Multimedia Tools and Applications (2023) 82:42803–42816
https://doi.org/10.1007/s11042-023-15409-7

Efficient feature coding based on performance analysis of Versatile Video Coding (VVC) in Video Coding for Machines (VCM)

Jin Young Lee[1] · **Yongho Choi**[1] · **The Van Le**[1] · **Kiho Choi**[2,3]

Received: 6 June 2022 / Revised: 1 December 2022 / Accepted: 18 April 2023 /
Published online: 24 April 2023

Abstract
Conventional video coding standards offer efficient compression of traditional 2D images. In particular, versatile video coding (VVC), which is the latest video coding standard, achieves very high compression efficiency, while maintaining high visual quality for humans. On the other hand, video coding for machines (VCM), which is developed as a new style of a video coding standard, mainly targets efficient compression of features extracted from deep neural networks. It generally employs VVC for feature coding. However, since VVC was developed for traditional images, an influence of the VVC based feature coding on VCM is not clear. Therefore, this paper proposes efficient tool combination by analyzing performance of VVC coding tools for the VCM feature coding, and then applies it into video captioning, which automatically generates natural language descriptions from videos. Experimental results show that the proposed tool combination is very efficient, in terms of coding performance and encoding complexity.

Keywords Versatile video coding (VVC) · Video coding for machines (VCM) · Video captioning

1 Introduction

Video data has been traditionally created for human entertainment and stored as a bitstream. It is generally broadcasted or transmitted to customers' terminals through broadcasting networks or Internet. Since video coding technologies give a strong influence on video quality

It has not been published elsewhere and that it has not been submitted simultaneously for publication elsewhere.

✉ Kiho Choi
aikiho@khu.ac.kr

[1] Department of Intelligent Mechatronics Engineering, Sejong University, Seoul, Korea

[2] Department of Electronics Engineering, Kyung Hee University, Yongin-si, Korea

[3] Department of Electronics and Information Convergence Engineering, Kyung Hee University, Yongin-si, Korea

that customers can view, how well high-quality videos are compressed and delivered into consumers becomes a very important issue. Because of this tendency, video coding technologies have doubled coding performance every ten years through a standardization work of many organizations. For example, ITU-T video coding experts group (VCEG) and ISO/IEC moving picture experts group (MPEG) are two major organizations that have worked for a long time to develop video coding standards providing high coding performance with higher resolution, higher quality, and higher frame rate. Thanks to their efforts, a versatile video coding (VVC) standard [43], which is the latest standard, was finalized in 2020.

Recently, as rapid development of video industries, the focus of data consumption is gradually shifting from humans into machines. For example, as many machine learning applications have been increasing in popularity and video sensors have been becoming more widely available, data consumption by various intelligent platforms with high requirements, such as connected vehicles, video surveillance, and smart city, becomes main data traffic in video data [23]. Connected vehicles include many devices to transmit and receive video data each other for in-car connectivity. Video surveillance has been monitored by humans, but various intelligent tasks, such as object detection and tracking, are currently machine-monitored. Smart city is also the important application that consumes massive volumes of data. Traffic monitoring and flow prediction would be the use cases in the smart city. Hence, a new video coding standard suited for machines are urgently needed.

To satisfy demands of efficient compression for the machine applications, MPEG has recently begun developing and standardizing video coding for machines (VCM), which is a new type of a standard. The VCM standard targets the standardization of bitstream formats and compression technologies for video data consumption by machines, rather than humans [23]. Unlike traditional 2D video coding standards, VCM deals with features, which represent interesting and meaningful parts in videos. They are generally extracted from deep neural networks, and then converted into feature maps to directly employ the VVC standard for the transmission between machines [24]. It is very difficult for humans to understand the features, whereas machines usually operate them for various vision tasks, such as object detection and tracking, segmentation, and video captioning. Figure 1 depicts examples of the feature maps achieved from three consecutive video frames through Inception-v4 [40]. It can be observed that pixel characteristics of the feature maps are very different from those of the traditional videos.

However, since VVC was developed for normal 2D images, an impact of VVC coding tools on VCM has never been fully evaluated. In this paper, we analyze performance of each VVC coding tool and propose the optimized tool combination for the efficient feature coding. For experimental tests, we evaluate the VVC tools on a video captioning scenario, which has been actively studied in an intersection of natural language processing

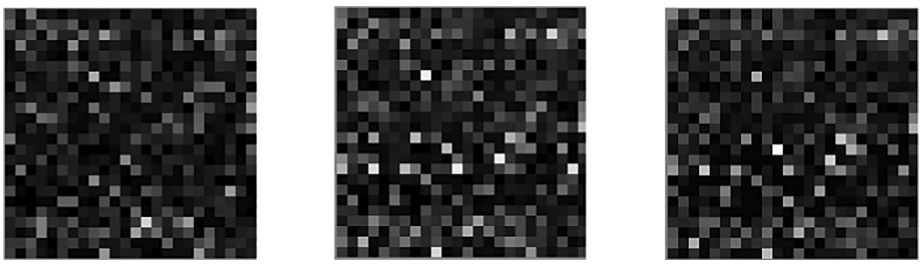

Fig. 1 Feature maps achieved from three consecutive video frames

and computer vision fields. The video captioning generates natural language descriptions from comprehensive understanding of videos. It is very useful in various applications, such as video retrieval and surveillance. Thanks to recently rapid progresses of deep learning techniques, lots of video captioning networks have been developed. [1, 5, 9, 35, 47]. In general, a combination of a convolutional neural network (CNN) [27] and a long short term memory (LSTM) [21] is widely employed for high captioning performance. CNN extracts features from frames within a video and LSTM generates sentences through the extracted features. In our scenario, CNN is performed in a sender side, and the CNN features are compressed with VVC. A receiver directly feeds the decompressed features into LSTM. To our knowledge, this is the first studies that not only evaluate the VVC coding tools for the VCM feature coding but also apply the VCM standard into the video captioning.

The remainder of this paper is organized as follows. Section 2 shows an overview of the VVC standard in detail. Section 3 analyzes the performance of each VVC coding tool and proposes the optimized tool combination for the efficient feature coding in the VCM standard. In Sect. 4, both the coding performance and the encoding complexity are evaluated, when the VCM standard is employed for the video captioning. Finally, Sect. 5 concludes this paper.

2 Overview of the VVC standard

VVC is the latest video coding standard, which was released in October 2020. According to its verification evaluations [3], VVC reduces bitrates by half, while keeping the same visual quality as a high efficiency video coding (HEVC) standard [20], which is one of the successful standards. HEVC is very useful for not only a standard dynamic range (SDR) content but also a high dynamic range (HDR) content [23, 48]. To outperform the superior predecessor, VVC uses many advanced tools with a flexible block partitioning structure that accurately represents diverse video properties to improve its coding efficiency. The partition candidates are exhaustively checked and compared to efficiently divide a coding unit (CU). Intra and inter prediction tools are used within the basic CU, followed by transform, quantization, in-loop filtering, and entropy coding with additional information for transmission [22].

Intra prediction is enhanced with 93 directional angles to provide the accurate prediction with non-square block sizes. Advanced intra tools, such as position-dependent prediction combination (PDPC) [41], a multiple reference line (MRL) [6], matrix based intra prediction (MIP) [19], a cross-component linear model (CCLM) [31], and intra sub-partition (ISP) [15], are performed to get closer to original pixel values in the prediction. Overall, by extending existing tools and using new tools, the intra coding performance has been significantly improved.

Many inter prediction tools were adopted to improve coding performance by reducing temporal duplication between sequential frames. In general, inter tools are classified into two groups, based on whether motion information is shared across an entire block or not. For example, history based motion vector prediction (HMVP) [51], merge with motion vector difference (MMVD) [25], symmetric motion vector difference (SMVD) [8], adaptive motion vector resolution (AMVR) [10], geometric partitioning mode (GPM) [17], bi-prediction with CU-level weights (BCW) [39], and combined intra and inter prediction (CIIP) [12] are whole block based inter prediction tools. On the other hand, affine motion (Affine) [32], sub-block based temporal motion vector prediction (SbTMVP) [9], decoder-side motion vector refinement (DMVR) [38], bi-directional optical flow (BDOF) [2], and

prediction refinement with optical flow (PROF) [18] are sub-block based tools. These tools improve motion vectors or refine predicted values to improve the prediction accuracy. Based on their comparison, VVC finds the best inter prediction.

For efficient transform and quantization of block residuals, multiple transform selection (MTS) [14], sub-block transform (SBT) [52], non-separable secondary transform (LFNST) [28], and dependent quantization (DQ) [37] were introduced in VVC. As new filters for in-loop filtering [26], an adaptive loop filter (ALF) and luma mapping with chroma scaling (LMCS) are used with conventional filters, such as a deblocking filter (DBK) and a sample adaptive offset (SAO).

All of these coding tools in the inter and intra prediction, transform, quantization, and in-loop filtering contribute the high coding performance through extensive encoding processes using an optimal rate-distortion (RD) based tool evaluation. It was reported that VVC improves the coding performance by about 25% and 36% in all intra (AI) and random access (RA) configurations on average, respectively, compared to HEVC [11]. However, the coding performance was evaluated with traditional images. It is possible that some tools are not efficient in the feature coding. Even though MPEG plans to use VVC for the feature coding [24], an impact of each VVC tool on VCM has never been thoroughly investigated. Hence, in this paper, the VVC tools are mainly analyzed for the VCM feature coding, in terms of the coding performance and encoding complexity.

3 Performance analysis of VVC

In this section, the VVC coding tools are analyzed for the feature coding. Based on the analysis, we propose the optimized tool combination from the perspective of the VCM effectiveness. For the tests, we used Microsoft Research Video Description Corpus (MSVD) [7], which consists of 1,970 YouTube short video clips. Features of each video frame were extracted with Inception-v4 [40]. The 1D feature vectors were converted into the 2D feature maps, and then compressed with a reference software VTM12.0 [45]. All coding parameters follow common test conditions (CTC) [4], which were employed for the standardization of VVC. Quantization parameters (QPs) of 22, 27, 32, and 37 were used to measure the overall coding performance in a high bitrate condition, and QPs of 27, 32, 37, and 42 were employed in a low bitrate condition. Coding performance and encoding complexity were measured with Bjøntegaard delta bitrates (BDBR) [5] and encoding time saving (ETS) in percentage, respectively. ETS is calculated as follow,

$$ETS = \frac{ET(T) - ET(O)}{ET(O)} \times 100 \tag{1}$$

where *ET(O)* and *ET(T)* indicate the encoding times when using the original tools and the tested tools, respectively. Some tools, such as PDPC and HMVP, could not be tested, because they are not separately controlled in the tool configuration. In addition, CCLM that enhances the prediction of chroma components was not analyzed, because there are only luma components in the feature maps.

3.1 Intra tools

Tables 1 and 2 show the intra coding performance in AI and RA, respectively. In the tables, on and off mean tool on and off on top of the reference setting, respectively, to evaluate

Table 1 Performance and complexity of intra tools in AI

Tools	On/Off	High		Low	
		BDBR	ETS	BDBR	ETS
ISP	Off	-0.02	-11.20	-0.03	-10.41
MRL	Off	0.94	1.92	1.01	0.37
MIP	Off	0.40	-14.74	0.48	-13.78
IBC	On	-0.01	9.85	-0.01	6.49
BDPCM	On	-0.09	2.92	-0.08	1.36

each coding tool [4]. In the CTC tool configuration [4], ISP, MRL, and MIP are only turned on. For example, ISP shows the meaningless performance, when it is turned off in both AI and RA. It divides CU vertically or horizontally into several sub-partitions, based on a block size. It works very well for traditional images, but sub-divisions on the feature map are inefficient, because of a low spatial correlation between neighboring feature values. MRL shows the performance of 0.94% and 1.01% at the high and low bitrate conditions in AI and 0.34% and 0.44% in RA, respectively. MRL uses multiple reference lines in the prediction. Its extended lines offer the more accurate prediction in the feature coding. MIP obtains the performance of 0.40% and 0.48% in AI and 0.06% and 0.13% in RA, respectively. Based on the linear interpolation method, it performs the prediction by using values interpolated between neighboring pixels. We also tested intra block copy (IBC) and block differential pulse code modulation (BDPCM) [46], which are screen content coding (SCC) tools of VVC, by turning on their corresponding options in the tool configuration. As illustrated in the tables, they give a small impact on the feature coding. Hence, the proposed tool combination only includes MRL and MIP providing the noteworthy coding performance, but the others are disabled to reduce the encoding complexity.

3.2 Inter tools

Table 3 shows the coding performance of the inter tools in RA, when each tool is disabled. As observed in Table 3, the performance shows -0.46% and -0.84% for MMVD, 0.05% and 0.06% for SMVD, 0.02% and 0.02% for AMVR, -0.01% and 0.00% for GPM, 0.79% and 0.61% for BCW, 0.96% and 0.81% for CIIP, -0.42% and -0.83% for Affine, 0.01% and 0.01% for SbTMVP, -0.41% and -0.85% for BDOF, -0.38% and -0.80% for PROF, and -0.38% and -0.79% for DMVR at the high and low bitrate conditions on average, respectively. Unfortunately, the inter tools do not provide the attractive performance

Table 2 Performance and complexity of intra tools in RA

Tools	On/Off	High		Low	
		BDBR	ETS	BDBR	ETS
ISP	Off	0.01	-5.96	0.02	-6.19
MRL	Off	0.34	-0.89	0.44	-0.87
MIP	Off	0.06	-7.91	0.13	-7.30
IBC	On	0.03	1.65	0.07	-0.24
BDPCM	On	-0.04	-1.98	-0.05	-2.95

Table 3 Performance and complexity of inter tools in RA

Tools	On/Off	High		Low	
		BDBR	ETS	BDBR	ETS
MMVD	Off	-0.46	-5.88	-0.84	-4.68
SMVD	Off	0.05	-2.70	0.06	-2.54
AMVR	Off	0.02	-1.68	0.02	-1.66
GPM	Off	-0.01	-6.09	0.00	-5.47
BCW	Off	0.79	0.54	0.61	0.74
CIIP	Off	0.96	0.96	0.81	1.12
Affine	Off	-0.42	-1.92	-0.83	-0.94
SbTMVP	Off	0.01	2.75	0.01	1.86
BDOF	Off	-0.41	-1.63	-0.85	-1.29
PROF	Off	-0.38	-0.74	-0.80	-1.72
DMVR	Off	-0.38	0.05	-0.79	-0.47

for the feature coding, except for BCW and CCIP. Most of the tools were developed to find an accurate motion vector between frames. MMVD, AMVR, and GPM improve the accuracy of the motion vector at the entire CU level. Affine, SbTMVP, BDOF, PROF, and DMVR improve at the sub-block level. However, as illustrated in Fig. 1, since a temporal correlation between feature maps is very low, the tools improving the accuracy become inefficient. Interestingly, BCW and CIIP perform well. BCW uses weighted averaging with preset weighted values for the bi-prediction, and CIIP combines the inter and intra prediction. It means that the tools refining the predicted values is relatively efficient for the feature coding, rather than those improving the motion vector. Therefore, the proposed tool combination only includes BCW and CIIP, but the others are disabled to reduce the encoding complexity.

3.3 Other tools

Tables 4 and 5 show the coding performance of the transform, quantization and in-loop filtering tools in AI and RA, respectively. In the tables, MTS, SBT, and LFNST show the negligible loss, but TS gives a very high impact on the feature coding. For example, it

Table 4 Performance and complexity of other tools in AI

Tools	On/Off	High		Low	
		BDBR	ETS	BDBR	ETS
MTS	Off	0.00	-14.17	-0.04	-9.62
LFNST	Off	-0.01	2.19	0.09	0.36
TS	Off	5.69	-5.52	10.45	-0.63
DQ	Off	1.44	-16.85	0.96	-15.72
DBK	Off	0.02	-3.04	-0.16	-6.53
SAO	Off	0.21	-0.08	0.16	-0.92
ALF	Off	0.00	-31.62	0.00	-44.83
LMCS	Off	-2.98	-2.42	-5.06	-2.85

Table 5 Performance and complexity of other tools in RA

Tools	On/Off	High		Low	
		BDBR	ETS	BDBR	ETS
MTS	Off	-0.01	-11.48	-0.01	-8.81
SBT	Off	-0.09	-4.95	-0.06	-4.82
LFNST	Off	0.00	-9.60	0.01	-7.47
TS	Off	4.01	-13.37	4.03	-11.32
DQ	Off	1.29	-10.10	0.88	-7.79
DBK	Off	-0.13	-1.01	-0.29	-1.71
SAO	Off	0.03	-1.16	-0.64	-1.54
ALF	Off	-0.39	-27.05	-0.82	-45.10
LMCS	Off	-0.39	2.16	-1.11	2.12

shows the performance of 0.00% and -0.04% at the high and low bitrate conditions in AI and -0.01% and -0.01% in RA for MTS, -0.09% and -0.06% in RA for SBT, -0.01% and 0.09% in AI and 0.00% and 0.01% in RA for LFNST, and 5.69% and 10.45% in AI and 4.01% and 4.03% in RA for TS on average, respectively. The transform coding normally redistributes energy by converting block values into a frequency domain. However, since feature values are randomly distributed, energy compaction is ineffective, even when different conversion kernels are applied. Hence, the tools that skip the transform are much more efficient than the advanced transform tools. Based on this observation, the proposed tool combination only includes TS.

DQ is the tool allowing the adaptive quantization by switching between two scalar quantizers with a pre-determined state machine. It provides the high performance of 1.44% and 0.96% at the high and low bitrate conditions in AI and 1.29% and 0.88% at the high and low bitrate conditions in RA on average, respectively. Hence, DQ is included in the proposed tool combination.

Some in-loop filtering tools affect the coding performance a lot, when the tool is disabled. For example, it shows the performance of 0.02% and -0.16% at the high and low bitrate conditions in AI and -0.13% and -0.29% in RA for DBK, 0.21% and 0.16% in AI and 0.03% and -0.64% in RA for SAO, 0.00% and 0.00% in AI and -0.39% and -0.82% in RA for ALF, and -2.98% and -5.06% in AI and -0.39% and -1.11% in RA for LMCS. DBK, SAO, and ALF, which were developed to reduce artifacts and minimize reconstruction error, are not suitable for the feature maps containing random textures. LMCS improves the coding performance by changing input pixel values, based on preset tables and linear mapping. However, since the linear mapping is optimized on traditional images, the significantly high coding loss is observed in the feature coding. Hence, all of the in-loop filtering tools are not recommended in the proposed tool combination, in terms of the coding performance and the encoding complexity.

4 Evaluations

We applied the proposed VVC tool combination in VCM into the video captioning. In the tests, MSVD [7] was used. This dataset consists of 1,200 training clips, 100 validation clips, and 670 testing clips. Each video clip has about 40 English sentences for a single

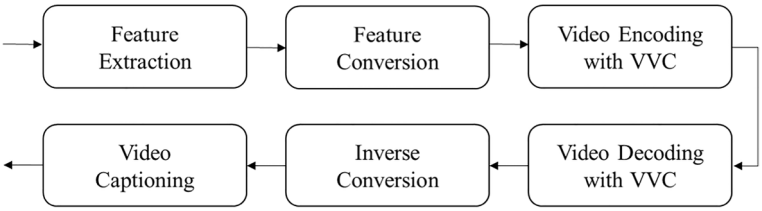

Fig. 2 Overall architecture for the evaluation of the VCM feature coding

activity. First, features of all frames were extracted from Inception-v4 [40]. Second, each feature vectors were converted into the 2D feature maps. Third, they were compressed with the proposed VVC tool combination. Next, they were decompressed and inversely converted into the feature vectors. Finally, the video captioning was performed with the reconstructed features. Figure 2 shows an overall architecture. For the video captioning, we used S2VT [42], which was designed with the combination of one CNN and two LSTM layers, as shown in Fig. 3. In our tests, the CNN features were compressed with VVC, and then the decompressed features were used in the first LSTM.

Table 6 shows the coding performance of the proposed tool combination in AI and RA. Based on the exhaustive tests, the proposed tool combination only includes MRL, MIP, CIIP, BCW, TS, and DQ. The other tools are disabled. As illustrated in Table 6, although many tools are disabled, the coding improvement is obtained. For example, the proposed tool combination has the coding gain of 2.71% and 5.26% at the high and low bitrate conditions in AI and 0.82% and 2.09% at the high and low bitrate conditions in RA on average, respectively. Because most of the VVC coding tools were mainly developed for traditional images, they have not been optimized in the feature coding. However, if VVC can select the efficient tools without any normative change, the VCM standard can be easily and early realized. In addition, the encoding complexity can be reduced by 63.30% and

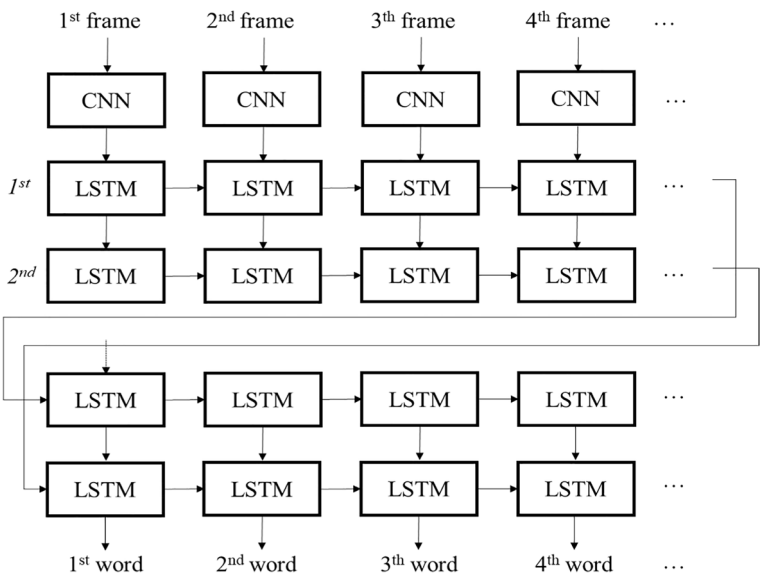

Fig. 3 S2VT video captioning network

Table 6 Performance and complexity of the proposed tool combination in AI and RA

Setting	Condition	BDBR	ETS
AI	High	-2.71	-63.30
	Low	-5.26	-69.70
RA	High	-0.82	-68.31
	Low	-2.09	-77.11

69.70% at the high and low bitrate conditions in AI and 68.31% and 77.11% in RA on average, respectively. Since machines are required to operate the video data with real time in intelligent platforms, the excessively high encoding complexity should be avoided. In this aspect, the proposed tool combination contributes to the complexity reduction for real-time processing.

Figures 4 and 5 show a rate-performance (RP) curve, when the feature coding is performed at QPs of 22, 27, 32, 37, and 42 in AI and RA, respectively. The x-axis and y-axis indicate coding bitrate and captioning accuracy, respectively. The accuracy was calculated with four different evaluation matrices, such as BLEU4 [36], METEOR [16], ROUGE-L [33], and CIDEr [44]. These matrices measure similarity between the original and predicted sentences in percentage. Here, the original sentence means the ground truth and the predicted sentence represents a sentence generated from the S2VT network, respectively. As depicted in Figs. 4 and 5, the RP curves of the methods using the original tools and

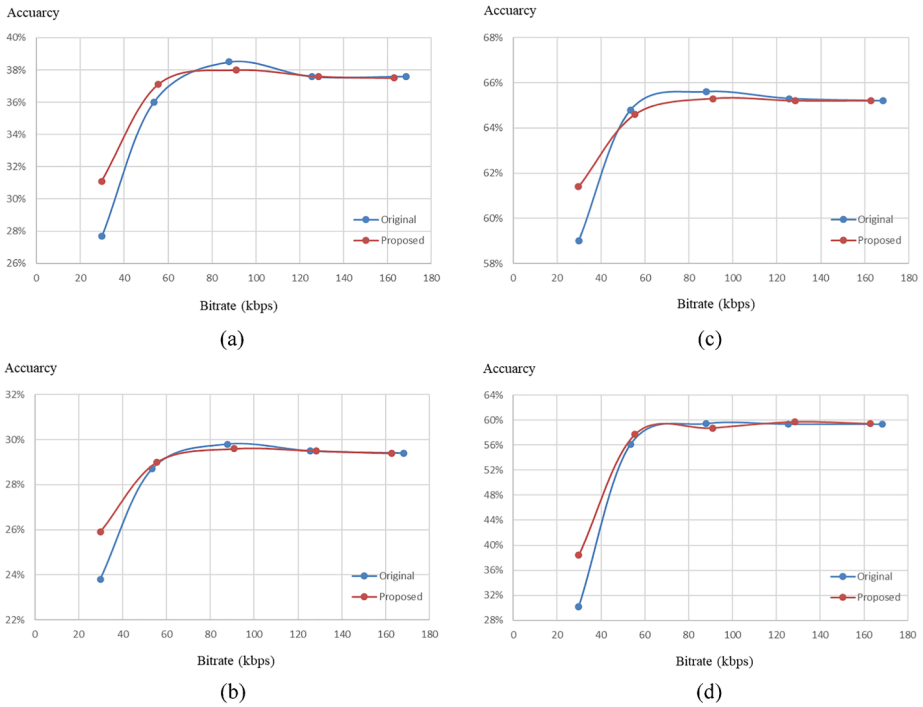

Fig. 4 RP curves of the proposed tool combination in AI, when the captioning matrices are (**a**) BLEU4, (**b**) METEOR, (**c**) ROUGE-L, and (**d**) CIDEr, respectively

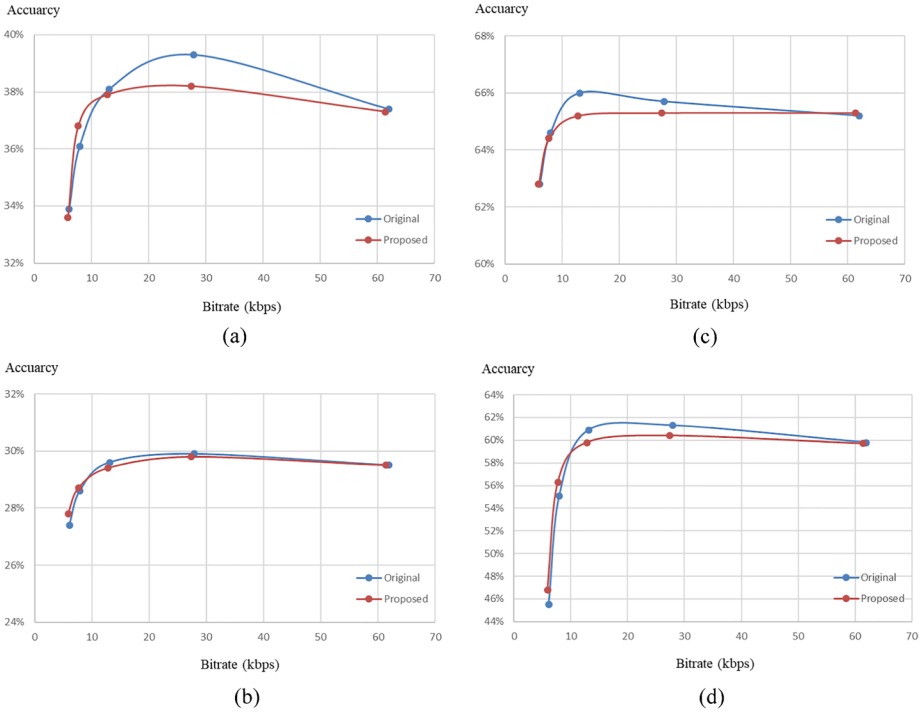

Fig. 5 RP curves of the proposed tool combination in RA, when the captioning matrices are (**a**) BLEU4, (**b**) METEOR, (**c**) ROUGE-L, and (**d**) CIDEr, respectively

the proposed tools are very similar for all the matrices. For example, at the same bitrate, a difference between their captioning accuracies is less than 3%. It indicates that the proposed tools are well-selected. It should be noted that the captioning accuracy is significantly reduced at the low bitrate condition, compared with the high bitrate condition. Since the very low QP, such as 42, usually degrades the reconstruction quality a lot, it affects the quality of the CNN features. Figure 6 illustrates examples of the original sentence and the sentences predicted from the features compressed with the five different QPs in AI and RA, respectively. As QP increases, the accuracy of the predicted sentence becomes very low. In order to solve this problem without any normative change in VVC, post-filtering techniques that can improve the quality of reconstructed features should be further studied.

Finally, we compared an essential video coding standard (EVC) [13], which is one of the recent video coding standards, with VVC. Since many EVC coding tools are similar to the VVC coding tools, we tested the video captioning with the frames reconstructed by the combination sets of the original VVC tools, the original EVC tools, and the proposed VVC tools, respectively. Tables 7 shows the captioning scores, when the feature coding was compressed at a target bitrate of 8 kbps in RA. As shown in Table 7, the proposed VVC tool combination provides the very competitive performance, compared with the full combination sets of the original VVC and EVC tools. Since the performance of the vision tasks, such as object detection and tracking, segmentation, and video captioning, usually depends on the quality of input images or videos, the result demonstrates that the selected

Original : A woman is applying makeup.
Predicted (QP=22): A woman is applying makeup.
Predicted (QP=27): A woman is applying makeup.
Predicted (QP=32): A woman is applying makeup.
Predicted (QP=37): A woman is applying shaving.
Predicted (QP=42): A man is putting a bug.

(a)

Original : A man is playing a guitar..
Predicted (QP=22): A man is singing and playing guitar.
Predicted (QP=27): A man is singing.
Predicted (QP=32): A man is singing on a stage.
Predicted (QP=37): A man is performing a violin.
Predicted (QP=42): A monkey is playing.

(b)

Fig. 6 Examples of the original and predicted sentences, when QPs of 22, 27, 32, 37, and 42 are used in (**a**) AI and (**b**) RA, respectively

VVC tools offering the relatively high coding performance, such as MRL, MIP, CIIP, BCW, TS, and DQ, are sufficient to maintain the accuracy of the video captioning.

5 Conclusion

In this paper, we thoroughly analyzed the VVC coding tools for the feature coding in the VCM standard. Since the VVC tools were mainly developed for traditional 2D images, the analysis was performed on the 2D feature maps, which are converted from the 1D feature vectors. Based on the exhaustive tests, the efficient tool combination that considers not only the coding performance but also the encoding complexity were proposed and the reasons why some coding tools provide the coding gain or loss for the feature coding were discussed. In addition, the features compressed with VVC employing the proposed tools were applied into the video captioning. The experimental results demonstrate that the proposed VVC tool combination is very efficient in the VCM standard. In future work, we are

Table 7 Captioning scores when the combination sets of the original VVC tools, the original EVC tools, and the proposed VVC tools are used for the feature coding, respectively

Combination	BLEU4	METEOR	ROUGE-L	CIDEr
Original VVC	36.1%	28.6%	64.6%	55.1%
Original EVC	35.9%	28.6%	64.6%	54.2%
Proposed VVC	36.8%	28.7%	64.4%	56.3%

planning to develop post-filtering techniques to improve the quality of the reconstructed features.

Acknowledgements This work was supported by the Institute of Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (IITP-2021-0-02067, IITP-2022-RS-2022-00156345) and the National Research Foundation of Korea(NRF) grant funded by the Korean government(MSIT) (NRF-2021R1F1A1060816).

Data availability Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Conflict of interest The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Aafaq N, Akhtar N, Liu W, Gilani SZ, Mian A (2019) "Spatio-temporal dynamics and semantic attribute enriched visual encoding for video captioning," in IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)
2. Alshin A and Alshina E (2016) "Bi-directional optical flow for future video codec," in Proc. Data Compress. Conf. (DCC)
3. Baroncini V and Wien M (2020) "VVC Verification Test Report for UHD SDR Video Content, document", JVET-T2020, ITU-T/ISO/IEC Joint Video Experts Team (JVET)
4. Bossen F, Boyce J, Suehring K, Li X, Seregin V (2019) "JVET common test conditions and software reference configurations for SDR video," ITU-T/ISO/IEC Joint Video Experts Team (JVET) JVET-N1010
5. Bjøntegaard G (2008) "Improvement of BD-PSNR Model", ITU-T SG16/Q6 VCEG-AI11
6. Bross B, Keydel P, Schwarz H, Marpe D, Wiegand T, Zhao L, Zhao X, Li X, Liu S, Chang Y-J, Jiang H-Y, Lin P-H, Kuo C-C, Lin C-C, Lin C-L (2018) "CE3: Multiple reference line intra prediction (Test 1.1.1, 1.1.2, 1.1.3 and 1.1.4)", JVET-L0283, ITU-T/ISO/IEC Joint Video Experts Team (JVET)
7. Chen DL, Dolan WB (2011) "Collecting highly parallel data for paraphrase evaluation," Association for Computational Linguistics, pp. 190–200
8. Chen H, Yang H, Chen J (2018) "Symmetrical Mode for Biprediction," JVET-J0063, ITU-T/ISO/IEC Joint Video Experts Team (JVET)
9. Chen H, Yang H, Chen J (2018) "CE4: Separate List for Sub-Block Merge Candidates (Test 4.2.8)", JVET-L0369, ITU-T/ISO/IEC Joint Video Experts Team (JVET)
10. Chen J, Chien W-J, Hu N, Seregin V, Karczewicz M, Li X (2016) "Enhanced Motion Vector Difference Coding," JVET-D0123, ITU-T/ISO/IEC Joint Video Exploration Team (JVET)
11. Chen W, Chen Y, Chernyak R, Choi K, Hashimoto R, Huang Y, Jang H, Liao R, Liu S (2020) "JVET AHG report: Tool reporting procedure (AHG13)," ITU-T/ISO/IEC Joint Video Experts Team (JVET) JVET-T0013
12. Chiang M-S, Hsu C-W, Huang Y-W, Lei S-M (2018) "CE10.1.1: Multi-hypothesis Prediction for Improving AMVP Mode, Skip or Merge Mode, and Intra Mode," JVET-L0100, ITUT/ISO/IEC Joint Video Experts Team (JVET)
13. Choi K, Chen J, Rusanovskyy D, Choi K-P, Jang ES (2020) An Overview of the MPEG-5 Essential Video Coding Standard. IEEE Signal Process Mag 37(3):160–167
14. Choi K, Piao Y, Kim C (2018) "CE6: AMT with reduced transform types (Test1.5)," JVET-K0171, ITUT/ISO/IEC Joint Video Experts Team (JVET)
15. De-Luxán-Hernández S, De-Luxán-Hernández S, George V, Ma J, Nguyen T, Schwarz H, Marpe D, Wiegand T (2019) "An intra subpartition coding mode for VVC," in Proceedings of IEEE Int. Conf. Image Process. (ICIP), pp. 1203–1207
16. Denkowski M, Lavie A (2014) "Meteor Universal: Language Specific Translation Evaluation for Any Target Language," Association for Computational Linguistics, pp. 376–380
17. Gao H, Esenlik S, Alshina E, Steinbach E (2021) Geometric Partitioning Mode in Versatile Video Coding: Algorithm Review and Analysis. IEEE Trans Circuits Syst Video Technol 31(9):3603–3617

18. He Y and Luo J (2019) "CE4–2.1: Prediction Refinement With Optical Flow for Affine Mode," JVET-O0070, ITUT/ISO/IEC Joint Video Experts Team (JVET)
19. Helle P, Pfaff J, Schäfer J, Rischke R, Schwarz H, Marpe D, and Wiegand T (2019), "Intra Picture Prediction for Video Coding with Neural Networks," In Proc. Data Compression Conference 2019
20. High Efficient Video Coding (HEVC) (2013) ITU-T Recommendation H.265 and ISO/IEC 23008–2
21. Hochreiter S, Schmidhuber J (1998) Long short-term memory. Neural Comput 9(8):1735–1780
22. Huang Y-W, An J, Huang H, Li X, Hsiang S-T, Zhang K, Gao H, Ma J, Chubach O (2021) Block partitioning structure in the VVC standard. IEEE Trans Circuits Syst Video Technol 31(10):3818–3833
23. ISO/IEC JTC1/SC 29/WG2, N0190 (2022) Use Cases and Requirements for Video Coding for Machines
24. ISO/IEC JTC1/SC 29/WG2, N0193 (2022) Evaluation Framework for Video Coding for Machines
25. Jeong S, Park MW, Piao Y, Park M, Choi K (2018) "CE4: Ultimate Motion Vector Expression (Test 4.5.4)," JVET-L0054, ITU-T/ISO/IEC Joint Video Experts Team (JVET)
26. Karczewicz M, Hu N, Taquet J, Chen C, Misra K, Andersson K, Yin P, Lu T, François E, Chen J (2021) VVC In-Loop Filters. IEEE Trans Circuits Syst Video Technol 31(10):3907–3925
27. Krizhevsky A, Sutskever I, Hinton GE (2012) "Imagenet classification with deep convolutional neural networks," Neural Information Processing Systems, pp. 1106–1114
28. Koo M, Salehifar M, Lim J, Kim S-H (2019) "Low frequency nonseparable transform (LFNST)," in Proc. Picture Coding Symp. (PCS)
29. Lee JY (2019) Deep multimodal embedding for video captioning. Multimed Tools Appl 78(22):31793–31805
30. Lei Z, Huang Y (2021) Video captioning based on channel soft attention and semantic reconstructor. Future internet 13(2):55
31. Li J, Wang M, Zhang L, Zhang K, Wang S, Wang S, Ma S, Gao W (2020) "Sub-Sampled Cross-Component Prediction for Chroma Component Coding," In Proc. Data Compression Conference
32. Li L, Li H, Liu D, Li Z, Yang H, Lin S, Chen H, Wu F (2018) "An efficient four-parameter affine motion model for video coding. IEEE Trans Circuits Syst Video Technol 28(8):1934–1948
33. Lin C-Y (2004) "ROUGE: A Package for Automatic Evaluation of Summaries," Association for Computational Linguistics, pp, 74–81
34. Nabati M, Behrad A (2020) Multi-sentence video captioning using content-oriented beam searching and multi-stage refining algorithm. Inf Process Manag 57(6):102302
35. Pan Y, Yao T, Li H, Mei T (2017) "Video captioning with transferred semantic attributes," in IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
36. Papineni K, Roukos S, Ward T, Zhu W-J (2002) "BLEU: a method for automatic evaluation of machine translation," Association for Computational Linguistics, pp. 311–318
37. Schwarz H, Nguyen T, Marpe D, Wiegand T (2018) "CE7: Transform Coefficient Coding and Dependent Quantization (Tests 7.1.2, 7.2.1)," JVET-K0071, ITUT/ISO/IEC Joint Video Experts Team (JVET)
38. Sethuraman S (2019) "CE9: Results of DMVR Related Tests CE9.2.1 and CE9.2.2," JVET-M0147, ITUT/ISO/IEC Joint Video Experts Team (JVET),
39. Su Y-C, Chen C-Y, Huang Y-W, Lei S-M, He Y, Luo J, Xiu X, Ye Y (2018) "CE4-related: Generalized Bi-prediction Improvements Combined from JVET-L0197 and JVET-L0296," JVET-L0646, ITU-T/ISO/IEC Joint Video Experts Team (JVET)
40. Szegedy C, Ioffe S, Vanhoucke V, and Alemi A (2016) "Inception-v4, Inception-ResNet and the impact of residual connections on learning," arXiv [cs.CV]
41. Van der Auwera G, Heo J, Filippov A (2018) "CE3: Summary Report on Intra Prediction and Mode Coding," JVET-J0023, ITU-T/ISO/IEC Joint Video Experts Team (JVET)
42. Venugopalan S, Rohrbach M, Donahue J, Mooney R, Darrell T, Saenko en K (2015) "Sequence to sequence-video to text", in Proceedings of the IEEE international conference on computer vision
43. Versatile Video Coding (VVC) (2020) ITU-T Recommendation H.266 and ISO/IEC 23090–3
44. Vedantam R, Zitnick CL, Parikh D (2015) "CIDEr: Consensus-based Image Description Evaluation," IEEE Conference on Computer Vision and Pattern Recognition, pp. 4566–4575
45. VVC Reference Software. https://vcgit.hhi.fraunhofer.de/jvet/VVCSoftware_VTM/-/tags/.
46. Xu X, Li X, Liu S (2019) "Intra block copy in Versatile Video Coding with Reference Sample Memory Reuse," in Proc. Picture Coding Symp. (PCS)
47. Yan C et al (2020) STAT: Spatial-temporal attention mechanism for video captioning. IEEE Trans Multimedia 22(1):229–241
48. Zhang Y, Agrafiotis D, Bull DR (2013) "High Dynamic Range image & video compression a review," In Proc. International Conference on Digital Signal Processing (DSP)
49. Zhang Y, Naccari M, Agrafiotis D, Mrak M, Bull DR (2016) High Dynamic Range Video Compression Exploiting Luminance Masking. IEEE Trans Circuits Syst Video Technol 26(5):950–964

50. Zhang Y, Naccari M, Agrafiotis D, Mrak M, Bull DR (2013) "High dynamic range video compression by intensity dependent spatial quantization in HEVC," In Proc. Picture Coding Symposium (PCS)
51. Zhang L, Zhang K, Liu H, Wang Y, Zhao P, Hong D (2018) "CE4: History-based Motion Vector Prediction (Test 4.4.7)," JVET-L0266, ITU-T/ISO/IEC Joint Video Experts Team (JVET)
52. Zhao Y, Yang H, Chen J (2018) "CE6: Spatially Varying Transform (Test 6.1.12.1)," JVET-K0139, ITUT/ISO/IEC Joint Video Experts Team (JVET)