IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 26 July 2023, accepted 10 August 2023, date of publication 14 August 2023, date of current version 21 August 2023.

*Digital Object Identifier 10.1109/ACCESS.2023.3305266*

RESEARCH ARTICLE

# CNN-LNN Based Fast CU Partitioning Decision for VVC 3D Video Depth Map Intra Coding

**FENGQIN WANG, ZHIYING WANG, AND QIUWEN ZHANG, (Member, IEEE)**
College of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou 450002, China

Corresponding author: Qiuwen Zhang (zhangqwen@126.com)

This work was supported in part by the National Natural Science Foundation of China under Grant 61771432 and Grant 61302118; in part by the Basic Research Projects of Education Department of Henan under Grant 21zx003 and Grant 20A880004; in part by the Key Projects Natural Science Foundation of Henan under Grant 232300421150; in part by the Scientific and Technological Project of Henan Province under Grant 232102211014; and in part by the Postgraduate Education Reform and Quality Improvement Project of Henan Province under Grant YJS2021KC12, Grant YJS2023JC08, and Grant YJS2022AL034.

**ABSTRACT** Currently, the coding efficacy of the cutting-edge video coding standard H.266/VVC surpasses that of 3D-HEVC (3D-High Efficiency Video Coding), but the existing VVC (Versatile Video Coding) low-complexity coding algorithm is mainly optimized for 2D video coding and cannot fully utilize the characteristics of the depth map itself. Based on this, we propose a fast decision algorithm employing the CNN (Convolutional Neural Network)-LNN (Lightweight Neural Network) model to diminish the intricacy of depth map intra coding in VVC 3D video. The algorithm treats the CU partitioning process in depth map coding as a two-stage process, first adding a non-local block and spatial pyramid pooling to the CNN model, enabling the proposed CNN model to skip the flat regions in the depth map and perform adaptive partitioning prediction of CUs in the edge regions; then, the LNN model is used to make early decision on TT (Ternary Tree) partition for CUs that need to be partitioned, and skip decisions for CUs that do not need to be partitioned by TT, so as to reduce some unnecessary RDO calculations. Experimental results illustrate that the algorithm achieves a notable reduction in encoding time amounting to 43.23% on average, with a negligible impact on the increase of BDBR.

**INDEX TERMS** VVC 3D video, depth map coding, CU early prediction, CNN-LNN.

## I. INTRODUCTION

Amidst the incessant advancement of information and communication technology, the application of digital video has become more widespread and people's pursuit of visual effects has become higher and higher, not only for the increasing requirements of clarity but also for the experience of watching video. To meet these requirements, video coding technology has evolved rapidly [1], the resolution of video is increasing from standard definition (SD) to ultra-high definition (UHD) [2] and multi-view stereoscopic video is another direction of video development, with views evolving from 2D to 3D and free-view, with stereoscopic video, multi-view video, virtual reality and augmented reality emerging as immersive videos with large viewing angles, high picture quality and a sense of image envelopment, overturning the traditional visual experience. Compared to traditional 2D video, 3D video usually contains multiple viewpoints of video, presenting a 3D effect and therefore providing the user with visual and auditory enjoyment that 2D video cannot [3]. Since 3D video provides an immersive visual experience, it has also been successfully applied in people's daily life, such as 3D film and television, free-viewpoint TV, virtual reality and medical equipment and other fields. 3D video requires more video transmission bandwidth and more video storage space due to the need to encode multiple views at the same time, and its data volume is several times that of 2D video, so reducing the complexity of 3D video has been a hot topic of research both domestically and internationally.

2D video coding technology can no longer meet people's new requirements for visual effects. Therefore, the Motion Image Expert Group and the Video Coding Expert Group

The associate editor coordinating the review of this manuscript and approving it for publication was Zhaoqing Pan.

jointly formed a 3D Video Joint Coding Group [4] to jointly develop a new generation of 3D video in July 2012 to develop a new generation of 3D video coding standard, and officially released 3D-HEVC in February 2015, which is a major progress based on multi-view video. Among them, 3D-HEVC adopts the encoding format of multi-viewpoint texture plus depth [5], which uses the rendering technology based on depth map to realize the synthesis of any virtual viewpoint, and reduces the data volume of the video to be encoded by reducing the number of viewpoints. Currently, the latest generation of traditional video coding standards is VVC announced in July 2020 [6]. In contrast to the previous generation of standard HEVC, the overall performance of VVC has been greatly improved, and as shown in Figure 1 CU (Coding Unit) partition adopts QTMT (Quad-tree with Nested Multi-type Tree) partition structure, which consists of six partitions (no partition, quadtree partition(QT),vertical binary tree partition(BTV), horizontal binary tree partition(BTH), vertical ternary tree partition(TTV) and horizontal ternary tree partition(TTH)), in which the maximum size of CTU (Coding Tree Unit) allowed is 128 × 128, the minimum size allowed for QT sub-nodes is 16 × 16, the maximum size allowed for BT and TT is 64 × 64, and the minimum size is 4 × 4. Making it more flexible while reducing the CU prediction residuals [7]. In addition, the intra prediction direction in VVC is the same as in HEVC, both from 45° to -135° in a clockwise direction, and the number of angle prediction modes has increased from 35 to 65, making the prediction more refined [8].

Compared with 2D video, the depth map introduced in 3D video differs substantially from the texture map. The depth map has large smooth areas and prominent edges. The introduction of depth map coding techniques to accurately encode the edge regions of the depth map has engendered a notable increase in the intricacy of coding, which is one of the reasons for the overall high coding complexity of 3D video [9]. In 3D video, the depth map predominantly comprises undulating terrains interspersed with precipitous contours, and the distortion of these delineations engenders resonant artifacts at the peripheries of objects. Depth map coding is an important part of 3D video coding, and its quality of the depth map directly affects the bit rate required for coding and the video quality of synthetic viewpoints. Figure 2 shows that the size and depth of the CU partition in VVC 3D video are exists a strong interrelation between the edge features of the depth map [10]. Complex edge regions are usually delineated using deeper depth and small-size CUs; conversely, simple edge regions are usually encoded using shallow depth and large-size CUs. Furthermore, distinct edges exhibit different pixel values [11]. Owing to the flexible QTMT partition structure and the coding technology in depth map, the coding complexity and coding time of the VVC 3D video is greatly increased. Since the MTT (Multi-Type Tree) structure is a newly introduced coding scheme, it is also one of the hotspots that people pay more attention to. However, the complexity of its structure also makes researchers feel difficult. Most research methods only focus on QT and BT [12], and a few studies on TT [14]. These researchers used statistical analysis and decision tree models to study the complexity of QTMT, but it is difficult to avoid the problem of overfitting, and high complexity will be generated in the process of feature extraction. That is, although the complexity of MTT can be reduced, the problem of reducing the complexity of TT is still a problem to be solved. Hence, to more effectively resolve the aforementioned issues, it is imperative to find a fast coding algorithm that can effectively diminish the complexity of

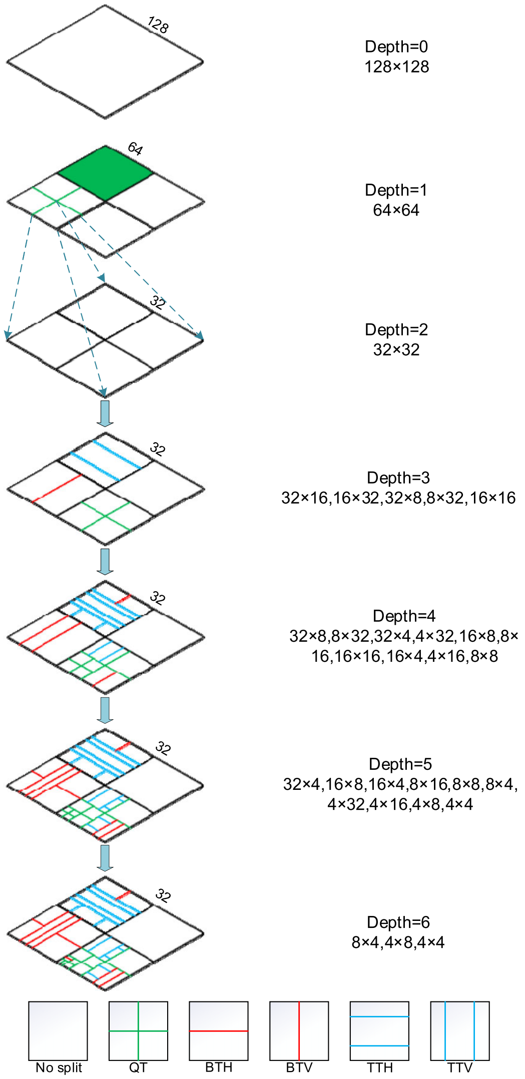


**FIGURE 1.** CU partition mode and depth in the QTMT partition structure.

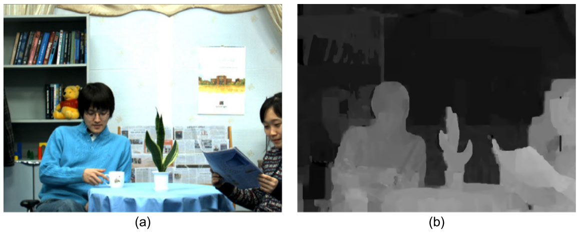


**FIGURE 2.** A frame from the video sequence "Newspaper" (a) texture map (b) depth map.

depth map coding in VVC 3D video while ensuring coding efficiency.

According to the characteristics of the depth map, to further mitigate the computational complexity of the depth map in the VVC 3D video, we mainly make the following contributions: (1) we propose a CNN-based adaptive CU partition prediction algorithm, which adds spatial pyramidal pooling to the CNN model to solve the problem of uniform input of multi-size CU into the CNN model caused by different CU sizes in the coding process; second, a non-local block is introduced, which enables the proposed CNN model to partition and predict CU in edge regions while skipping large-scale flat regions in the depth map. (2) an LNN-based early TT decision algorithm for CU is proposed, using the LNN model to make early judgments on TT partition for CU that need to be divided in the previous stage and skip decisions for CU that do not need TT partition, to accomplish the objective of reducing some unnecessary RDO (Rate Distortion Optimization) calculations, thus reducing the larger computational complexity.

The subsequent sections of this paper are structured as follows. In Section II, related work on diminishing the complexity of 3D video coding is reviewed. Section III presents the proposed fast intra coding algorithm for depth map. Section IV presents the empirical findings and comprehensive analysis of the algorithm. Finally, Section V concludes the paper.

## II. RELATED WORKS

Currently, a multitude of expeditious intra coding algorithms exist for 2D video, demonstrating remarkable aptitude in diminishing the intricacy of HEVC and VVC intra coding, but because depth map is introduced in 3D video and texture map and depth map have different coding techniques, therefore it is not suitable for intra coding of 3D video depth map [16]. To effectively diminish the intricacy entailed in coding the depth map within 3D video, researchers have proposed the following three main types of fast decision algorithms.

### A. FAST INTRA MODE DECISION ALGORITHM

The computational intricacy of mode decision in depth map coding in 3D video is so high that it is common to skip unnecessary prediction mode using either traditional methods or in combination with machine learning methods. In the reference [17], Zhang et al. introduced two highly efficacious algorithms for intra decision in depth modeling mode, one of these approaches delves into the statistical attributes of variance distribution within two partitions of depth modeling mode, it subsequently suggests a straightforward, yet potent criterion founded upon the squared Euclidean distance of variance to accurately assess the RD cost of a prospective DMM (Depth Modeling Mode) candidate; the other is to propose a probabilistic-based early depth modeling mode decision to merely the utmost plausible modes are chosen and to determine the use of SDCs (Segment wise Depth Coding) in advance grounded upon the RD cost of diminished intricacy in the deliberations of coarse mode selection. A fast coding algorithm for depth map is proposed in reference [18], which first analyses the intricacy of RD cost computation in 3D-HEVC; subsequently designs the RD cost calculation to sequentially calculate the coded bits, depth distortion and SVDC (Synthesized View Distortion Change); and finally designs an early termination method for RD cost calculation. To reduce the computational intricacy of depth map intra coding, Wang et al. proposed a multi-rate depth intra pattern decision algorithm in reference [19] that combines an early RMD (Rough Mode Decision) termination strategy, a candidate pattern reduction strategy, and a fast DMM decision strategy to skip unnecessary DMM patterns by employing a simplified edge detection method with minimal computational complexity. Reference [20] proposes a depth region segmentation-based intra prediction model that integrates segmented CNN into intra prediction for depth map coding, using DRS-Net (Deep Region Segmentation Network) to acquire partitioning results at the frame level to locate target locations more accurately. Song et al. proposed a content adaptive pattern decision to mitigate the intricacy entailed in encoding depth map in 3D-HEVC in reference [21], which adaptively skipped certain unnecessary prediction patterns.

### B. FAST CU PARTITION DECISION ALGORITHM

As texture views and depth map in 3D video have different characteristics, existing fast intra prediction methods are used to encode texture views and depth map respectively. Based on the self-learning residual model, the research in reference [22] presents a swift algorithm for intra size decision in the context of texture views and depth map within 3D-HEVC intra coding, which firstly acquires the residual signal by extracting it from both the pristine luminance pixels and the optimally predicted luminance pixels corresponding to each CU size; additionally, it employs the self-learning residual model for expeditious determination of CU intra size to predict the optimal CU size ahead of time using the residual signal features. In reference [23], Li et al. proposed an unsupervised learning-based scheme that enables adjustable early decision-making regarding CU size for intra coding of depth map within the framework of 3D-HEVC, proposing three clustering models for clustering 64 × 64, 32 × 32 and 16 × 16 CUs to determine early whether they are further divided and introducing similarity distances to achieve adjustable early CU size decisions to attain varying degrees of reduction in coding complexity. In reference [24] presented a fast depth map coding algorithm for 3D-HEVC, which incorporates the principles of data mining and machine learning to establish associations between encoder context attributes, culminating in the construction of a static decision tree, and then judging whether the present CU necessitates partitioning. In [10], a fast depth map intra coding method based on CNN is proposed to diminish the intricacy of 3D-HEVC. In the initial stage, a database of independent views based on the depth map is established, and which curated repository encompasses the CU partition data associated with the depth map.

Subsequently, a sophisticated framework known as the Deep Edge Classification Convolutional Neural Network (DEC-CNN) is established that primary objective revolves around the classification of the intricate edges found within the depth map. Lastly, the pixel values derived from the binarized depth image are utilized to rectify and refine the aforementioned classification outcomes.

### C. JOINT FAST INTRA MODE DECISION AND CU PARTITIONING ALGORITHM

Certain research methodologies employ concurrent techniques to address both the expedited intra mode decision and the predicted CU size decision, thereby augmenting the overall efficiency of depth map intra coding. A size-decision algorithm for intra prediction of depth map is proposed in [25], which creates a size-decision model to expedite intra encoding of depth map based on an automatic merging likelihood clustering method for a set of selective data. Chen et al. [26] proposed a 3D-HEVC depth map intra coding algorithm on the basis of boundary continuity. Initially, the fast intra mode decision diminishes the number of intra mode and RMD candidates founded upon the intricacy of boundaries; subsequently, the fast DMM decision method determines whether DMM prediction should be employed based on the difference in boundary variance; and finally, the fast CU early termination algorithm incorporates RD cost constraints to prevent superfluous CU splitting in the smoothed region. Reference [27] proposed a 3D-HEVC fast coding algorithm on the basis of visual perception, which extracts the visual edge and depth map of color texture respectively, classifies CTU into various types, and designs acceleration algorithms for each type. Harnessing the features of the human visual system, a fast algorithm for accelerated 3D-HEVC depth intra coding on the basis of visual perception was proposed in the reference [28], the depth map was split into different regions by automatic thresholding of Otsu, the dominant edge direction was classified for every prediction unit, and the perceptual edges were detected based on a disparity depth difference model to extract regions that could potentially influence visual perception; based on the partition of the depth map and analysis of edge distribution, the associated intra-corner patterns were reduced and the determination of whether to perform the depth modeling model was made, and fast CU decisions were proposed in combination with boundary continuity and RD cost thresholds.

The above three types of fast decision-making algorithms are mainly applied to 3D-HEVC, which are studied from the aspects of fast intra mode decision-making, fast CU segmentation decision-making and joint fast intra mode decision-making and CU size segmentation decision-making, respectively. At present, there are fewer fast decision-making algorithms for VVC 3D video coding, while VVC has become the mainstream video coding standard, and 3D video is also more popular, so the complexity reduction improvement for VVC 3D video coding has become a hot research direction. Therefore, we propose CNN-LNN based fast decision making for intra coding to reduce the complexity of intra coding in depth maps for the characteristics of depth maps in VVC 3D video.

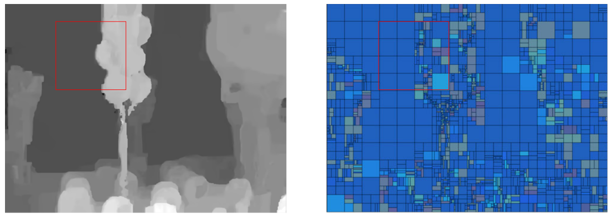
**TABLE 1.** **Official 3D video test sequence for JVT-3V.**

| Video sequences | Resolution | Frames to be encoded | Frame Rate | 3-views input |
|---|---|---|---|---|
| Undo_Dancer | 1920×1088 | 250 | 25 | 1-5-9 |
| Poznan_Hall2 | | 200 | 25 | 7-6-5 |
| Poznan_Street | | 250 | 25 | 5-4-3 |
| Shark | | 300 | 30 | 1-5-9 |
| GT-Fly | | 250 | 25 | 9-5-1 |
| Kendo | 1024×768 | 300 | 30 | 1-3-5 |
| Balloons | | 300 | 30 | 1-3-5 |
| Newspaper | | 300 | 30 | 2-4-6 |

## III. PROPOSED ALGORITHM

In contrast to 2D video, 3D video exhibits distinct characteristics in terms of its access unit structure and coding sequence, where the depth map can be thought of as a grey-scale image with pixel values representing the quantization values from the object to the camera, and the depth map uses non-uniform quantization. The different ways in which depth map describe scenes determine their particular nature and furthermore determine the coding technique. The depth map has the characteristic of segment smoothness, the pixel values of the object interior and the background area are almost unchanged and are divided by sharp edges; in addition, the depth map is employed to depict the virtual synthetic perspective; and the inter and the correlation between view-points is poor, but has a similar motion direction to the texture map. The depth map coding in VVC 3D video adopts the QTMT partition structure, and there are many types of CU partitions. Therefore, we divide the intra coding of the depth map into two stages. First, construct a CNN model to make an early judgment on whether the current CU is partitioned, and then construct an LNN model to judge whether to skip the calculation of the RD cost of the TT of the CU that needs to be partitioned.

The depth map coding in VVC 3D video adopts the QTMT partition structure, where each CU partition needs to iteratively calculate the RD cost of the QT, BT and TT partition patterns for all depths, and the partition pattern with the smallest RD cost is the best. This method has high partition accuracy but is also very time-consuming. The depth map standard test sequence selected for the establishment of the dataset is displayed in Table 1, which respectively includes the resolution, frames to be encoded, frame rate and view-point of the video sequence. These are part of the official 3D video test sequences of JVT-3V (Joint Collaborative Team on 3D Video Coding Extension Development). The chosen video sequences encompass various resolutions and scenes, ensuring that the trained model exhibits robust generalizability. To obtain the dataset for training the model, the depth map dataset was divided into multiple classes based on CU size and the data was enhanced by choosing to scale the image



(a)Depth map of a frame of the "ballons" sequence

(b)Scheme for the division of different regional CUs

**FIGURE 3.** **CU partition of a frame in the "Balloons" sequence.**

frames for the UHD sequences and to zoom in on the image frames for the HD sequences.

### A. ADAPTIVE CU PARTITION PREDICTION ALGORITHM BASED ON CNN

Depth maps are less detailed than texture maps, with sharp edges and more flat areas [29]. The QTMT partition structure tends to select larger-sized CUs for flat areas, and smaller-sized partitions for edge areas. As shown in Figure 3, this picture is the depth map of a certain frame in the sequence ‘‘balloons’’, and the red box is the depth map and partition scheme of the same area, from which we can see that the edge region has a deeper partition depth and a smaller partition size, which is a more accurate partition of the edge region, and the CU partition of the edge region often determines the coding quality of the depth map.

Neural Networks have good feature extraction ability for images as well as the ability to learn from large amounts of data, and have been used with good results in video analysis and image quality assessment [30]. Here we select CNN as the base model. In addition, the depth map coding in VVC 3D video still adopts the QTMT partition structure, that is, each CU has six possible partitions, which may lead to the input CU being cropped or distorted if different sizes are directly input into the CNN model, making distortion or information loss, thus limiting the accuracy of recognition [18]. When the size of the input CU changes, the model may not be able to adapt to the change of the current CU, resulting in the need to train multiple CNN models according to the size of the CU, which reduces the utilization of the model and wastes resources [31]. Therefore, we add a spatial pyramid pooling structure to the CNN model to solve this problem, which can pool the input feature maps by three sizes of lattices [32] to avoid the time loss caused by processing different sizes of CU.

So that the proposed CNN model can skip flat regions in the depth map and focus more on regions with sharp edges, we add a non-local block to the CNN model. The non-local block used here is based not only on the design of human brain theory but also on the non-local mean value of the non-local mean filter. That is, the basic idea is to calculate the mean of the weights between all pixels in an image, the weights being used to represent the correlation between pixels.

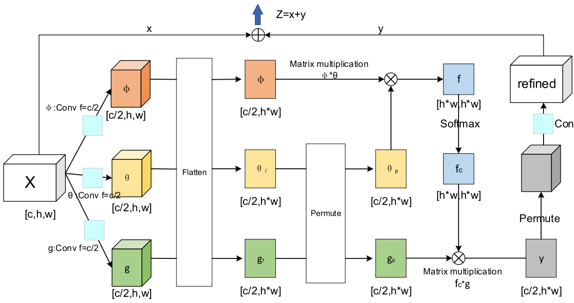


**FIGURE 4.** **Non-local block model structure.**

Therefore, we employ the non-local block in the proposed CNN model, which can be expressed as:

$$y_i = \frac{1}{C\left(x\right)} \sum_{\forall j} f\left(x_i, x_j\right) g\left(x_j\right) \tag{1}$$

where $x_i$ denotes the present pixel, $x_j$ denotes all pixels within the image, and $f\left(x_i, x_j\right)$ denotes the similarity between $x_i$ and $x_j$. $C\left(x\right)$ is the influence factor, which is used for normalization, and $g\left(x_j\right)$ is the mapping function, which is used to calculate the representation of the feature map at position $j$. The following equations are chosen here to represent the $f\left(x_i, x_j\right)$ function, the $C\left(x\right)$ function and the $g\left(x_j\right)$ function respectively:

$$f\left(x_i, x_j\right) = e^{\theta(x_i)^T \phi(x_j)} \tag{2}$$

$$C\left(x\right) = \sum_{\forall j} f\left(x_i, x_j\right) \tag{3}$$

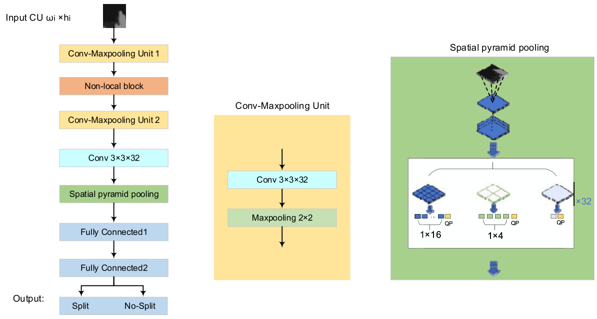
$$g\left(x_j\right) = W_g x_j \tag{4}$$

where $W_g$ is the matrix of weights to be learned. Thus $y_i$ can be formulated as follows:

$$\begin{aligned} y_i &= \frac{1}{C\left(x\right)} \sum_{\forall j} f\left(x_i, x_j\right) g\left(x_j\right) \\ &= \frac{1}{\sum_{\forall j} e^{\theta(x_i)^T \phi(x_j)}} e^{\theta(x_i)^T \phi(x_j)} W_g x_j \end{aligned} \tag{5}$$

The structural diagram of the non-local block is shown in Figure 4, where X is a feature map of size $[c, h, w]$, $c$ signifies the quantity of channels, $h$ and $w$ signify the height and width of the input feature map. First X will pass through three convolution layers of $1 \times 1$ convolution kernel to obtain $\theta$, $\phi$ and $g$, which correspond to the three functions of $\theta(x_i)$, $\phi(x_j)$ and $g(x_j)$ in the above formula respectively. Stretching $\theta$, $\phi$ and $g$ as one-dimensional vectors give $\theta_f$, $\phi_f$ and $g_f$, and transposing $\theta_f$ and $g_f$ gives $\theta_p$ and $g_p$, which are matrix multiplied to give a matrix of size $[\mathrm{h} \times \mathrm{w}, \mathrm{h} \times \mathrm{w}]$. After normalizing this matrix by the Softmax function, it is matrix multiplied with $g_p$ and its dimensions are re-stretched to $\left[c/2, h, w\right]$. Then it is convolved to expand the channel to the original $c$ dimension. Finally, X is added to the acquired feature map y to obtain the output feature map Z.

We therefore propose a CNN model structure based on a Spatial pyramid pooling structure, and non-local block layers, including one input layer, two Conv-Maxpooling layers, one



(a)Overall proposed adaptive CNN structure

(b)Details about Conv-Maxpooling Unit and Spatial pyramid pooling

**FIGURE 5.** Proposed CNN structure.

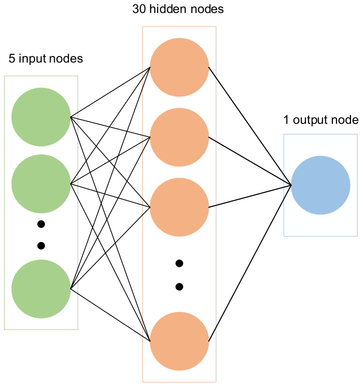


**FIGURE 6.** LNN structure diagram.

non-local block layer, one convolutional layer, one Spatial pyramid pooling layer, two fully connected layers, and one output layer. Specifically, as shown in Figure 5, a CU of size $h_i \times w_i$, where $h_i$ is the height of the CU and $w_i$ is the width of the CU, is input and the Conv-Maxpooling layer contains a convolution kernel of size 3 × 3, a convolution layer and a pooling layer with a pooling kernel of size 2 × 2, using the maximum pooling method; the non-local block layer performs non-local block operations on the feature map output from the upper layer using the mentioned non-local model; the Spatial pyramid pooling layer incorporates QP (Quantization Parameter) to make the model more focused on the edge CU; the fully connected layer uses Relu (Rectified Linear Unit) function as the activation function and will also be lost with 50% probability as it goes along in order to prevent overfitting; the output layer uses the sigmoid function as the activation function to obtain the output values.

According to the QP, the depth map dataset is divided into four groups: 34, 39, 42, and 45, and a multi-scale training method is adopted for the proposed CNN model, i.e. in each epoch, the first size model was trained to generate then load this model and train the second size until all sizes were trained. The Adam optimizer is selected, the initial learning rate is set to $10^{-3}$, the minimum learning rate is $10^{-6}$, and a total of 1000 epochs are iterated. The cross-entropy function can prevent the model from falling into a local optimum during the learning process and is widely used in classification problems, so the loss function uses the cross-entropy function:

$$loss = -\frac{1}{n}\sum_{i=1}^{n}\left[y_i \log \hat{y} + (1 - y_i)\log\left(1 - \hat{y}\right)\right] \tag{6}$$

where $\hat{y}$ is the predicted value of the current training sample, and $y_i$ is the label of the current training sample.

### B. CU EARLY TT DECISION ALGORITHM BASED ON LNN

The depth map encoding in VVC 3D video adopts the QTMT partition structure, and there are five types of CU partitions. If the original VTM algorithm is used to perform a violent RDO search on it, the complexity will be too high. The calculation formula of the RD cost is:

$$J_m = D + \lambda \cdot B_m \tag{7}$$

where $J_m$ denotes the RD cost function, $B_m$ represents the encoding bit rate, and $\lambda$ stands for the Lagrangian multiplier. $D$ represents the distortion of the synthesized view and depth map in depth map encoding.

Hence, we propose to use the LNN model for early TT decision prediction for the CUs that need to be divided in the preceding partitioning process. In terms of the CU division structure, the complexity of the MTT structure accounts for the main aspect, containing a total of four partitions: BTV partition, BTH partition, TTV partition, and TTH partition. Therefore, the LNN model we proposed mainly makes judgments on the TT partition, establishes an LNN model for each partition direction of TT partition, and reduces unnecessary RDO searches to accomplish the purpose of reducing complexity. The purpose of using the LNN model is to effectively use a limited number of features to make judgments about the type of CU partitioning and curtail the intricacy. LNN models are mostly low number of parameters, high computational speed, and small memory footprint and lower complexity, which can achieve high accuracy of neural network models even with a small number of feature parameters [33]. That is, the fewer the set of features employed by the LNN model, the less complex the architecture will be [34], so here we must improve the precision of the LNN model by selecting the most critical features relevant to CU division in VVC 3D video, and here we choose two types of features to form the feature set.

One category is features that can be directly obtained during the coding process. Firstly, the size and shape of CU blocks are considered, and these features all have some correlation with the type of CU partition, while the size and shape are usually represented by the depth of the CU, and the proposed LNN model mainly targets TT partition, so the MTT depth (MTD) is chosen here as a feature that can be directly obtained during the CU coding process.

Another category includes features that enable finer tuning of the TT partition in addition to the CU features obtained directly during the encoding process. The tendency of the CU to split in different directions is influenced by the aspect ratio, so the block ratio BSR is chosen as a feature measure the shape of the CU, where the block ratio BSR is represented as follows:

$$BSR = \begin{cases} \dfrac{h}{w+h} & BTH \ or \ TTH \\ \dfrac{w}{w+h} & BTV \ or \ TTV \\ 0.5 & QT \end{cases} \tag{8}$$

where $w$ signifies the width of the CU, and $h$ signifies its height.

In addition, the partition of the CU is closely related to the texture direction of the CU. A CU with a horizontal texture direction should choose BTH partition or TTH partition in the MTT partition and a CU with a vertical texture direction should choose BTV partition or TTV partition, at the same time, the corresponding RD cost should be smaller than the RD cost in the other direction. Therefore, before judging whether to skip TT partition, first use the RD cost of BTH and BTV to judge the direction of TT partition. The BT advantage (BTA) based on RD cost is then chosen as one of the features, with BTA being 0.5 when the RD cost of BTV or BTH is lower than in the other partitions, and 1 when the RD cost of both BTV and BTH is lower than in the other partitions.

Additionally, by employing the Sobel operator to compute the texture distribution across various directions within the CU, where A is the pixel matrix, then the texture in the vertical and horizontal orientations of the CU can be expressed respectively as

$$Texture_V = \frac{\sum_{i=1}^{w}\sum_{j=1}^{h}\left| A * \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \right|}{w \cdot h} \tag{9}$$

$$Texture_H = \frac{\sum_{i=1}^{w}\sum_{j=1}^{h}\left| A * \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \right|}{w \cdot h} \tag{10}$$

The above are the selected features and assuming that the proposed method is exclusively employed for CUs that require further MTT partitioning, the proposed LNN model consists of 5 input nodes, 30 hidden nodes, and 1 output node. The model structure is shown in Figure 6, with the computation between the input, hidden and output layers defined as:

$$y_j = f\left(\sum_i w_{ij}x_i + b_j\right) \tag{11}$$

where $x_i$ signifies the value of the i-th input, $w_{ij}$ signifies the weight from the i-th current node to the j-th next layer node, $b_j$ represents the bias, and $j= 1, 2, \ldots, \varphi$ denotes the neuronal count within each layer. To augment the LNN's non-linear fitting capacity, a Sigmoid function is incorporated as the activation function following $y_i$.

First, there is a 5D input feature vector in the input layer, $x = (MTD, BSR, BTA, Texture_V, Texture_H)^T$, and feed $x$ into the neuron of the $j$-th node in the second hidden layer with a non-linear weighted sum. Then transfer the $y$ value obtained in the second layer to the third output layer in the same way, and finally output the $y$ value, and judge whether to skip TTH partition or TTV partition according to the output $y$ value. In the output layer, the Sigmoid function is still used as the activation function.

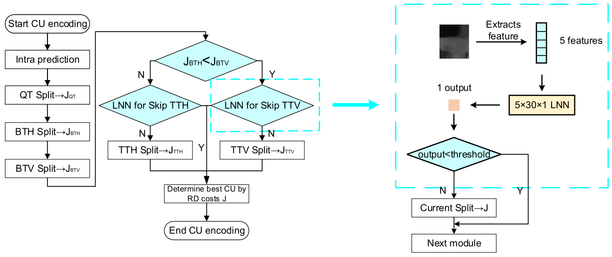


FIGURE 7. Overview of the proposed LNN model. The blue box is the proposed early TT decision algorithm.

Our proposed LNN model therefore uses a total of 2 LNN models, as shown in Figure 7, which have the same structure and are judged in advance at the TT partition decision stage, deciding whether the RD cost calculation for TTH partition or TTV partition needs to be skipped predicated on the output value of each LNN model compared to a preset threshold. The 2 LNN models operate independently and are therefore also trained separately.

Using the same depth map dataset as the CNN model proposed earlier, the depth map dataset is divided into four groups according to QP: 34, 39, 42, and 45. Using the Adam optimization method, set the learning rate to 0.01, the calculated gradient is applied to the $w_{ij}$ and $b_j$ of the output layer, and the chain rule is also used to apply it to the $w_{ij}$ and $b_j$ of the hidden layer, and adopt MSE (Mean Squared Error) as the loss function of the LNN model. During the training process, the most suitable model (with the best weights and deviations) can be updated iteratively and stored to obtain the best model, which can then be implemented in the VTM 10.0 software.

### C. OVERALL ALGORITHM

Based on the above work, we propose that the algorithm consists of two stages of decision making, namely a CNN-based adaptive CU partition prediction algorithm and an LNN-based early TT decision algorithm, which are applied to fast CU partition for VVC 3D video depth map coding as a way to reduce computational complexity. First of all, by adding a non-local block and spatial pyramid pooling structure to the proposed CNN model, the CNN model thus constructed can skip flat regions in the depth map and focus more on regions with sharp edges, and is more flexible in processing CU of different sizes, which can effectively avoid unnecessary CU judgments and time loss caused by processing CU of different sizes. In addition, the basic idea of the LNN-based early TT decision algorithm makes the TT early skip decision for the CUs that need to be divided in the previous stage, and based on the extracted CU features to judge whether they need to skip TTH partition or TTV partition, the unnecessary partition of CUs can be reduced again. Figure 8 illustrates the overall framework diagram of the proposed expedited CU partition decision for depth map encoding in VVC 3D video.

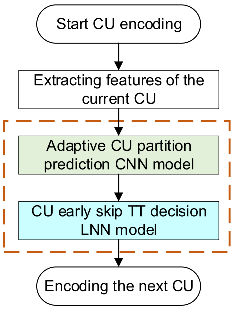

Start CU encoding
Extracting features of the current CU
Adaptive CU partition prediction CNN model
CU early skip TT decision LNN model
Encoding the next CU

**FIGURE 8.** Flow for the algorithm proposed.

**TABLE 2.** Experimental configuration.

| Hardware | |
|---|---|
| CPU | Intel(R) Core (TM) i7-11800H |
| RAM | 16 GB |
| OS | Microsoft Windows 10 64bits |
| **Software** | |
| Reference software | VTM 10.0 |
| Configuration | All intra |
| QP(depth) | 34 , 39 , 42 , 45 |

## IV. EXPERIMENTAL RESULTS

To assess the efficacy of the proposed fast CU partition decision algorithm for VVC 3D video depth map, the devised scheme was integrated into the reference software VTM10.0 and then tested using the official 3D standard video test sequences of JVT-3V with a total of 8 video sequences, including two resolutions of 1024 × 768 and 1920 × 1088, details of the video sequences are described in Table 1 in Section III-A. Table 2 lists the configuration of the algorithm for conducting the experiments.

The algorithm posited within this paper only makes improvements for depth map coding, so we measure the savings in depth map coding time of the proposed algorithm using TS, where the depth map coding time savings TS is defined as:

$$\mathrm{TS} = \frac{T_{pro} - T_{ori}}{T_{ori}} \times 100\% \tag{12}$$

where $T_{pro}$ symbolizes the encoding time of the depth map through the algorithm proposed in this paper and $T_{ori}$ denotes the depth map coding time of the reference model VTM10.0. In addition, BDBR (Bjøntegaard delta bit rate) is used as a criterion to assess the coding efficiency gains achieved by various methods while maintaining a consistent target quality.

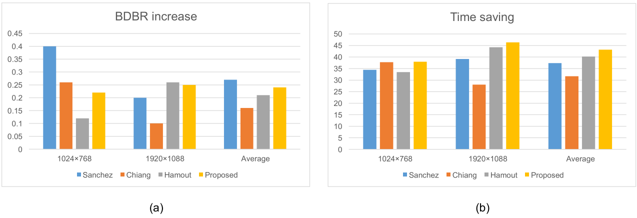
### A. ANALYSIS OF EXPERIMENTAL RESULTS

The proposed overall scheme includes a CNN-based adaptive CU partition prediction algorithm and an LNN-based early TT decision algorithm, where the CNN-based adaptive CU partition prediction algorithm can avoid RDO calculations for flat regions in the depth map and make early partition judgments for CU in edge regions, while the LNN-based early TT decision algorithm can make early TT partition judgments for CU that need to be divided and skip decisions for CU that do not need TT partition, and reducing some unnecessary RDO calculations. Table 3 displays the coding performance results attained by the proposed individual algorithm, the overall algorithm and the VTM10.0 anchoring algorithm. Evidently, it can be discerned that in the CNN-based adaptive CU partition prediction algorithm, the average time saving is 35.77% and the BDBR only increases by 0.22%, indicating that the algorithm can efficaciously skip the flat regions in the depth map and extract the CU partition of terminated flat regions, making the CU partition more focused on the edge regions, and there is no need to train multiple CNN models based on the CU size, and the spatial pyramid pooling structure in the proposed CNN model can make the CNN model adaptive to the CU size. In addition, the experimental results delineated in Table 3 evince an exemplary mean reduction of 27.85% in coding time and accompanied by a mere increment of 0.27% in BDBR in the LNN-based early TT decision algorithm, indicating that the algorithm can effectively skip unnecessary TT partition and reduce RDO calculations.

Simultaneously, Table 3 shows the coding performance achieved by the proposed overall scheme, which combines the CNN-based adaptive CU partition prediction algorithm and the LNN-based early TT decision algorithm, and the proposed algorithm can diminish the coding time by 45.37% and increase the BDBR by only 0.24% (negligible) compared to the anchoring algorithm. The Poznan_Hall2 sequence contains more flat regions and therefore has the highest coding time saving of 52.09%; the Shark sequence has the lowest coding time saving of 36.74%. Thus, the algorithm put forth can significantly increase the coding time saving while maintaining the coding quality, indicating that the present algorithm can significantly diminish the intricacy of encoding.

**TABLE 3.** The overall results of the proposed method and the results of individual method.

| Sequence | CNN | | LNN | | Overall | |
|---|---|---|---|---|---|---|
| | BDBR (%) | TS (%) | BDBR (%) | TS (%) | BDBR (%) | TS (%) |
| Balloons | 0.16 | 22.42 | 0.28 | 26.33 | 0.14 | 37.28 |
| Kendo | 0.14 | 31.05 | 0.19 | 24.21 | 0.16 | 34.52 |
| Newspaper | 0.28 | 25.81 | 0.36 | 23.79 | 0.37 | 42.16 |
| GT_Fly | 0.21 | 42.89 | 0.31 | 30.12 | 0.24 | 51.09 |
| Poznan_Hall2 | 0.37 | 50.74 | 0.29 | 34.58 | 0.35 | 48.27 |
| Poznan_street | 0.18 | 39.27 | 0.17 | 24.94 | 0.23 | 47.36 |
| Undo_dancer | 0.29 | 37.56 | 0.4 | 27.36 | 0.31 | 45.42 |
| Shark | 0.12 | 36.43 | 0.14 | 31.47 | 0.11 | 39.71 |
| 1024×768 | 0.19 | 26.43 | 0.28 | 24.78 | 0.22 | 37.99 |
| 1920×1088 | 0.23 | 41.38 | 0.26 | 29.69 | 0.25 | 46.37 |
| Average | 0.22 | 35.77 | 0.27 | 27.85 | 0.24 | 43.23 |



(a)

(b)

**FIGURE 9.** The average coding proficiency exhibited by the proposed algorithm with other algorithms in different video sequences categories (a) BDBR increase (b) Time saving.

**TABLE 4.** Comparison of the experimental results of the proposed algorithm with other algorithms.

| Sequence | Sanchez[35] | | Chiang[36] | | Hamout[25] | | Proposed | |
|---|---|---|---|---|---|---|---|---|
| | BDBR (%) | TS (%) | BDBR (%) | TS (%) | BDBR (%) | TS (%) | BDBR (%) | TS (%) |
| Balloons | 0.36 | 34.1 | 0.03 | 34.4 | 0.12 | 32.9 | 0.14 | 37.28 |
| Kendo | 0.37 | 33.9 | 0.67 | 46.7 | 0.17 | 35.2 | 0.16 | 34.52 |
| Newspaper | 0.46 | 35.4 | 0.07 | 32.1 | 0.08 | 32.3 | 0.37 | 42.16 |
| GT_Fly | 0.12 | 40.6 | 0.12 | 29.5 | 0.08 | 35.0 | 0.24 | 51.09 |
| Poznan_Hall2 | 0.43 | 38.8 | 0.07 | 38.0 | 0.39 | 51.6 | 0.35 | 48.27 |
| Poznan_street | 0.22 | 41.7 | 0.05 | 23.6 | 0.26 | 41.6 | 0.23 | 47.36 |
| Undo_dancer | 0.12 | 38.5 | 0.03 | 26.3 | 0.29 | 49.3 | 0.31 | 45.42 |
| Shark | 0.11 | 36.2 | 0.21 | 22.8 | 0.26 | 44.0 | 0.11 | 39.71 |
| Average | 0.27 | 37.4 | 0.16 | 31.7 | 0.21 | 40.2 | 0.24 | 43.23 |

### B. COMPARISON WITH OTHER ALGORITHMS

The proposed algorithm is compared with the experimental findings reported by Sanchez [35] , Chiang [36] , and Hamout [25] , where the algorithm in Sanchez [35] is related to the reduction of the RDO process, and the algorithms in both Chiang [36] and Hamout [25] are related to the CU partitioning decision. As shown in Table 4 and Figure 9, where Figure 9 better shows the coding time saving and BDBR increase of these algorithms, with the proposed algorithm mainly targeting the main improvement in reducing the complexity in depth map coding. The algorithm exhibits superior performance in video sequences with a resolution of 1920 × 1088, which can save encoding time by 46.37%, and BDBR only increases by 0.25%.

Compared with the algorithm proposed by Sanchez [35] to mitigate the computational intricacy of depth map intra prediction, the proposed algorithm accomplishes an average increase in encoding time savings of 7.97% and a reduction in BDBR of 0.03% on average. Notably, the coding time saving of the GT_Fly sequence is increased by 11.49% on average. The algorithm proposed by Chiang [36] consists of two parts, fast pattern decision and fast CU size decision, in which the coding time saving of the proposed algorithm in this paper is remarkably improved compared to the algorithm for the fast decision of CU size in depth map, with an average improvement of 13.67%, while the BDBR only increases by 0.08%, which is negligible, especially in the 1920 × 1088 video sequence, the saving of encoding time is the most improved, which is 20.13%. Compared with the CU size decision algorithm proposed by Hamout [25] aimed at diminishing the intricacy of depth map intra coding, the proposed algorithm saves an average of 5.17% coding time, with a trifling increase in BDBR. From Table 4 and Figure 9, we can evidently compare with Sanchez [35] , Chiang [36] , and Hamout [25] , the proposed algorithm exhibits superior coding performance and efficiently reduces the complexity of depth map coding, thereby demonstrating its superiority over existing methods.

## V. CONCLUSION

To effectively mitigate complexity, this paper proposes a fast decision algorithm for depth map coding in VVC 3D video, which consists of two schemes, namely CNN-based adaptive CU partition prediction and LNN-based early TT decision algorithm. The algorithm uses a CNN model to skip flat regions in the depth map, to make partition predictions for CU in edge regions, and an LNN model to make early judgments on CUs that need to be divided for TT partition, and to make skip decisions for CUs that do not need TT partition, reducing some of the unnecessary RDO calculations. The experimental results demonstrate a substantial reduction in coding complexity achieved by the current algorithm, reducing the coding time by 43.23% on average, while the increase in BDBR is negligible, and also shows excellent coding performance compared to other existing algorithms used for depth map coding. Therefore, while ensuring the coding quality, this algorithm also greatly reduces the intricacy of depth map coding in VVC 3D video.

## REFERENCES

[1] J. M. Boyce, R. Doré, A. Dziembowski, J. Fleureau, J. Jung, B. Kroon, B. Salahieh, V. K. M. Vadakital, and L. Yu, "MPEG immersive video coding standard," *Proc. IEEE*, vol. 109, no. 9, pp. 1521–1536, Sep. 2021, doi: 10.1109/JPROC.2021.3062590.

[2] M. Cheon and J.-S. Lee, "Subjective and objective quality assessment of compressed 4K UHD videos for immersive experience," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 7, pp. 1467–1480, Jul. 2018, doi: 10.1109/TCSVT.2017.2683504.

[3] M. Wien, J. M. Boyce, T. Stockhammer, and W.-H. Peng, "Standardization status of immersive video coding," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 9, no. 1, pp. 5–17, Mar. 2019, doi: 10.1109/JETCAS.2019.2898948.

[4] G. Tech, Y. Chen, K. Müller, J.-R. Ohm, A. Vetro, and Y.-K. Wang, "Overview of the multiview and 3D extensions of high efficiency video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 1, pp. 35–49, Jan. 2016, doi: 10.1109/TCSVT.2015.2477935.

[5] J. Lei, Y. Shi, Z. Pan, D. Liu, D. Jin, Y. Chen, and N. Ling, "Deep multidomain prediction for 3D video coding," *IEEE Trans. Broadcast.*, vol. 67, no. 4, pp. 813–823, Dec. 2021, doi: 10.1109/TBC.2021.3090261.

[6] M. Saldanha, G. Sanchez, C. Marcon, and L. Agostini, "Complexity analysis of VVC intra coding," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Abu Dhabi, United Arab Emirates, Oct. 2020, pp. 3119–3123, doi: 10.1109/ICIP40778.2020.9190970.

[7] O. Akbulut and M. Z. Konyar, "Improved intra-subpartition coding mode for versatile video coding," *Signal, Image Video Process.*, vol. 16, no. 5, pp. 1363–1368, Jul. 2022, doi: 10.1007/s11760-021-02088-w.

[8] Y.-W. Huang, C.-W. Hsu, C.-Y. Chen, T.-D. Chuang, S.-T. Hsiang, C.-C. Chen, M.-S. Chiang, C.-Y. Lai, C.-M. Tsai, Y.-C. Su, Z.-Y. Lin, Y.-L. Hsiao, O. Chubach, Y.-C. Lin, and S.-M. Lei, "A VVC proposal with quaternary tree plus binary-ternary tree coding block structure and advanced coding techniques," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 5, pp. 1311–1325, May 2020, doi: 10.1109/TCSVT.2019.2945048.

[9] G. Sanchez, J. Silveira, L. V. Agostini, and C. Marcon, ''Performance analysis of depth intra-coding in 3D-HEVC,'' *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 8, pp. 2509–2520, Aug. 2019, doi: 10.1109/TCSVT.2018.2865645.

[10] C. Liu, K. Jia, and P. Liu, ''Fast depth intra coding based on depth edge classification network in 3D-HEVC,'' *IEEE Trans. Broadcast.*, vol. 68, no. 1, pp. 97–109, Mar. 2022, doi: 10.1109/TBC.2021.3106143.

[11] S. Bakkouri and A. Elyousfi, ''Effective CU size decision algorithm based on depth map homogeneity for 3D-HEVC inter-coding,'' in *Proc. Int. Conf. Intell. Syst. Comput. Vis. (ISCV)*, Fez, Morocco, Jun. 2020, pp. 1–6, doi: 10.1109/ISCV49265.2020.9204037.

[12] H. Yang, L. Shen, X. Dong, Q. Ding, P. An, and G. Jiang, ''Low-complexity CTU partition structure decision and fast intra mode decision for versatile video coding,'' *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 6, pp. 1668–1682, Jun. 2020, doi: 10.1109/TCSVT.2019.2904198.

[13] M. Amna, W. Imen, S. F. Ezahra, and A. Mohamed, ''Fast intra-coding unit partition decision in H.266/FVC based on deep learning,'' *J. Real-Time Image Process.*, vol. 17, no. 6, pp. 1971–1981, Dec. 2020, doi: 10.1007/s11554-020-00998-5.

[14] S.-H. Park and J.-W. Kang, ''Context-based ternary tree decision method in versatile video coding for fast intra coding,'' *IEEE Access*, vol. 7, pp. 172597–172605, 2019, doi: 10.1109/ACCESS.2019.2956196.

[15] Z. Jin, P. An, C. Yang, and L. Shen, ''Fast QTBT partition algorithm for intra frame coding through convolutional neural network,'' *IEEE Access*, vol. 6, pp. 54660–54673, 2018, doi: 10.1109/ACCESS.2018.2872492.

[16] Q. Zhang, Y. Wang, L. Huang, B. Jiang, and R. Su, ''Adaptive CU split prediction and fast mode decision for 3D-HEVC texture coding based on just noticeable difference model,'' *Digit. Signal Process.*, vol. 106, Nov. 2020, Art. no. 102851, doi: 10.1016/j.dsp.2020.102851.

[17] H.-B. Zhang, C.-H. Fu, Y.-L. Chan, S.-H. Tsang, and W.-C. Siu, ''Probability-based depth intra-mode skipping strategy and novel VSO metric for DMM decision in 3D-HEVC,'' *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 2, pp. 513–527, Feb. 2018, doi: 10.1109/TCSVT.2016.2612693.

[18] J. Huo, X. Zhou, H. Yuan, S. Wan, and F. Yang, ''Fast rate-distortion optimization for depth maps in 3-D video coding,'' *IEEE Trans. Broadcast.*, vol. 69, no. 1, pp. 21–32, Mar. 2023, doi: 10.1109/TBC.2022.3192992.

[19] C. Wang, G. Feng, C. Cai, X. Han, and H. Cao, ''Multi-strategy depth intra mode decision algorithm in 3D-HEVC,'' *Multimedia Tools Appl.*, vol. 79, nos. 13–14, pp. 8841–8861, Apr. 2020, doi: 10.1007/s11042-019-7715-0.

[20] J. Zhang, Y. Hou, Z. Zhang, D. Jin, P. Zhang, and G. Li, ''Deep region segmentation-based intra prediction for depth video coding,'' *Multimedia Tools Appl.*, vol. 81, no. 25, pp. 35953–35964, Oct. 2022, doi: 10.1007/s11042-022-13344-7.

[21] W. Song, P. Dai, and Q. Zhang, ''Content-adaptive mode decision for low complexity 3D-HEVC,'' *Multimedia Tools Appl.*, vol. 82, no. 17, pp. 26435–26450, Mar. 2023, doi: 10.1007/s11042-023-14874-4.

[22] Y. Li, N. Zhu, G. Yang, Y. Zhu, and X. Ding, ''Self-learning residual model for fast intra CU size decision in 3D-HEVC,'' *Signal Process., Image Commun.*, vol. 80, Feb. 2020, Art. no. 115660, doi: 10.1016/j.image.2019.115660.

[23] Y. Li, G. Yang, A. Qu, and Y. Zhu, ''Tunable early CU size decision for depth map intra coding in 3D-HEVC using unsupervised learning,'' *Digit. Signal Process.*, vol. 123, Apr. 2022, Art. no. 103448, doi: 10.1016/j.dsp.2022.103448.

[24] M. Saldanha, G. Sanchez, C. Marcon, and L. Agostini, ''Fast 3D-HEVC depth map encoding using machine learning,'' *IEEE Trans. Circuits Syst. Video Technol.*, vol. 30, no. 3, pp. 850–861, Mar. 2020, doi: 10.1109/TCSVT.2019.2898122.

[25] H. Hamout and A. Elyousfi, ''A computation complexity reduction of the size decision algorithm in 3D-HEVC depth map intracoding,'' *Adv. Multimedia*, vol. 2022, Jun. 2022, Art. no. 3507201, doi: 10.1155/2022/3507201.

[26] M.-J. Chen, J.-R. Lin, Y.-C. Hsu, Y.-S. Ciou, C.-H. Yeh, M.-H. Lin, L.-J. Kau, and C.-Y. Chang, ''Fast 3D-HEVC depth intra coding based on boundary continuity,'' *IEEE Access*, vol. 9, pp. 79588–79599, 2021, doi: 10.1109/ACCESS.2021.3083498.

[27] J. Lin, M. Chen, C. Yeh, S. D. Lin, K. Sue, L. Kau, and Y. Ciou, ''Vision-oriented algorithm for fast decision in 3D video coding,'' *IET Image Process.*, vol. 16, no. 8, pp. 2263–2281, Jun. 2022, doi: 10.1049/ipr2.12488.

[28] J.-R. Lin, M.-J. Chen, C.-H. Yeh, Y.-C. Chen, L.-J. Kau, C.-Y. Chang, and M.-H. Lin, ''Visual perception based algorithm for fast depth intra coding of 3D-HEVC,'' *IEEE Trans. Multimedia*, vol. 24, pp. 1707–1720, 2022, doi: 10.1109/TMM.2021.3070106.

[29] S. Xie and Z. Tu, ''Holistically-nested edge detection,'' *Int. J. Comput. Vis.*, vol. 125, nos. 1–3, pp. 3–18, Dec. 2017, doi: 10.1007/s11263-017-1004-z.

[30] Z. Pan, H. Zhang, J. Lei, Y. Fang, X. Shao, N. Ling, and S. Kwong, ''DACNN: Blind image quality assessment via a distortion-aware convolutional neural network,'' *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 11, pp. 7518–7531, Nov. 2022, doi: 10.1109/TCSVT.2022.3188991.

[31] H. Zhang, W. Yao, H. Huang, Y. Wu, and G. Dai, ''Adaptive coding unit size convolutional neural network for fast 3D-HEVC depth map intracoding,'' *J. Electron. Imag.*, vol. 30, no. 4, Jun. 2021, Art. no. 041405, doi: 10.1117/1.JEI.30.4.041405.

[32] K. He, X. Zhang, S. Ren, and J. Sun, ''Spatial pyramid pooling in deep convolutional networks for visual recognition,'' *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015, doi: 10.1109/TPAMI.2015.2389824.

[33] Z. Pan, F. Yuan, W. Yu, J. Lei, N. Ling, and S. Kwong, ''RDEN: Residual distillation enhanced network-guided lightweight synthesized view quality enhancement for 3D-HEVC,'' *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 9, pp. 6347–6359, Sep. 2022, doi: 10.1109/TCSVT.2022.3161103.

[34] S.-H. Park and J.-W. Kang, ''Fast multi-type tree partitioning for versatile video coding using a lightweight neural network,'' *IEEE Trans. Multimedia*, vol. 23, pp. 4388–4399, 2021, doi: 10.1109/TMM.2020.3042062.

[35] G. Sanchez, L. Agostini, and C. Marcon, ''A reduced computational effort mode-level scheme for 3D-HEVC depth maps intra-frame prediction,'' *J. Vis. Commun. Image Represent.*, vol. 54, pp. 193–203, Jul. 2018, doi: 10.1016/j.jvcir.2018.05.003.

[36] J.-C. Chiang, K.-K. Peng, C.-C. Wu, C.-Y. Deng, and W.-N. Lie, ''Fast intra mode decision and fast CU size decision for depth video coding in 3D-HEVC,'' *Signal Process., Image Commun.*, vol. 71, pp. 13–23, Feb. 2019, doi: 10.1016/j.image.2018.10.009.

**FENGQIN WANG** received the Ph.D. degree in signal and information processing from Northwestern Polytechnical University, Xi'an, China, in 2010. Since 2005, she has been a Faculty Member of the College of Computer and Communication Engineering, Zhengzhou University of Light Industry, where she is currently an Associate Professor. She has published over 20 technical papers in the field of image processing and coding. Her major research interests include video information processing, machine learning, and pattern recognition.

**ZHIYING WANG** received the B.S. degree in software engineering from the Zhongyuan University of Technology, Zhengzhou, China, in 2020. She is currently pursuing the master's degree in computer technology with the School of Computer and Communication Engineering, Zhengzhou University of Light Industry, Zhengzhou. Her current research interests include image processing, deep learning, versatile video coding, and extensions of the versatile video coding.

**QIUWEN ZHANG** (Member, IEEE) received the Ph.D. degree in communication and information systems from Shanghai University, Shanghai, China, in 2012. Since 2012, he has been a Faculty Member of the College of Computer and Communication Engineering, Zhengzhou University of Light Industry, where he is currently a Professor. He has published over 30 technical papers in the field of pattern recognition and image processing. His major research interests include 3D signal processing, machine learning, pattern recognition, video codec optimization, and multimedia communication.