256 = 341$. At the decoder side, the upper bound number of operation is that all these blocks are partitioned with size of 8×8, and they all select CNNCP, which can be calculated as follows, $(128 \div 8) \times (128 \div 8) = 256$. As same as the proposed scheme, for an asymmetric block, the pixel values will be directly copied from the associated symmetric case at the co-located location. For example, an asymmetric block with size of 16×8 will copy the pixel values from the associated 16×16 block, which is produced by CNNCP.

The difference between the first and second alternative schemes lies in that for every type of block partitions, there is a CNNCP model, not only for the symmetric blocks, but also for asymmetric blocks. As a result, more CNNCP models are required, and more operations will be conducted at the encoder side. The available information (spatial and cross component information) are symmetric, the changes should be made accordingly for the case of asymmetric blocks. As such the specific hyper-parameters for the asymmetric cases are required to be designed. In addition, the time cost of networks training is expensive.

The advantages and disadvantages of these three candidate schemes are compared, which are summarized in Table III. The proposed block scheme is finally used in this paper because of the benefits of simple implementation, less CNNCP models for storage, less time cost of network training, and less operations at the encoder and decoder sides. In the following experimental section, the parameter of N is fixed and set as 64 in the CNNCP.

C. Loss Function

There are two networks in CNNCP, i.e., down-sampling and chroma prediction networks. For the down-sampling network, the loss function can be represented by,

$$L_1 = ||\mathbf{F}_1(\mathbf{Y}) - \mathbf{Y}^*||^2, \tag{3}$$

where $\boldsymbol{Y}$ is the luma component with size of $4N \times 4N$, $\mathbf{F}_1(\cdot)$ is the down-sampling network, $\mathbf{Y}^*$ is the ground truth of luma component with size of $2N \times 2N$. For the chroma prediction network, the loss function is represented by

$$L_2 = \lambda||\mathbf{Cb'} - \mathbf{Cb}^*||^2 + (1 - \lambda)||\mathbf{Cr'} - \mathbf{Cr}^*||^2, \tag{4}$$

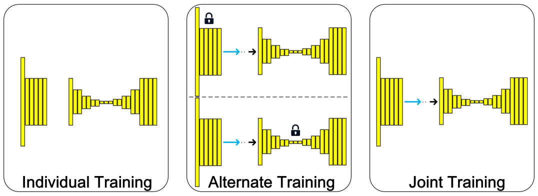
where $\lambda \in [0, 1]$ is a weight, $\mathbf{Cb'}$ and $\mathbf{Cr'}$ are cropped from the outputs of chroma prediction network with size of $N \times N$, $\mathbf{Cb}^*$ and $\mathbf{Cr}^*$ are the ground truth of two chroma components with size of $N \times N$.

$$\mathbf{Cb'}, \mathbf{Cr'} = \mathbf{F}_2(\mathbf{F}_1(\mathbf{Y}), \mathbf{D}, \mathbf{Cb}, \mathbf{Cr}), \tag{5}$$

where $\mathbf{F}_2(\cdot)$ is the chroma prediction network, $\mathbf{F}_1(\mathbf{Y})$ is the down-sampled luma component with size of $2N \times 2N$, $\mathbf{D}$ is the map of coding distortion level characterized by QP with size of $2N \times 2N$, $\mathbf{Cb}$ and $\mathbf{Cr}$ are the neighboring chroma information plus CCLM results as initialization with size of $2N \times 2N$.

D. Neural Network Training

The training dataset consists of 886 images from UCID database [44] and 900 images from DIV2K database [45].

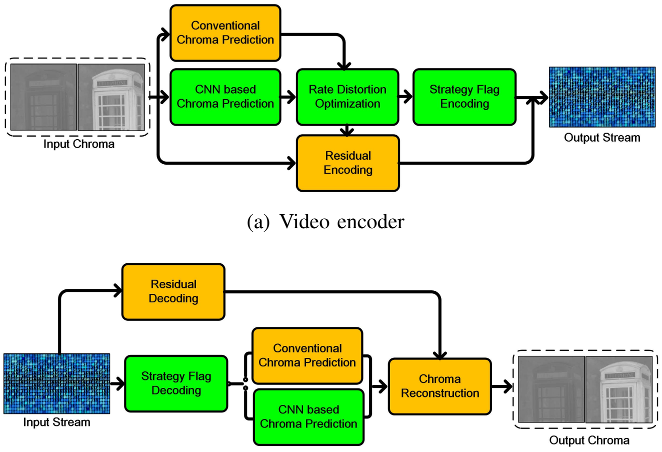


(a) Individual (b) Alternate (c) Joint (proposed)

Fig. 6. Three candidate training strategies.

The resolution of images from UCID database is 512×384. The resolution of images from DIV2K database is 2K (from 2040×648 to 2040×2040). They are encoded by the VVC Test Model (VTM) version 4.0 [46] with QPs {22, 27, 32, 37} under All Intra (AI) configuration. During the process of encoding, the training samples are collected, including the ground truth. The blocks located at the bottom-right of two chroma components are required to be predicted, which is replaced by the ground truth in the training pair. Due to the different resolutions of images in DIV2K, they are resized to 2048 × 1536 and packed as a sequence for coding. To collect more patches, the data augmentation is performed, i.e., the images are rotated in four directions. In UCID database, the number of patches can be calculated as follows, $886\times(512/128-1)\times(384/128-1)\times4\times4 = 85056$. In DIV2K database, the number of patches can be calculated as follows, $900 \times (2048/128 - 1) \times (1536/128 - 1) \times 4 \times 4 = 2376000$. Therefore, there are 2461056 training patches in total. 128000 of them will be used for validation.

Generally speaking, for these two networks, there are three candidate training strategies, i.e., individual, alternate, and joint, which are illustrated in Fig. 6. For individual training, the luma down-sampling network is firstly trained. After the luma down-sampling network is available, the chroma prediction network is trained. At last, they are combined together for the chroma prediction. This strategy makes the training relatively easy and can be conducted in parallel, but it may not achieve the best performance because the training of these two networks is individual, and global optimum is difficult to be achieved. For alternate training, the training of these two networks will be conducted one by one until both converge. It means that in a training epoch, down-sampling network is firstly trained, then the parameters of it will be fixed to generate down-sampled luma component for chroma prediction network training. In the next training epoch, this procedure will be re-called. The alternate training will finish when it reaches the defined number of epochs or converge. This training strategy may achieve the better performance, but its training will become time consuming because it is difficult to reach converge. It is worth mentioning that for individual and alternate training strategies, it is difficult to determine the ground truth of down-sampled luma. For joint training, it does not care the performance of luma down-sampling, which focuses on performance of chroma prediction. More accurate the chroma prediction, more coding gains can be achieved. During the training, the loss L_1 is ignored, the parameters of luma down-sampling network and chroma prediction are updated simultaneously. After the above discussion and analysis, the third training strategy is adopted in this paper due to its benefits of global optimum, easy converge and potential performance.



(a) Video encoder

(b) Video decoder

Fig. 7. Proposed method incorporated in the video codec for chroma coding.

In this paper, the Tensorflow package is utilized for networks training on NVIDIA GeForce GTX 1080 Ti GPU with AdamOptimizer. The batch size and the learning rate are set as 128 and 1×10^{-4}, respectively. During training, the value of λ in Eq. (4) is set as 0.5, as the Cb component is the same important as the Cr component.

E. Incorporation to the VVC Codec

In VVC, the technique of dual tree has been adopted, which means that intra luma and chroma components are separately encoded in a CTU. They can have different partition sizes. The luma component will be firstly processed and then the two chroma components. As such, the reconstructed luma component can be directly utilized as the cross component information. The proposed CNNCP can be feasibly incorporated into the video codec, as shown in Fig. 7.

At the encoder side, the conventional chroma prediction (including angular prediction, CCLM, MDLM) and CNNCP are both performed. Then the selection process is conducted based on RDO, such that the strategy with the minimum RD cost will be selected. One additional binary flag is adopted to indicate which strategy is selected, i.e., conventional or CNNCP. This flag is signalled in the bitstream. This process can be represented by

$$m^* = \arg\min_{m} \{D_m + \lambda_0(R_m + R_m^{'})\}, \tag{6}$$

where m is the chroma prediction strategy, i.e., conventional chroma prediction (including angular prediction, CCLM, MDLM) or CNNCP. D_m is the distortion, R_m is the coding bit of residue and other information, $R_m^{'}$ is the coding bit of the binary flag. λ_0 is the Lagrange Multiplier, which balances the coding distortion and bits. It is worth mentioning that the CNNCP is only performed at the unit of CTU. For the blocks smaller than CTU, the copying operation will be performed

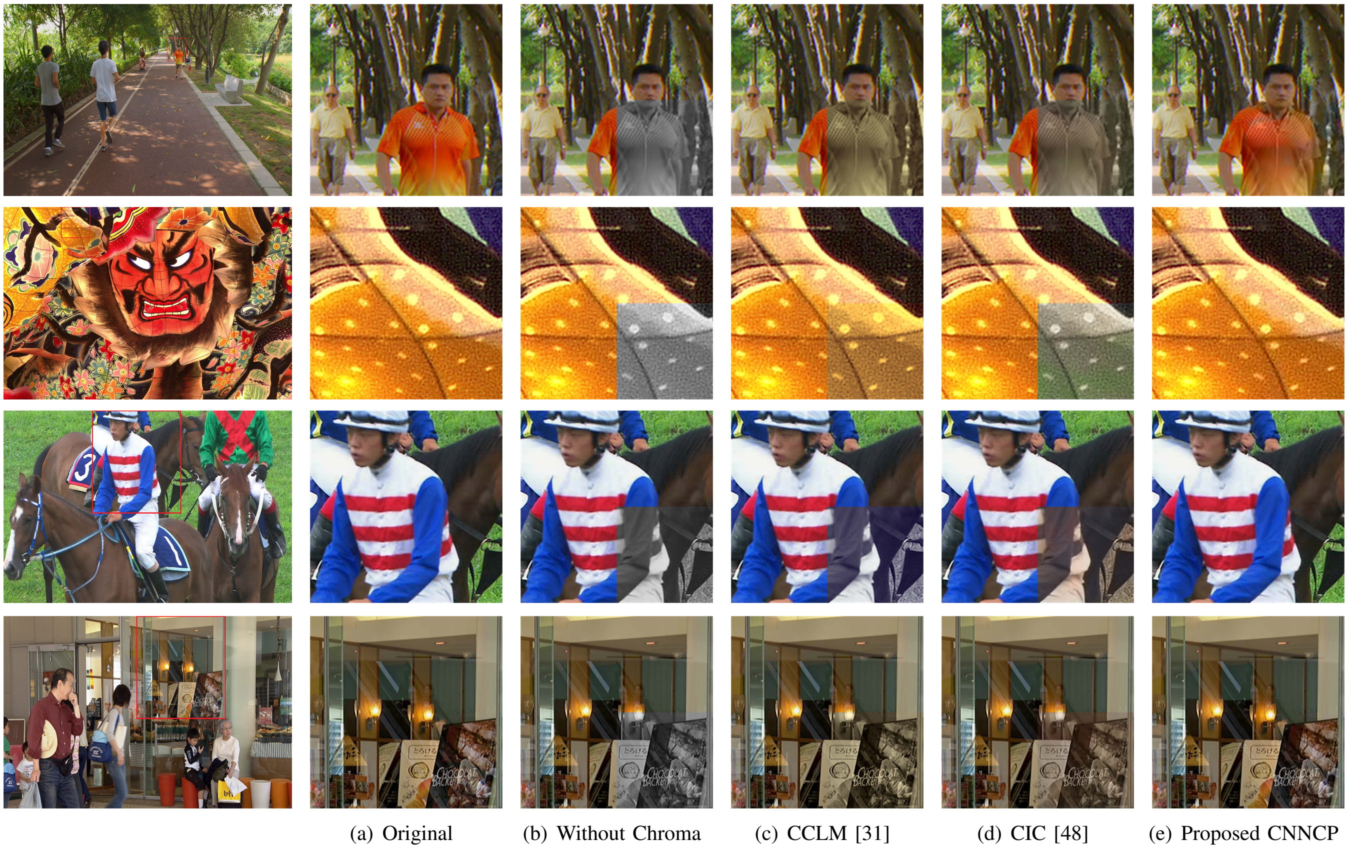


Fig. 8. Chroma prediction performance comparison with images from VVC sequences, and the blocks located at the bottom-right are predicted. From top to bottom, the sequence indicates ParkRunning3, NebutaFestival, RaceHorsesC, and BQMall, respectively. The PSNR values are illustrated in Table IV.

TABLE IV
CHROMA PREDICTION PERFORMANCE COMPARISON IN TERMS OF PSNR. [UNIT: DB]

Sequence	CCLM[31]		CIC[48]		CNNCP	
	Cb	Cr	Cb	Cr	Cb	Cr
ParkRunning3	16.97	14.21	18.33	17.38	22.97	23.27
NebutaFestival	20.73	20.74	16.77	16.35	37.22	39.75
RaceHorcesC	15.89	19.90	18.34	21.31	36.58	34.90
BQMall	25.35	26.00	27.50	29.89	36.57	36.81
AVERAGE	**19.74**	**20.21**	**20.24**	**21.23**	**33.33**	**33.68**

directly. The benefits have been discussed in detail in the last subsection. Because N equals to 64 in CNNCP, the spatial information fed to the network is absent for the blocks (128×128) located at the first row and column, such that the outputs of CNNCP are manually set as half of the largest pixel value, i.e., $\lfloor(2^k-1)/2+0.5\rfloor$, where k is the bit depth, and $\lfloor\rfloor$ is the floor operation.

At the decoder side, the flag will be decoded firstly. In a CTU, if one of the decoded flags is 1, CNNCP will be performed with the inputs of spatial and cross component information. In analogous to the encoder, the copying operation will be employed to the blocks smaller than CTU if the strategy of CNNCP has been used. For the above and left boundaries, if the CNNCP is selected, the blocks are filled with the value of $\lfloor(2^k-1)/2+0.5\rfloor$. More blocks select CNNCP, more decoding time may be consumed.

V. Experimental Results and Analyses

In this section, experiments are conducted on the platform of VTM 4.0 [46], in which the proposed CNNCP has been implemented in both the video encoder and decoder. The workstation equipped with the Intel Core i7-6950X CPU, 64GB memory, Windows 10 Enterprise 64-bit operating system, is used in our experiments. The GPU is only activated for networks training, while the CPU is used for encoding and decoding. The original VTM 4.0 is utilized as the anchor for RD performance comparison. The RD performance is measured by Bjøntegaard Delta Bit Rate (BD-BR) [47], and the negative value implies the RD performance improvement and vice versa. We focus on high resolution of videos that the sequences of Class D are not used.

A. Chroma Prediction Performance Comparison

Firstly, the results of chroma prediction are compared with two approaches, i.e., CCLM, and Colorful Image Colorization (CIC) [48]. CCLM is a simple linear model, which has been adopted in VVC. CIC is a deep learning based scheme, where the CNN has been used to generate the colorful information from the gray information. The values of PSNR are also calculated with respect to the original in case of Cb and Cr components. Twenty images are randomly selected from Imagenet Large Scale Visual Recognition Challenge 2012 (ILSVRC2012) database [49]. To adapt to the scenario of block based coding, blocks with size of 256×256 are collected

TABLE V
PERFORMANCE EVALUATION IN TERMS OF BD-BR WITH QPS {22, 27, 32, 37}. [UNIT: %]

Class	Sequence	CNNCP + CCLM vs. CCLM[31]				CNNCP + MMLM + MFLM + CCLM vs. MMLM[32] + MFLM[32] + CCLM				CNNCP + MDLM + CCLM vs. MDLM[33] + CCLM			
		Y	U	V	YUV	Y	U	V	YUV	Y	U	V	YUV
A	Tango2	-4.397	-1.562	-1.857	-3.966	-4.817	-3.571	1.914	-4.418	-4.155	0.436	1.323	-3.731
	FoodMarket4	-9.522	-4.086	-2.799	-8.304	-8.908	-2.457	-4.273	-7.975	-9.309	-3.378	-2.178	-7.794
	Campfire	-0.052	0.185	-0.107	-0.025	0.089	-0.227	0.064	0.042	0.029	0.222	0.242	0.077
	CatRobot1	-9.006	-11.32	-8.023	-9.441	-8.180	-5.779	-6.208	-7.905	-8.724	-9.846	-6.638	-8.999
	DaylightRoad2	-2.314	3.853	-2.109	-2.165	-4.438	6.150	-2.504	-3.956	-2.184	4.564	-2.338	-2.063
	ParkRunning3	-15.16	-12.88	-13.61	-14.35	-20.10	-17.23	-17.95	-18.98	-15.03	-12.44	-12.96	-14.03
B	MarketPlace	-3.249	0.000	5.410	-2.716	-4.156	-2.836	2.426	-3.833	-3.170	1.182	6.637	-2.538
	RitualDance	0.041	-0.412	0.311	0.005	-0.061	-2.150	-0.223	-0.218	-0.048	0.590	0.494	0.058
	BasketballDrive	-5.107	-2.904	-1.173	-4.769	-6.373	-5.992	-2.200	-6.153	-5.081	-1.532	0.988	-4.520
	BQTerrace	-1.522	-1.256	-2.317	-1.574	-2.629	-5.096	-6.116	-2.733	-1.659	-0.243	-1.432	-1.625
	Cactus	-6.069	-3.644	-5.658	-6.184	-5.542	-1.212	1.075	-5.009	-6.072	-2.666	-5.149	-6.028
	Kimono	-3.784	-0.002	-2.982	-3.202	-4.998	-2.358	-8.161	-5.064	-3.822	1.229	-3.041	-3.065
	ParkScene	-1.550	3.125	-2.880	-1.340	-2.609	5.261	-5.825	-2.314	-1.460	2.974	-1.645	-1.210
C	BQMall	-2.652	-3.778	-4.523	-2.832	-4.166	3.826	4.542	-3.429	-2.783	0.570	-2.944	-2.633
	PartyScene	-2.467	-2.415	-2.626	-2.441	-2.960	3.411	4.902	-2.258	-2.495	-1.410	-1.160	-2.317
	BasketballDrill	-1.775	-2.202	-3.406	-1.901	-3.661	-3.218	-0.918	-3.488	-2.191	-3.433	-0.524	-2.151
	RaceHorsesC	-3.122	-2.373	-4.443	-3.108	-6.044	-2.498	-2.068	-5.589	-3.145	-1.947	-4.135	-3.083
E	FourPeople	-4.376	-9.087	-14.66	-5.266	-4.365	-1.739	-3.907	-4.191	-4.583	-7.674	-12.83	-5.282
	Johnny	-5.287	-18.40	-19.84	-7.163	-5.276	-7.750	-4.454	-5.335	-5.130	-18.54	-20.66	-7.000
	KristenAndSara	-5.327	-14.25	-18.47	-6.833	-5.443	-7.753	-7.570	-5.620	-5.297	-13.92	-17.18	-6.676
	Vidyo1	-4.095	-8.241	-13.57	-4.970	-3.794	-7.234	-7.215	-4.045	-3.634	-4.933	-12.18	-4.317
AVERAGE		**-4.323**	**-4.364**	**-5.682**	**-4.407**	**-5.163**	**-2.878**	**-3.079**	**-4.879**	**-4.283**	**-3.343**	**-4.634**	**-4.235**

from these images for chroma prediction comparison. The results reveal that the Cb and Cr PSNR values of CCLM, CIC and proposed CNNCP can reach (21.97dB, 20.87dB), (21.94dB, 21.39dB), and (24.45dB, 25.75dB) on average, respectively. The performance of proposed CNNCP is better than the other two schemes.

In addition, the VVC test sequences are utilized as well. The visual results are vividly shown in Fig. 8. These blocks are collected from *ParkRunning3*, *NebutaFestival*, *RaceHorcesC*, and *BQMall*. The resolution of all these sub-blocks presented is 256×256. The chroma component of sub-block (128×128) located at the bottom-right are predicted by the above mentioned schemes, and the other sub-blocks located at the above-left, above, and left are as same as the original. Here, the prediction is performed in case of YCbCr 4:2:0 format, and it is re-scaled to the same resolution as luma component for visualization.

From Fig. 8, it can be found that the results of the proposed CNNCP are more consist with the neighboring pixels, and they are more close to the original in terms of visual quality. Additionally, as shown in Table IV, from the perspective of objective evaluation, the PSNR values of the proposed method are higher than those of other two schemes, which indicates that the proposed method achieves the best performance. The reasons are that the spatial information and cross component information are both considered in the proposed CNNCP. The spatial information may provide the clues for chroma prediction. However, it is absent in CIC. Although the spatial information is used in CCLM, the number of neighboring reference pixels is limited, and the simple linear model cannot handle diverse contents.

B. Coding Performance Evaluation

In addition, the proposed method is evaluated with the state-of-the-art chroma prediction methods, i.e., CCLM, MMLM, MFLM, and MDLM. It is worth mentioning that CCLM and MDLM have been incorporated in VTM 4.0, while CCLM, MFLM and MDLM have been incorporated in BenchMark Set (BMS) 1.0. Therefore, we implement the proposed CNNCP into VTM 4.0 and BMS 1.0 for the comparison. Twenty one test sequences, which are different from the training data, are encoded under four QPs, including {22, 27, 32, 37}. The experimental results are shown in Table V. The values of BD-BR are calculated individually for Y, U, V, and YUV components. The PSNR value of YUV component is the weighted value of Y, U, and V components.

For the first case, CNNCP + CCLM vs. CCLM, it reduces 4.323%, 4.364%, 5.682%, and 4.407% bit rate on average for Y, U, V, and YUV components, respectively. For the case of CNNCP + MMLM + MFLM + CCLM vs. MMLM + MFLM + CCLM, it can achieve 5.163%, 2.878%, 3.079%, and 4.879% bit rate reductions for Y, U, V, and YUV components, respectively. For the last case, CNNCP + MDLM + CCLM vs. MDLM + CCLM, 4.283%, 3.343%, 4.634%, and 4.235% bit rate reductions for Y, U, V, and YUV components can be achieved, respectively. Obviously, with the incorporated CNNCP, it can further improve the coding performance. The reasons are that CNNCP considers the spatial information besides cross component information. Also, the influence of luma down-sampling is taken into account with a neural network. In addition, the outputs of CCLM are utilized as the initialization, which may provide clues for chroma prediction. In particular, the sequence of *ParkRunning3* achieves the best coding gain. The reason is that there are many trees shown in the sequence of *ParkRunning3*, and the color of tree can

TABLE VI
CODING PERFORMANCE IN TERMS OF BD-BR WITH SSIM METRIC. [UNIT: %]

Class	Sequence	Y	U	V
B	BQTerrace	-1.15	2.09	0.97
	Cactus	-6.54	-2.60	-1.73
	BasketballDrive	-4.78	5.48	3.20
	Kimono	-3.94	-1.29	4.74
C	PartyScene	-2.78	0.76	-0.33
	RaceHorsesC	-3.69	-2.40	-2.59
	BQMall	-3.39	2.07	2.48
	BasketballDrill	-2.33	-2.59	1.04
E	FourPeople	-4.88	-8.80	-2.62
	KristenAndSara	-5.50	-4.00	-2.63
	Johnny	-6.00	-7.64	-6.99
	Vidyo1	-3.96	-7.52	-0.64
AVERAGE		**-4.08**	**-2.20**	**-0.43**

TABLE VII
CODING PERFORMANCE UNDER PLATFORM OF VTM VERSION 9.3 IN TERMS OF BD-BR. [UNIT: %]

Class	Sequence	Y	U	V	YUV
A	Tango2	-2.893	-3.589	-0.907	-2.792
	FoodMarket4	-8.275	-1.915	-2.783	-6.971
	Campfire	-0.045	-0.095	0.255	-0.026
	CatRobot1	-5.817	-7.049	-4.682	-5.922
	DaylightRoad2	-1.503	1.138	-1.362	-1.435
	ParkRunning3	-13.81	-12.38	-14.02	-13.59
B	MarketPlace	-2.660	0.822	0.478	-2.277
	RitualDance	0.105	0.203	0.098	0.116
	BasketballDrive	-4.272	-1.029	1.715	-3.706
	BQTerrace	-1.088	-0.745	-2.334	-1.093
	Cactus	-5.335	-6.853	-10.76	-6.051
	Kimono	-3.441	-0.938	-2.393	-2.960
	ParkScene	-1.105	0.843	-1.556	-0.961
C	BQMall	-2.116	-1.740	-2.682	-2.167
	PartyScene	-1.815	-2.333	-1.723	-1.808
	RaceHorsesC	-2.171	0.373	0.477	-1.736
	BasketballDrill	-1.291	-3.487	-0.862	-1.416
E	FourPeople	-3.389	-9.996	-12.42	-4.478
	Johnny	-4.161	-20.64	-17.19	-6.283
	KristenAndSara	-4.297	-9.868	-10.41	-5.289
	Vidyo1	-3.130	-7.869	-13.42	-4.050
AVERAGE		**-3.453**	**-4.149**	**-4.594**	**-3.566**

be inferred according to the natural scene statistics. Thus, the learned network can reduce the redundancy from the perspective of prior statistics to some extent. In this work, the luma gain and the chroma gain are similar. The reason is that the linear models, including CCLM and MDLM, are enabled in the designed video codec. The proposed scheme is required to compete with them in terms of RD cost and the performance is finally calculated with respect to the anchor codec equipped with linear models (including CCLM and MDLM).

Besides the PSNR metric, the Structural Similarity Index (SSIM) [50] is also adopted to evaluate the quality of reconstructed sequences. From Table VI, it can be found that the bit rate reduction can reach 4.08%, 2.20%, and 0.64% on average for Y, U, and V components. The performance of luma component is similar to that of PSNR metric, while there is a gap for the chroma component. The reason is that in the video codec the distortion is still represented by Mean Squared Error (MSE) or Sum of Absolute Difference (SAD), and the loss function of neural network is related to them.

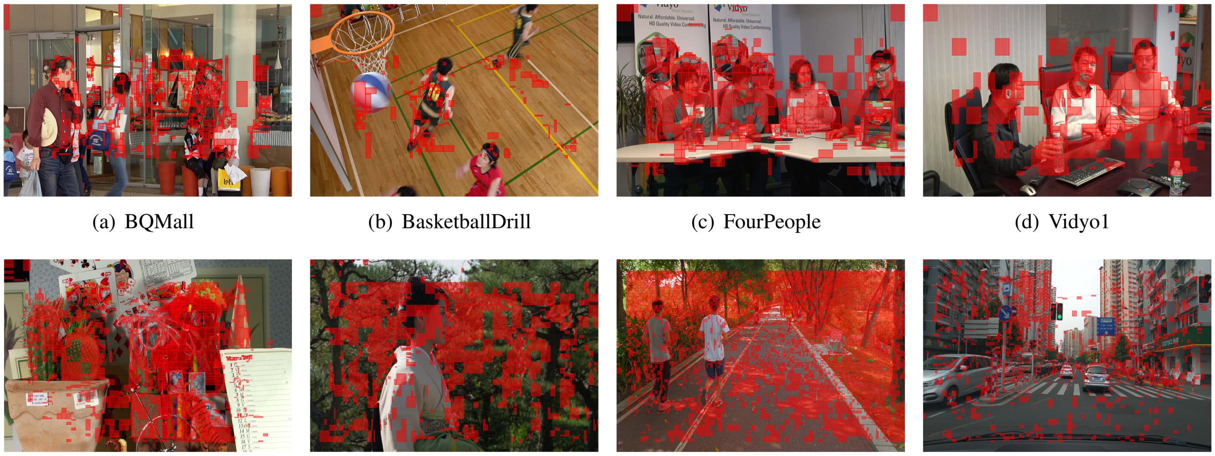
In addition, the proposed CNNCP has been implemented on the VTM 9.3 for performance evaluation, and the results are illustrated in Table VII. It can be observed that the bit rate savings on average can reach 3.453%, 4.149%, 4.594%, and 3.566% for Y, U, V and YUV components, which are a little worse than those on the VTM 4.0. The reason is that more advanced coding tools have been adopted on the VTM 9.3, and there is overlap between them and the proposed CNNCP.

The distributions of CNNCP selected in the chroma prediction are illustrated in Fig. 9 and Table VIII. Sequences of *BQMall* (832×480), *BasketballDrill* (832×480), *FourPeople* (1280×720), *Vidyo1* (1280×720), *Cactus* (1920×1080), *Kimono* (1920×1080), *ParkRunning3* (3840×2160), and *DaylightRoad2* (3840×2160) are encoded with QP 22 by the proposed method. They are the first frames of each sequence and all re-scaled to the same resolution for visualization. The blocks with red color use CNNCP. It can be found that abundant areas select CNNCP, which further provides evidences regarding the advantage of CNNCP. Additionally, it can be observed that most of blocks with CNNCP are small blocks, including symmetric and asymmetric ones. The blocks with CNNCP mainly locate at the texture area. From Table VIII, it can be found that the percentage that selects CNNCP can reach 12.4%, 19.9%, 31.7%, and 41.3% on average for four different QP settings. The percentage increases as the QP increases, which is related to the coding gains shown in Table V. The larger percentage, the more coding gain can be achieved.

C. Ablation Study

In the proposed architecture of CNNCP, the modules include luma down-sampling network, CCLM initialization and chroma prediction network. To evaluate the effectiveness of the architecture, the ablation study is conducted, including (1) CNNCP with conventional down-sampling, (2) luma down-sampling network generating 1 output and (3) CNNCP without CCLM initialization. The first case aims to show whether the luma down-sampling network is necessary; the second case aims to show whether more luma down-sampling versions are able to improve the performance; the third case aims to show whether the CCLM initialization is able to enhance the chroma prediction. In case of the luma down-sampling network being absent, the conventional method is adopted instead, where the down-sampled luma pixel value is calculated with four neighboring luma pixels and the associated weights are identical. In case of CNNCP without CCLM initialization, the to-be-predicted chroma pixel values are set as half of the largest pixel value, i.e., $\lfloor (2^k - 1)/2 + 0.5 \rfloor$, k is the bit depth. Two sequences of every class are tested, and they are encoded with four QPs, including $\{22, 27, 32, 37\}$. The original VTM 4.0 is utilized as the anchor. The results are shown in Table IX in terms of BD-BR. The values of BD-BR are calculated individually for Y, U, and V components.

With conventional luma down-sampling method, it can achieve 1.733%, 4.302%, and 5.265% bit rate saving on average for Y, U, and V components, respectively. For the case



(a) BQMall (b) BasketballDrill (c) FourPeople (d) Vidyo1

(e) Cactus (f) Kimono (g) ParkRunning3 (h) DaylightRoad2

Fig. 9. CNNCP selected in the chroma prediction.

TABLE VIII
PERCENTAGE OF THE PROPOSED METHOD SELECTED. [UNIT: %]

Class	Sequence	QP			
		22	27	32	37
A	Tango2	4.48	8.92	19.6	33.6
	FoodMarket4	3.79	10.4	20.1	31.5
	Campfire	0.08	0.02	0.02	0.78
	CatRobot1	16.2	24.7	45.3	57.7
	DaylightRoad2	10.5	18.6	45.1	54.8
	ParkRunning3	32.7	36.8	46.8	58.1
B	MarketPlace	5.20	8.94	20.1	32.6
	RitualDance	0.65	1.19	2.12	2.59
	BasketballDrive	2.73	6.64	20.9	28.2
	BQTerrace	3.54	4.31	13.8	26.1
	Cactus	28.8	38.0	46.8	56.9
	Kimono	19.7	20.7	30.2	48.3
	ParkScene	4.20	3.74	8.99	12.6
C	BQMall	5.05	12.6	33.2	44.4
	PartyScene	4.23	9.12	16.5	21.4
	RaceHorsesC	1.19	1.27	3.82	5.86
	BasketballDrill	2.29	5.49	20.7	36.2
E	FourPeople	31.8	61.2	79.9	90.5
	Johnny	20.9	42.6	59.5	68.1
	KristenAndSara	30.0	47.7	64.0	78.5
	Vidyo1	31.9	54.4	69.1	78.9
AVERAGE		**12.4**	**19.9**	**31.7**	**41.3**

of CNNCP with down-sampling network generating 1 output, it reduces 4.090%, 4.358%, and 5.039% bit rate on average for Y, U, and V components. If the CCLM initialization is removed, it achieves 4.157%, 3.872%, and 4.408% bit rate reduction on average for Y, U, and V components. For the proposed CNNCP, it is able to achieve 5.876%, 6.268%, and 7.689% bit rate saving on average for Y, U, and V components.

From the results, it can be found that more coding gains are achieved with the proposed CNNCP, as the luma down-sampling network provides more down-sampled versions when compared to the conventional down-sampling method, and CCLM initialization can provide the clues for chroma prediction. Therefore, the luma down-sampling network, more down-sampled luma versions and CCLM initialization are effective and can further improve the performance of chroma prediction.

D. Cross-Validation of CNNCP Under Different QP Settings

Additionally, we conduct experiments to validate the coding performance for other QP settings. Here, two QP settings are tested, including low QP setting $\{11, 16, 21, 26\}$ and high QP setting $\{33, 38, 43, 48\}$. It is conducted on the platform of VTM 4.0, i.e., CNNCP + MDLM + CCLM vs. MDLM + CCLM. However, it should be noted that the CNNCP model is not changed, which is trained with QPs $\{22, 27, 32, 37\}$. The results are shown in Table X.

For the low QP setting, the proposed method achieves 0.283%, 0.325% and 0.315% on average bit rate reductions for luma and two chroma components. For the high QP setting, the proposed method can reduce 2.539%, 9.105% and 8.429% bit rate on average for luma and two chroma components. It can be observed that more coding gains are achieved as the QP increases. The reasons are that as the QP increases, the reference pixels would be severely degraded, and the number of them is limited for the conventional chroma prediction, which limits the coding performance improvement. By contrast, for the proposed method, more reference pixels are adopted which can provide more useful information. In addition, the coding distortion level (QP value) is adopted as an input fed into the network, which may make the prediction results more accurate.

E. Computational Complexity Analyses

Although the neural network incorporated into video codec can achieve promising coding gains, it significantly increases the computational complexity of encoding and decoding due to the convolutional operation. The computational complexity is calculated by,

$$\Delta T = \frac{1}{4}\sum_{i=1}^{4}\frac{T_{\Psi}(QP_i)}{T_c(QP_i)}, \tag{7}$$

where $T_c(QP_i)$ is the coding/decoding time of the anchor video codec under QP_i, and $T_{\Psi}(QP_i)$ is the coding/decoding time of the video codec equipped with proposed method under QP_i. The results of computational complexity are illustrated

TABLE IX
CODING PERFORMANCE OF ABLATION STUDY IN TERMS OF BD-BR. [UNIT: %]

Class	Sequence	CNNCP with conventional luma down-sampling			Luma down-sampling network generating 1 output			CNNCP without CCLM initialization			Proposed CNNCP		
		Y	U	V	Y	U	V	Y	U	V	Y	U	V
A	CatRobot1	-2.738	-3.737	-3.517	-6.433	-6.824	-5.346	-6.110	-5.791	-5.129	-8.724	-9.846	-6.638
	ParkRunning3	-1.406	-1.536	-1.495	-7.855	-6.790	-6.584	-9.862	-8.060	-8.001	-15.03	-12.44	-12.96
B	BQTerrace	-0.024	0.123	-1.338	-0.768	1.395	0.141	-0.168	1.057	0.042	-1.659	-0.243	-1.432
	Cactus	-3.698	-4.040	-8.314	-4.670	-1.183	-1.408	-4.327	-1.122	-0.902	-6.072	-2.666	-5.149
C	PartyScene	-0.189	-0.062	0.015	-1.403	2.137	1.432	-1.613	2.209	1.579	-2.495	-1.410	-1.160
	RaceHorsesC	-0.091	0.031	1.631	-2.311	0.402	0.556	-2.546	0.062	-1.432	-3.145	-1.947	-4.135
E	FourPeople	-3.481	-7.701	-9.571	-4.230	-8.874	-11.15	-3.830	-6.741	-9.730	-4.583	-7.674	-12.83
	KristenAndSara	-4.234	-17.49	-19.53	-5.052	-15.13	-17.95	-4.799	-12.59	-14.09	-5.297	-13.92	-17.18
AVERAGE		**-1.733**	**-4.302**	**-5.265**	**-4.090**	**-4.358**	**-5.039**	**-4.157**	**-3.872**	**-4.408**	**-5.876**	**-6.268**	**-7.689**

in Table XI, where the coding experiments are all conducted on the platform of CPU, and the original VTM is utilized as the anchor. It should be noted that the multi-thread speedup is not used in the coding experiment. For the encoding, the computational complexity only increases 16% on average. However, the computational complexity of decoding increases 834% on average. The more blocks select the proposed method of CNNCP, the more decoding time will be consumed. From Tables V and XI, it can be found that the sequence of *ParkRunning3* achieves the best coding gain, i.e., 14.03% bit rate saving in case of YUV component, but the running time of its decoding is 31.5 times compared to the original VTM. In addition, it can be observed that limited runtime is spent on the module of model inference.

At present, the computational complexity is still a problem for the deep learning based video coding. However, the development of the advanced coding tools cannot be impeded by the complexity issues. We sincerely believe that the fast algorithms and advanced hardware developed in future could make the deep learning based video coding applied in real scenarios.

VI. CONCLUSIONS

In this paper, a deep learning based chroma prediction method for intra coding has been proposed. Different from the conventional angular and linear models, the chroma prediction is performed with deep neural networks. The sophisticated neural networks make it possible to transfer from the given gray version to the colorful version in a data-driven manner, where the spatial information and cross component information are both fully considered. To further improve the performance, the coding distortion level is also fed to the neural network, and the results of CCLM are adopted for the chroma initialization. In addition, the RDO is performed to select the better prediction strategy from the original method and CNNCP with an additional binary flag. Extensive experimental results demonstrate the superior performance of the proposed scheme compared to the state-of-the-art chroma prediction methods.

REFERENCES

[1] K. Müller, H. Schwarz, D. Marpe, C. Bartnik, S. Bosse, H. Brust, T. Hinz, H. Lakshman, P. Merkle, F. H. Rhee, G. Tech, M. Winken, and T. Wiegand, "3D high-efficiency video coding for multi-view video and depth data", *IEEE Trans. Image Process.*, vol. 22, no. 9, pp. 3366-3378, Sept. 2013.

[2] E. Francois, C. Fogg, Y. He, X. Li, A. Luthra, and A. Segall, "High dynamic range and wide color gamut video coding in HEVC: status and potential future enhancements", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 63-75, Jan. 2016.

[3] T. Wiegand, G. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560-576, Jul. 2003.

[4] G. Sullivan, J. Ohm, W. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649-1668, Dec. 2012.

[5] B. Bross, "Versatile video coding (Draft 2)", Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC/ 29/WG 11, Tech. Rep. Doc. JVET-K1001-v7, Ljubljana, SI, Jul. 10-18, 2018.

[6] J. Li, B. Li, J. Xu, and R. Xiong, "Efficient multiple-line-based intra prediction for HEVC", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 4, pp. 947-957, Apr. 2018.

[7] X. Xu, S. Liu, T. Chuang, Y. Huang, S. Lei, K. Rapaka, C. Pang, V. Seregin, Y. Wang, and M. Karczewicz, "Intra block copy in HEVC screen content coding extensions", *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, vol. 6, no. 4, pp. 409-419, Dec. 2016.

[8] K. Zhang , Y. Chen, L. Zhang, W. Chien, and M. Karczewicz, "An improved framework of affine motion compensation in video coding", *IEEE Trans. Image Process.*, vol. 28, no. 3, pp. 1456-1469, Mar. 2019.

[9] B. Bross, J. Chen, and S. Liu, "Versatile video coding (Draft 4)", Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Tech. Rep. Doc. JVET-M1001-v7, Marrakech, MA, Jan.9-18, 2019.

[10] K. Naser, T. Poirier, and F. L. Leannec, "Non-CE6: shape adaptive transform selection for ISP, SBT and MTS", Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Tech. Rep. Doc. JVET-N0388-v5, Geneva, CH, Mar.19-27, 2019.

[11] S. Ma, X. Zhang, C. Jia, Z. Zhao, S. Wang, and S. Wang, "Image and video compression with neural networks: a review", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 6, pp. 1683-1698, Jun. 2020.

[12] Y. Zhang, S. Kwong, and S. Wang, "Machine learning based video coding optimizations: a survey", *Inf. Sci.*, vol. 506, pp. 395-423, Jan. 2020.

[13] Z. Zhao, S. Wang, S. Wang, X. Zhang, S. Ma, and J. Yang, "Enhanced bi-prediction with convolutional neural network for high efficiency video coding", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 11, pp. 3291-3301, Nov. 2019.

[14] L. Zhao, S. Wang, X. Zhang, S. Wang, S. Ma, and W. Gao, "Enhanced motion-compensated video coding with deep virtual reference frame generation", *IEEE Trans. Image Process.*, vol. 28, no. 10, pp. 4832-4844, Oct. 2019.

[15] J. Li, B. Li, J. Xu, R. Xiong, and W. Gao, "Fully connected network-based intra prediction for image coding", *IEEE Trans. Image Process.*, vol. 27, no. 7, pp. 3236-3247, Jul. 2018.

[16] L. Zhu, S. Kwong, Y. Zhang, S. Wang and X. Wang, "Generative adversarial network based intra prediction for video coding", *IEEE Trans. Multimedia*, vol. 22, no. 1, pp. 45-58, Jan. 2020.

[17] Y. Li, L. Li, Z. Li, J. Yang, N. Xu, D. Liu, and H. Li, "A hybrid neural network for chroma intra prediction", IEEE International Conference on Image Processing, Athens, Greece, pp. 1797-1801, Oct. 7-10, 2018.

TABLE X
PERFORMANCE COMPARISON IN TERMS OF BD-BR WITH DIFFERENT QP SETTINGS UNDER PLATFORM OF VTM VERSION 4.0 (CNNCP + MDLM[33] + CCLM[31] VS. MDLM + CCLM). [UNIT: %]

Class	Sequence	Low QP (11,16,21,26)				High QP (33,38,43,48)			
		Y	U	V	YUV	Y	U	V	YUV
A	Tango2	-0.036	0.198	-0.315	-0.046	-3.018	-4.106	-0.441	-2.918
	FoodMarket4	-0.030	0.151	0.759	0.023	-2.628	-2.513	-3.145	-2.661
	Campfire	0.009	-0.032	-0.111	-0.016	-0.034	-0.012	-0.337	-0.056
	CatRobot1	-0.349	-0.488	-0.221	-0.374	-6.749	-14.42	-13.20	-7.381
	DaylightRoad2	-0.021	-0.104	-0.358	-0.051	-1.748	-8.797	-6.809	-2.045
	ParkRunning3	-0.434	-0.506	-0.523	-0.490	-3.296	-4.487	-4.672	-3.708
B	MarketPlace	0.003	-0.139	-0.243	-0.035	-1.171	0.670	0.338	-1.080
	RitualDance	-0.104	0.501	-0.568	-0.068	-0.092	1.592	0.611	-0.009
	BasketballDrive	0.054	-0.553	-0.653	-0.026	-2.694	-6.06	-2.546	-2.791
	BQTerrace	0.022	-0.135	-0.058	-0.006	-0.244	-2.082	-2.149	-0.283
	Cactus	-0.917	-1.020	-0.244	-0.902	-6.269	-23.99	-23.20	-7.332
	Kimono	-0.031	-0.449	-0.160	-0.106	-1.095	1.031	-4.290	-1.086
	ParkScene	-0.015	-0.292	0.169	-0.033	-0.333	1.129	0.796	-0.266
C	BQMall	0.073	-0.232	-0.279	0.000	-1.219	0.875	0.003	-1.126
	PartyScene	0.018	-0.081	-0.172	-0.031	-0.380	-0.512	-0.870	-0.379
	RaceHorsesC	-0.008	0.262	-0.052	0.033	-0.033	-0.876	1.253	-0.036
	BasketballDrill	0.010	0.046	-0.212	-0.011	-1.158	-4.605	-3.307	-1.412
E	FourPeople	-0.912	-1.014	-0.627	-0.924	-5.190	-25.62	-29.54	-6.064
	Johnny	-1.080	-1.339	-0.698	-1.125	-6.439	-36.06	-27.42	-7.985
	KristenAndSara	-1.670	-1.813	-2.172	-1.759	-5.972	-36.32	-38.61	-7.610
	Vidyo1	-0.518	0.224	0.119	-0.448	-3.555	-26.04	-19.46	-4.332
AVERAGE		**-0.283**	**-0.325**	**-0.315**	**-0.305**	**-2.539**	**-9.105**	**-8.429**	**-2.884**

TABLE XI
COMPUTATIONAL COMPLEXITY OF THE PROPOSED METHOD UNDER PLATFORM OF CPU.

Class	Sequence	Encode ΔT	Decode ΔT	Ratio
A	Tango2	1.16	5.84	2.33%
	FoodMarket4	1.29	3.15	5.56%
	Campfire	1.17	1.67	2.44%
	CatRobot1	1.17	9.45	2.23%
	DaylightRoad2	1.14	11.8	1.57%
	ParkingRunning3	1.16	31.5	2.14%
B	MarketPlace	1.17	4.88	1.83%
	RitualDance	1.23	4.72	4.90%
	BasketballDrive	1.13	10.5	1.72%
	BQTerrace	1.17	10.1	2.08%
	Cactus	1.12	8.91	1.33%
	Kimono	1.16	18.7	2.32%
	ParkScene	1.13	9.27	1.60%
C	BQMall	1.14	8.82	0.90%
	PartyScene	1.23	7.37	0.71%
	RaceHorsesC	1.14	7.16	0.81%
	BasketballDrill	1.12	7.94	0.95%
E	FourPeople	1.15	10.1	1.83%
	Johnny	1.18	5.62	2.43%
	KristenAndSara	1.15	9.85	2.54%
	Vidyo1	1.13	8.89	1.93%
AVERAGE		**1.16**	**9.34**	**2.10%**

Note: Ratio indicates the percentage of runtime of model inference module in the video coding.

[18] N. Yan, D. Liu, H. Li, B. Li, L. Li, and F. Wu, "Convolutional neural network-based fractional-pixel motion compensation", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 3, pp. 840-853, Mar. 2019.

[19] J. Liu, S. Xia, W. Yang, M. Li, and D. Liu, "One-for-all: grouped variation network-based fractional interpolation in video coding", *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2140-2151, May 2019.

[20] L. Yu, L. Shen, H. Yang, L. Wang, and P. An, "Quality enhancement network via multi-reconstruction recursive residual learning for video coding", *IEEE Signal Process. Lett.*, vol. 26, no. 4, pp. 557-561, Apr. 2019.

[21] C. Jia, S. Wang, X. Zhang, S. Wang, J. Liu, S. Pu, and S. Ma, "Content-aware convolutional neural network for in-loop filtering in high efficiency video coding", *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3343-3356, Jul. 2019.

[22] S. Lee, S.-W. Park, P. Oh, and M. Kang, "Colorization-based compression using optimization", *IEEE Trans. Image Process.*, vol. 22, no. 7, pp. 2627-2636, Jul. 2013.

[23] A. Bugeau, V.-T. Ta, and N. Papadakis, "Variational exemplar-based image colorization", *IEEE Trans. Image Process.*, vol. 23, no. 1, pp. 298-307, Jan. 2014.

[24] B. Li, F. Zhao, Z. Su, X. Liang, Y.-K. Lai, and P. L. Rosin, "Example-based image colorization using locality consistent sparse representation", *IEEE Trans. Image Process.*, vol. 26, no. 11, pp. 5188-5202, Nov. 2017.

[25] P. Peter, L. Kaufhold, and J. Weickert, "Turning diffusion-based image colorization into efficient color compression", *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 860-869, Feb. 2017.

[26] B. Li, Y.-K. Lai, M. John, and P. L. Rosin, "Automatic example-based image colourisation using location-aware cross-scale matching", *IEEE Trans. Image Process.*, vol. 28, no. 9, pp. 4606-4619, Sept. 2019.

[27] Z. Cheng, Q. Yang, and B. Sheng, "Colorization using neural network ensemble", *IEEE Trans. Image Process.*, vol. 26, no. 11, pp. 5491-5505, Nov. 2017.

[28] Z. Cheng, Q. Yang, and B. Sheng, "Deep colorization", International Conference on Computer Vision (ICCV), Santiago, Chile, pp. 415-423, Dec. 11-18, 2015.

[29] S. Iizuka, E. S.-Serra, and H. Ishikawa, "Let there be color!: joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification", *ACM Trans. Graph*, vol. 35, no. 4, Artical 110, pp. 1-11, Jul. 2016.

[30] B. Zhang, M. He, J. Liao, P. V. Sander, L. Yuan, A. Bermak, and D. Chen, "Deep exemplar-based video colorizaion", IEEE Computer Vision Pattern Recognition (CVPR), Long Beach, CA, USA, pp. 8052-8061, Jun.16-20, 2019.

[31] S. H. Lee and N. I. Cho, "Intra prediction method based on the linear relationship between the channels for YUV 4:2:0 intra coding", IEEE International Conference on Image Processing (ICIP), Cairo, Egypt, pp. 1037-1040, Nov.7-10, 2009.

[32] K. Zhang, J. Chen, L. Zhang, X. Li, and M. Karczewicz, "Enhanced cross component linear model for chroma intra prediction in video coding", *IEEE Trans. Image Process.*, vol. 27, no. 8, pp. 3983-3997, Aug. 2018.

[33] L. Zhang, W. Chien, J. Chen, X. Zhao, and M. Karczewicz, "Multiple direct mode for intra coding", IEEE Visual Communications and Image Processing (VCIP), St. Petersburg, Florida, USA, Dec.10-13, 2017.

[34] W. Kim, W. Pu, A. Khairat, M. Siekmann, J. Sole, J. Chen, M. Karczewicz, T. Nguyen, and D. Marpe, "Cross component prediction in HEVC", *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 6, pp. 1699-1708, Jun. 2020.

[35] T. Nguyen, A. Khairat, D. Marpe, M. Siekmann, and T. Wiegand, "Extended cross component prediction in HEVC", Picture Coding Symposium (PCS), Cairns, Australia, pp. 164-168, May 31 - Jun. 3, 2015.

[36] A. Khairat, T. Nguyen, M. Siekmann, D. Marpe, and T. Wiegand, "Adaptive cross component prediction for 4:4:4 high efficiency video coding", IEEE International Conference on Image Processing (ICIP), Pairs, France, pp. 3734-3738, Oct. 27-30, 2014.

[37] C. Yeo, Y. Tan, Z. Li, and S. Rahardja, "Chroma intra prediction using template matching with reconstructed luma components", IEEE International Conference on Image Processing (ICIP), Brussels, Belguim, pp. 1637-1640, Sept. 11-14, 2011.

[38] X. Zhang, C. Gisquet, E. Francois, F. Zou, and O. C. Au, "Chroma intra prediction based on inter channel correlation for HEVC", *IEEE Trans. Image Process.*, vol. 23, no. 1, pp. 274-286, Jan. 2014.

[39] L. N. Trudeau, N. E. Egge, and D. Barr, "Predicting chroma from luma in AV1", Data Compression Conference (DCC), Snowbird, Utah, USA, pp. 374-382, Mar. 27-30, 2018.

[40] M. Meyer, J. Wiesner, J. Schneider, and C. Rohlfing, "Convolutional neural networks for video intra prediction using cross-component adaptation", IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, United Kingdom, May 12-17, 2019.

[41] M. Blanch, S. Blasi, A. Smeaton, N. E. OConnor, and M. Mrak, "Chroma intra prediction with attention based CNN architectures", IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, Oct.25-28, 2020.

[42] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising", *IEEE Trans. Image Process.*, vol. 26, no.7, pp. 3142-3155, Jul. 2017.

[43] S. Iizuka, E. S.-Serra and H. Ishikawa, "Globally and locally consistent image completion", *ACM Trans. on Graph.*, vol. 36, no. 4, pp. 107:1-107:14, Jul. 2017.

[44] G. Schaefer and M. Stich, "UCID: an uncompressed color image database", in Proceedings of SPIE: Storage and Retrieval Methods and Applications for Multimedia 2004, Eds. M. M. Yeung, R. W. Lienhart, and C.-S. Li, vol. 5307, pp. 472-480, 2003.

[45] R. Timofte, *et al.*, "NTIRE 2017 challenge on single image superresolution: Methods and results", IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Honolulu, HI, USA, pp. 1110-1121, Jul. 21-26, 2017.

[46] F. Bossen, X. Li, K. Suehring, and A. Norkin, "AHG report: test model software development (AHG3)", Joint Video Exploration Team (JVET) of ITU-T SG 16 WP 3 and ISO/IEC JTC 1/SC 29/WG 11, Tech. Rep. Doc. JVET-N0003-v1, Geneva, CH, Mar.19-27, 2019.

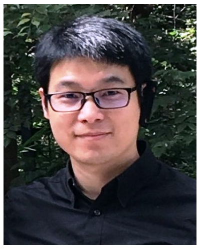
[47] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves", *ITU-T Video Coding Experts Group (VCEG)*, doc. M33, Austin, TX, 2001.

[48] R. Zhang, P. Isola, and A. A. Eforos, "Colorful image colorization", European Conference on Computer Vision (ECCV), Amsterdam, The Netherlands, Oct. 8-16, 2016.

[49] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. Berg, and F. Li, "Imagenet large scale visual recognition challenge", Int. J. Comput. Vis., vol. 115, pp. 211-252, Apr. 2015.

[50] Z. Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli, "Image quality assessment: from error visibility to structural similarity", *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600-612, Apr. 2014.

Linwei Zhu received the B.S. degree in applied physics from Tianjin University of Technology, China, in 2010, the M.S. degree in signal and information processing from Ningbo University, China, in 2013, and the Ph.D. degree from the Department of Computer Science, City University of Hong Kong, Hong Kong, China, in 2019. Now, he is a Postdoctoral Fellow with Shenzhen Institutes of Advanced Technology (SIAT), Chinese Academy of Sciences (CAS). His research interests mainly include depth image based rendering, depth estimation and machine learning based video coding/transcoding.



Yun Zhang (M'12-SM'16) received the B.S. and M.S. degrees in electrical engineering from Ningbo University, Ningbo, China, in 2004 and 2007, respectively, and the Ph.D. degree in computer science from Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS), Beijing, China, in 2010. From 2009 to 2014, he was a Visiting Scholar with the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong. From 2010 to 2017, he was an Assistant Professor and an Associate Professor in Shenzhen Institutes of Advanced Technology (SIAT), CAS, where he is currently a Professor in SIAT, CAS, Shenzhen, China. His research interests are video compression, 3D video processing, visual perception and machine learning.

Shiqi Wang (M'15) received the B.S. degree in computer science from the Harbin Institute of Technology, Harbin, China, in 2008, and the Ph.D. degree in computer application technology from the Peking University, Beijing, China, in 2014. From March 2014 to March 2016, he was a Postdoc Fellow with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON, Canada. From April 2016 to April 2017, he was with the Rapid-Rich Object Search Laboratory, Nanyang Technological University, Singapore, as a Research Fellow. He is currently an Assistant Professor in the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong. He has proposed more than 30 technical proposals to ISO/MPEG, ITU-T, and AVS standards. His research interests include image/video compression, analysis and quality assessment.

Sam Kwong (F'13) received the B.S. and M.S. degrees in electrical engineering from State University of New York at Buffalo in 1983, University of Waterloo, Waterloo, ON, Canada, in 1985, and the Ph.D. degree from University of Hagen, Germany, in 1996. From 1985 to 1987, he was a Diagnostic Engineer with Control Data Canada. He joined Bell Northern Research Canada as a Member of Scientific Staff. In 1990, he became a Lecturer at the Department of Electronic Engineering, City University of Hong Kong, where he is currently a Professor at the Department of Computer Science, City University of Hong Kong, Kowloon, Hong Kong. His research interests are video and image coding and evolutionary algorithms.

Xin Jin (S'03-M'09-SM'11) received the M.S. degree in communication and information system and the Ph.D. degree in information and communication engineering, from the Huazhong University of Science and Technology, Wuhan, China, in 2002 and 2005, respectively. From 2004 to 2005, she was an Intern with the Internet Multimedia Group, Microsoft Research Asia, Beijing, China. From 2006 to 2008, she was a Postdoctoral Fellow with The Chinese University of Hong Kong. From 2008 to 2012, she was a Visiting Lecturer with the Information Technology Research Organization, Waseda University, Fukouoka, Japan. Since March 2012, she has been with Shenzhen International Graduate School, Tsinghua University, Shenzhen, China, where she is currently a Professor. Her current research interests include power-constrained video processing and compression, computational imaging, and multimedia cloud computing.

Yu Qiao (SM'13) received the Ph.D. degree from the University of Electro-Communications, Japan, in 2006. He was a JSPS Fellow and a Project Assistant Professor with The University of Tokyo from 2007 to 2010. He is currently a Professor with the Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences. He has authored over 180 articles in journals and conferences, including PAMI, IJCV, TIP, ICCV, CVPR, ECCV, and AAAI. His research interests include computer vision, deep learning, and intelligent robots. He was a recipient of the Lu Jiaxi Young Researcher Award from the Chinese Academy of Sciences in 2012. He was the winner of video classification task in the ActivityNet Large Scale Activity Recognition Challenge 2016 and the first runner-up of scene recognition task in the ImageNet Large Scale Visual Recognition Challenge 2015.