Fast CU Determination Algorithm Based on Convolutional Neural Network for HEVC

Takafumi Katayama, Tian song, Wen Shi, Xiantao Jiang, and Takashi Shimamoto

Abstract—**High efficiency video coding (HEVC) is the current video coding standard. HEVC achieved very high coding efficiency compared with previous video coding standards. However, the increasing of the computational complexity and the hardware implementation difficulty are the critical problems for HEVC. In this paper, we propose a fast coding unit (CU) size decision algorithm for HEVC based on convolutional neural network. The proposed fast algorithm contribute to decrease no less than two CU partition modes in each coding tree unit for full rate-distortion optimization processing, thereby reducing the encoder hardware complexity. Moreover, our algorithm only use the texture information and it does not depend on the correlations among CU depths or spatially nearby CUs. It is friendly to the parallel processing of RDO. The proposed algorithm is evaluated by the reference software of HEVC (HM16.7). The simulation results show that the proposed algorithm can achieve over 66.7% computation complexity reduction comparing to the original HEVC algorithm.**

Index Terms—**High efficiency video coding (HEVC), intra coding, convolutional neural network (CNN).**

I. Introduction

High Efficiency Video Coding (HEVC) was developed by Joint Collaborative Team on Video Coding (JCT-VC) [1], [2]. HEVC achieved double the coding efficiency as compared to the predecessor H.264/AVC, especially when processing the high-resolution sequence s(HD/UHD). However, this increases the computational complexity up by 10 times as compared to H.264/AVC. On the other words, the high computational complexity in HEVC becomes a hardware implementation bottle-neck. In HEVC, recursive coding unit (CU) size decision method occupy most of the computational complexity. Intra coding is particularly an important coding tool adopted in almost all mainstream video compression standard such as MPEG-2, H.264/AVC and HEVC.

Although the quad-tree structure enables each CU to be coded optimally and can greatly improves the encoding efficiency significantly, it imposes significant computational complexity on the encoder during the exhaustive rate-distortion cost calculation of total 85 CUs, where all possible combinations of CU, prediction unit (PU) and transform unit (TU) are tried to find the optimal combination. Thus, it is important to find a practical implementation of HEVC to reduce the complexity while maintaining its performance. To overcome this problem, a number of algorithms on accelerating the encoder of HEVC have been proposed to reduce the computational complexity.

To alleviate the intra encoding complexity, many algorithms have been developed for fast intra coding mode decision. The previous work can be classified into three categories.

1) The method of the first category reduces the RDO complexity of intra prediction mode in every CU depth. For example, Ma applied the rough mode decision scheme to reduce the number of candidate prediction modes, which will perform the RDO processing [3]. Zhu simplified the computation of rate and distortion estimation [4], [5]. These previous works contribute to the RDO complexity reduction.

2) The methods of the second category dynamically skip the CU depth decision process based on some preprocessing [6]-[8]. Similarly, another algorithms skip the early terminating the CU/PU depth RDO procedure based on the CU depth information of previously coded slices and neighboring CUs [9].

3) The method of the third category show a fast CU partitioning algorithm using machine learning which has been actively discussed in recent years. Some algorithms decide the optimal CU depth by using the convolutional neural network (CNN) [10], [11]. Because the pip-line processing of coding tree unit (CTU) is considered, these previous works is oriented for hardware implementation.

For the hardware encoder design, the first kinds of methods did not reduce the depth of CU/PU. On the other hand, the methods of the second category did not shrink the maximum complexity at the CTU. For example, in literature [9], in the parameter training stage, all CU levels must be searched with the exhaustive RDO. Moreover, in the third category, the inherent drawbacks of [11] induce 4.79% BD-BR increment. In fact, the optimal CU coding modes are determined by not only the edge information, the texture strength, and the quantization step, but also neighboring CU parameter. Therefore, our research focuses on the texture information of neighboring spatial blocks.

We consider that the coding performance is improved by the utilizing the neighboring blocks. However, the input texture of neighboring spatial blocks increase the computation complexity of CNN. In this paper, the fast CU depth decision is implemented by the optimal CNN architecture, that is specially devised to deal with 16x16 pixel block.

Manuscript received August 17, 2018; revised September 26, 2018.

Takafumi Katayama, Wen Shi, Tian Song, and Takashi Shimamoto are with the Department of Electrical and Electronics Engineering of Tokushima University, Tokushima city, Tokushima, 770-8506 Japan (e-mail: katayama@ee.tokushima-u.ac.jp, tiansong@ee.tokushima-u.ac.jp).

Xiantao Jiang is with the Department of Information Engineering Shanghai Maritime University, Pudong New Dist., Shanghai 201306, P. R. China (e-mail: xtjiang@shmtu.edu.cn).

doi: 10.18178/ijmlc.2018.8.5.726

II. All Intra Mode

The HEVC standard inherits the well-known blockbased hybrid coding architecture of H.264/AVC. However, in contrast to 16×16 pixels macro blocks (MB) used in H.264/AVC, it employs a flexible quad-tree coding block partitioning structure that enables the usage of large and multiple sizes of CU, PU and TU. One of the frame is divided into a sequence of CTUs and the maximum size allowed for the luma block in a CTU is specified to be 64×64. Each $2N\times 2N$ CUs which shares the same prediction mode can be divided into four smaller $N\times N$ CUs recursively until the maximum CU depth is reached. The sizes of CU range from 64×64 to 8x8. Furthermore, the number of intra prediction modes for each CU is also increased to 35. Therefore, the computation of CU candidate modes for a largest coding unit (LCU) is exhausted, as shown in Fig. 1. As a result, intra prediction in HEVC encoding is much more complicated than H.264/AVC. Therefore, it is effective for the computation complexity reduction to early determine the optimal CU size and prediction mode.

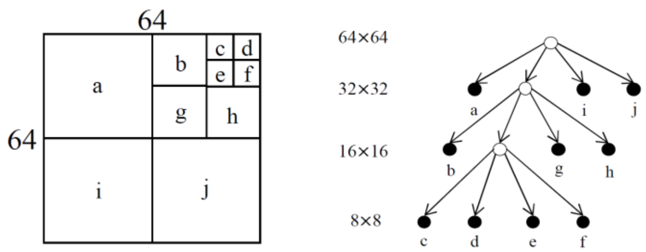

Fig. 1. CTU description.

III. Analysis of CNN for Fast Intra Coding

A. Verification of CNN Structure

Previous work proposed the hardware-oriented algorithm using CNN for HEVC encoder [11]. This work achieved high computational complexity reduction, and proved easy to implement hardware of CNN. However, the proposed algorithm induced BD-BR increase. Considering the encoding of super-resolution (4K), a more efficient CU decision algorithm by CNN is required for super-resolution encoder. For this reason, to clarify the optimal CNN structure including the convolutional layer (Conv), kernel, and full connection layer (FCL), we evaluate the relationship of the validity and the parameter. The evaluation is performed by using CNN structure of [11], and the block division of 32×32 is judged by single CNN process. CNN structure of [11] is shown as Fig. 2. The structure consists of two Conv, two max pooling, and two FCL. The parameters represent the number of Conv, kernel, and FCL. The sequence of ClassA and ClassB are used as the training sequences.

Table I show validation accuracy (Valid_acc) and training accuracy (Train_acc) when training of 20000 epoch is performed. Reference CNN represent simple and small network based on the CNN structure of [11]. To evaluate the accuracy with some conditions, the Conv, kernel, and FCL are added to the reference CNN. The evaluation result show about 70% Train_acc and 65% Valid_acc. From this evaluation result, it is observed that both accuracy are not greatly affected by the variation of the number of Conv, kernel, and FCL. In other words, in the simple structure, training and validation accuracy have the limit accuracy. Hence, for achieving higher accuracy performance, our approach adopts the CNN structure of the multiple inputs.

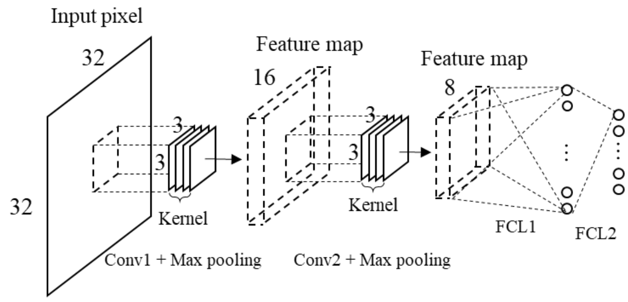

Fig. 2. Reference CNN structure.

TABLE I: Analyzing of Single Input CNN

	Number of parameter			Prediction accuracy(%)	
	Conv	Kernel	FCL	Valid_acc	Train_acc
Default parameter[11]	2	6	2	68.40	71.80
Comparison of Conv	3	6	2	65.00	71.10
	4	6	2	60.40	70.70
Comparison of kernel	2	8	2	49.90	65.10
	2	10	2	65.90	71.00
Comparison of FCL	2	6	3	63.50	70.90
	2	6	4	50.00	55.70

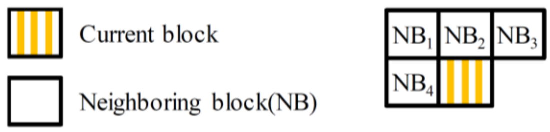

Fig. 3. Mapping neighboring block and current block.

In HEVC intra coding, best CU size depend on the complexity of the neighboring blocks. For this reason, the pixels of the neighboring blocks are important for high prediction accuracy. Fig. 3 shows the block position which is used as input to our proposed CNN. In our approach, only neighboring blocks are used for the input of CNN. Moreover, the prediction of division pattern in 32×32 block have high complexity. Considering the improvement of the division accuracy, the reduction of block division pattern is required. Therefore, our proposed CNN use 16×16 block as input texture. The CNN structure of multiple inputs that we considered from this evaluation result is shown in Fig. 4. To clarify the structure of more efficiency multiple inputs CNN, in the next subsection, the evaluation of the number of input and parameter is performed by using Fig. 4.

B. Evaluation of Multiple Inputs CNN

With the support of powerful computational devices, many deep learning networks become deeper. For example, a deep residual network is proposed for image recognition [12]. However, the deeper network model does not necessarily give the high accuracy. Therefore, for identifying the most suitable multiple inputs CNN, we evaluate the prediction accuracy according to the different of the number of inputs.

Variation of the number of the neighboring block and the number of Conv, kernel, FCL were evaluated with the sequence of ClassA and ClassB, as shown in Table II. Table II shows that the increasing of the neighboring block and kernel lead to improvement of accuracy. It is clear that the extraction of feature map by kernel is important for block division. On the other hands, about the variation of Conv and FCL, these parameters had little change compared to kernel variation.

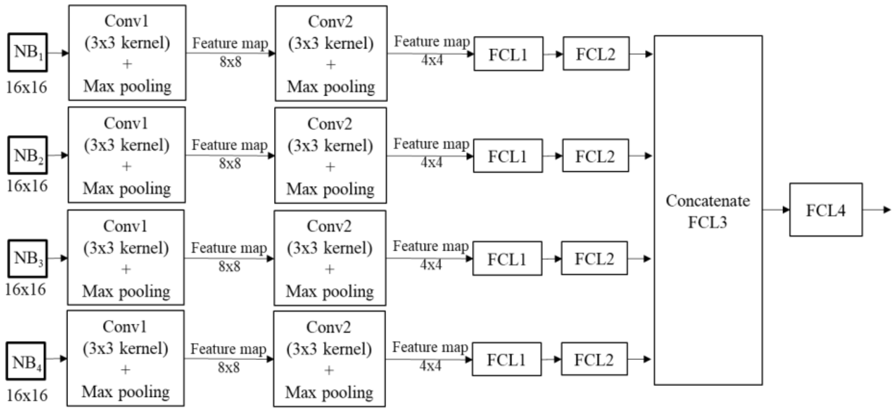

Fig. 4. Structure of 4-inputs CNN.

TABLE II: TRAIN AND VALIDATION ACCURACY (%) EVALUATION OF THE NEIGHBORING BLOCK AND THE PARAMETER VARIATION

	Number of parameter			Number of input			
	Conv	Kernel	FCL	NB2, NB4		NB1, NB2, NB3, NB4	
				Valid_acc	Train_acc	Valid_acc	Train_acc
Default parameter	2	6	2	68.40	73.40	69.40	76.80
Comparison of Conv	3	6	2	69.10	72.80	70.40	74.80
Comparison of kernel	2	8	2	73.20	78.10	75.80	80.20
	2	10	2	79.90	84.50	84.40	89.20
Comparison of FCL	2	6	3	66.80	68.80	67.20	71.80
	2	6	4	65.40	67.80	66.40	68.80

IV. PROPOSED ALGORITHM

A module with our CNN model is implemented and embedded in HM16.7 encoder software before intra prediction. The CNN classifier outputs the optimal CTU division information. The detail of flowchart is shown in Fig. 5. Compared with the conventional encoding processing, the encoding process using our proposed algorithm is not induce the many iteration for determining the optimal CU depth. The CU classification algorithm will be helpful to the computational complexity reduction of intra encoding. This means that the hardware area of RDO process in intra coding mode can be reduced.

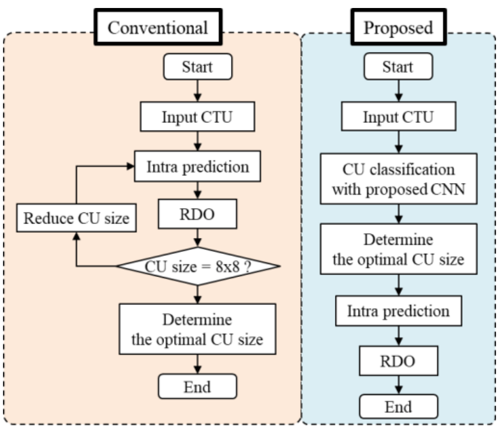

Fig. 5. Comparison of conventional flowchart and the proposed flowchart.

From evaluations previous section, our approach supplies the best performance condition to CNN model. In Table II, the comparison of Conv, Kernel, FCL are evaluated. Obviously, the number of input affect the prediction accuracy. Regarding Conv and FCL parameter, the prediction accuracy improved a little. As a possible reason for that increasing the Conv in 16×16 block is ineffective and many parameters in FCL increase the complexity of the prediction accuracy. On the other hand, the number of kernel increase the effective parameter for classification. Therefore, our proposed CNN consist of the multiple inputs using neighboring block, two Convs, two FCLs, and ten kernels, as shown in Fig. 6.

The input of CNN is the block partition patterns from 64×4 to 8×8 which are converted to 16×16 block. The first layer is a convolutional layer with ten kernels. Each neuron is connected to a 3×3 receptive field in the input. The size of the feature map is 8 × 8 and the convolution calculation is performed with zero padding mode. The kernels in this layer are deemed as feature extractors. The second layer performs the max pooling. Similarly, the third and fourth layer perform the convolution and max pooling. The fifth and sixth layer use FCL which the parameter is 256 and 64 to each input. The seventh layer concatenate FCL of each input with 256 parameters. The eighth layers perform FCL which the parameter is 64. The output layer use softmax units.

V. EXPERIMENTAL RESULTS AND DISCUSSION

For all of the evaluations, the coding structure which is set to all intra mode is used. The simulation environment is Intel(R) Core(TM) i7-4770 CPU 3.40GHz with 4 cores, RAM 8.00 GB and Windows 10 Home Edition 64-bit. Several test sequences (100 frames) with picture size from

Class 4K to Class B are used. The number of 50 frame in Class B are selected as the training samples i.e. BasketballDrive, BQTerrace, Cactus, Kimono, and ParkScene. The performance of our proposed complexity reduction is compared with that of the unmodified HM16.7 encoder in terms of execution time, and impact on bit-rate and peak signal to noise ratio (PSNR)[13]. The difference value of execution time between the unmodified HM16.7 and the proposed algorithm is represented as time saving (*TS*). *TS* is defined as

$$TS(\%) = \frac{T_{HM16.7} - T_{\text{proposed}}}{T_{HM16.7}} \tag{1}$$

where $T_{HM16.7}$ is the encoding time of the unmodified HM16.7 and $T_{Proposed}$ is that of the proposed algorithm. The computation complexity reduction (*CR*) is evaluated with QP from 22 to 37. *CR* is defined as

$$CR(\%) = \sum_{f=1}^{F} \sum_{j=1}^{J} \frac{\sum_{i=1} BestCU(i, j, f)}{J \times F \times 85} \times 100 \tag{2}$$

where $BestCU(i, j, f)$ indicates the *i*th best CU of the *j*th CTU of the *f*th frame of the test sequence. *J* and *F* are the total number of CTUs in each frame and total number of frames of the test video, respectively. 85 is represented by the number of iteration which is required for the best CU size decision from 64×64 to 8×8 in reference software HM16.7.

TABLE III: RESULT OF PROPOSED ALGORITHM COMPARED TO HM16.7

Class	Sequences	CR (%)	TS (%)	BD-rate (piecewise cubic) Y(%)	U(%)	V(%)
Class 4K	CampfireParty	69.5	62.0	1.5	1.6	1.7
	CatRobot	78.3	62.9	1.9	2.1	2.4
	DaylightRoad	62.8	62.4	2.6	2.6	2.0
	Drums100	77.9	66.1	1.6	1.9	0.9
	TrafficFlow	65.2	61.6	2.3	2.1	2.8
	Tango	68.2	62.7	1.7	1.6	1.9
	ToddlerFountain	70.3	62.5	1.3	1.4	1.7
	Rollercoaster	66.1	60.6	2.3	2.2	2.2
	Average	69.8	62.6	1.9	1.9	2.0
Class A	Traffic	70.4	68.7	1.4	1.9	1.8
	PeopleOnStreet	72.8	66.5	2.4	1.9	1.8
	Nebuta	72.5	67.1	1.5	1.7	1.8
	SteamLocomotive	71.4	64.1	1.8	0.6	1.1
	Average	71.8	66.6	1.8	1.5	1.8
Class B	BasketballDrive	68.3	72.1	1.8	1.3	0.8
	BQTerrace	66.1	73.3	1.8	0.4	0.1
	Cactus	68.9	71.8	2.0	1.6	1.7
	Kimono	72.5	75.8	1.7	1.5	1.7
	ParkScene	71.3	73.7	1.2	0.1	0.7
	Average	69.4	73.3	1.7	1.0	1.2
Average		70.1	66.7	1.8	1.6	1.6

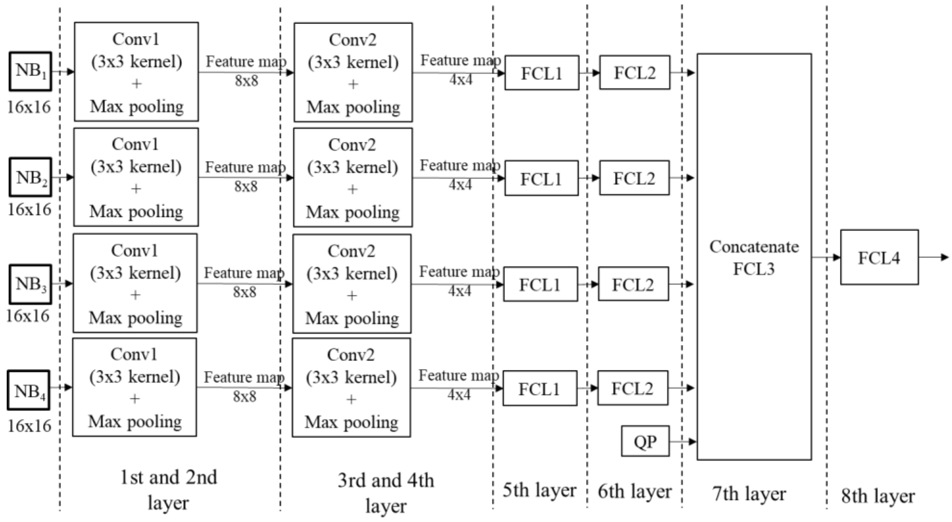

Fig. 6. Proposed CNN structure.

TABLE IV: COMPARISON WITH OTHER PAPER IN TIME SAVING, BD-BR AND BD-PSNR

Sequences	Previous work [14] TS (%)	BD-BR (%)	BD-PSNR (dB)	Proposed algorithm TS (%)	BD-BR (%)	BD-PSNR (dB)	[14] vs Proposed algorithm ΔTS (%)	ΔBD-BR (%)	ΔBD-PSNR (dB)
PeopleOnStreet	74.6	5.24	-0.26	72.8	3.23	-0.13	-1.8	-2.01	0.13
Traffic	73.4	5.01	-0.24	70.4	2.11	-0.09	-3.0	-2.90	0.15
BasketballDrive	76.1	5.52	-0.14	76.3	1.93	-0.08	0.2	-3.59	0.06
BQTerrace	72.3	4.03	-0.20	73.1	1.92	-0.08	0.8	-2.11	0.12
Cactus	77.5	4.72	-0.16	74.9	2.16	-0.18	-2.6	-2.56	-0.02
Kimono	62.6	3.64	-0.12	75.5	1.56	-0.06	12.9	-2.08	0.06
ParkScene	72.0	3.97	-0.16	74.3	1.13	-0.05	2.3	-2.84	0.11
Average	72.6	4.59	-0.18	73.9	2.01	-0.10	1.3	-2.58	0.09

The results of our experiment are summarized in Table III and Table IV. The coding performance comparisons between the proposed algorithm and the original HM16.7 are shown in Table III. The proposed algorithm shows a consistent gain in encoding time saving for all sequences with the least gain of 60.6\% in Rollercoaster and the most gain of 75.8\% in Kimono. For all sequences, the proposed algorithm can save 66.7\% encoding time and 70.1\% complexity reduction.

In Table IV, the time reduction percentage compared to previous work [14] is shown, the impact on the bit-rate (bjontegaard delta rate (BD-BR)), and the video quality in terms of PSNR (BD-PSNR in dB) [15]. Table IV shows that the proposed algorithm reduces TS by about 1.2\% better than in [14]. Additionally, BD-BR-2.58\% and BD-PSNR 0.09dB are improved by our proposed algorithm, respectively. In particularly, our approach has an effect on the complexity texture such as Traffic and BasketballDrive.

However, our approach has some drawbacks. Our proposed algorithm may impact to increase of hardware area because the CNN structure of multiple inputs requires the parallel processing. Additionally, the utilization of much kernels need to reserve many parameters. Therefore, in our CNN structure, a large memory area is required. To solve these problems, we need to consider the reduction of number of input pixel and kernel parameter in our future work.

VI. Conclusion and Future Work

The focus of this paper is on developing a complexity reduction scheme for HEVC encoder. The proposed algorithms use fast intra coding. Our scheme utilizes the CNN analysis to predict the CU sizes of the CTUs of intra coding. To realize the low complexity of CU size decision for HEVC, our approach notice the CNN structure of multiple inputs. The performance of the proposed algorithm was tested on a representative set of video sequences and was compared against the unmodified HM encoder as well as two of the art complexity reduction schemes and combinations. Performance evaluations show that our proposed algorithms reduce encoding time on average 67.3\% and increases BD-rate about 1.8\%, compared with HM 16.7. In our future, we discuss the reduction of some parameters used for CNN.

Acknowledgment

This work was supported by National Natural Science Foundation of China (NSFC, NO.61701297) and JSPS KAKENHI Grant Numbers 15K00152, 17K00157.

References

[1] B. Bross, W.-J. Han, J.-R. Ohm, G. J. Sullivan, Y.-K. Wang, and T. Wiegand, “High Efficiency Video Coding (HEVC) text specification draft 10,” Doc. JCT-VC-L1003, 2013.

[2] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, “Overview of the HIGH Efficiency Video Coding (HEVC) standards,” *IEEE Trans. Circuit Syst. Video Technol.*, vol. 22, no. 12, pp. 1649-1668, 2012.

[3] S. Ma, S. Wang, S. Wang, L. Zhao, Q. Yu, and W. Gao, “Low complexity rate distortion optimization for HEVC,” in *Proc. Data Compression Conference (DCC)*, Mar. 2013, pp. 73-82.

[4] J. Zhu, Z. Liu, D. Wang, Q. Han, and Y. Song, “Fast prediction mode decision with hadamard transform based rate-distortion cost estimation for HEVC intra coding,” in *Proc. IEEE International Conference on Image Processing(ICIP)*, Sep. 2013, pp. 1977-1981.

[5] Z. Liu, S. Guo, and D. Wang, “Binary classification based linear rate estimation model for HEVC RDO,” in *Proc. IEEE International Conference on Image Processing(ICIP)*, Sep. 2014, pp.3676-3680.

[6] H. Zhang and Z. Ma, “Fast intra mode decision for High Efficiency Video Coding (HEVC),” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 24, pp. 660-668, Nov. 2012.

[7] Y. Zhang, Z. Li, and B. Li, “Gradient-based fast decision for intra prediction in HEVC,” in *Proc. Visual Communications and Image Processing (VCIP)*, Jan. 2012, pp. 1-6.

[8] B. Min and R. C. C. Cheung, “A fast cu size decision algorithm for the HEVC intra encoder,” *IEEE Trans Circuits and Systems for Video Technology*, vol. 25, pp. 892-896, Oct. 2015.

[9] N. Hu and E.-H. Yang, “Fast mode selection for HEVC intra frame coding with entropy coding refinement based on transparent composite model,” *IEEE Trans. Circuits and Systems for Video Technology*, vol. 25, pp. 1521-1532, Jan. 2015.

[10] X. Yu, Z. Liu, J. Liu, Y. Gao, and D. Wang, “VLSI friendly fast CU/PU mode decision for HEVC intra encoding: Leveraging convolution neural network,” in *Proc. IEEE International Conference on Image Processing (ICIP)*, Sept. 2015, pp. 1285-1289.

[11] Z. Liu, X. Yu, S. Chen, D. Wang, “CNN oriented fast HEVC intra CU mode decision,” in *Proc. IEEE International Symposium on Circuits and Systems (ISCAS)*, Aug. 2016, pp. 2270-2273.

[12] K. He, X. Zhang, S. Ren, and J. Sun, “Identity mappings in deep residual networks,” arXiv preprint arXiv:1603.05027, 2016.

[13] F. Bossen, “Common HM test conditions and software reference configurations,” Doc. JCTVC-L1100, Apr. 2013.

[14] Z. Liu, X. Yu, Y. Gao, S. Chen, X. Ji, and D. Wang, “CU partition mode decision for HEVC hardwired intra encoder using convolution neural network,” *IEEE Trans. Image Processing*, vol. 25, pp. 5088-5103, Aug. 2016.

[15] S. Pateax, “An excel add-in for computing Bjontegaard metric and its evolution,” Doc. VCEG-AE07, Apr. 2007.

Takafumi Katayama received his B.E. and M.E. degrees in electrical engineering from Tokushima University. He belonged to the Renesas Electronics Corporation from 2012 to 2014. Now he is proceeding to the Ph.D degree in electrical engineering of Tokushima University from 2015. His current research interests include video coding algorithms, hardware design, and machine learning.

Tian Song received his B.E. degree from Dalian University of Technology, China, in 1995, his M.E. and Dr.E. degrees from Osaka University in 2001 and 2004, respectively. He joined Tokushima University in 2004 as an assistant professor. Presently, he is an associate professor of the Department of Electrical and Electronic Engineering, Graduate School of Advanced Technology and Science, Tokushima University. His current research interests include video coding algorithms, VLSI architectures, and system design methodology.

Wen Shi received his B.E. degree in electrical engineering and automation from Harbin Institute of Technology, China, in 2012, and his M.E. degree from Tokushima University, Japan, in 2015. Currently, he is pursuing the Ph.D. degree in electrical and electronic engineering at the Graduate School of Advanced Technology and Science, Tokushima University, Japan. His research interests include video coding algorithms and VLSI design.

Xiantao Jiang received the M.E. and Dr.E. degrees from Shanghai Maritime University and Tokushima University in 2012 and 2016. He joined Shanghai Maritime University in 2016 as an assistant professor. His interested research areas are the video coding, computer vision, machine learning and parallel processing algorithm. His interested research areas are the generation video coding

Takashi Shimamoto received his B.E. and M.E. degrees in electrical engineering from Tokushima University, and his Dr.E. degree from Osaka University in 1982, 1984, and 1992, respectively. He joined Tokushima University in 1984 as an assistant professor. Presently, he is a professor of the Department of Electrical and Electronic Engineering, Graduate School of Advanced Technology and Science, Tokushima University. His research interests include heuristic algorithms for VLSI CAD.