*IEIE Transactions on Smart Processing and Computing, vol. 3, no. 3, June 2014*
http://dx.doi.org/10.5573/IEIESPC.2014.3.3.118


*IEIE Transactions on Smart Processing and Computing*

# Novel Motion and Disparity Prediction for Multi-view Video Coding

**Woong Lim, Junghak Nam, and Donggyu Sim**

Department of Computer Engineering, Kwangwoon University / Seoul, South Korea
{limwoong, qejixfyza, dgsim}@kw.ac.kr

***** Corresponding Author: Donggyu Sim

*Received November 22, 2013; Revised January 5, 2014, Accepted March 20, 2014; Published June 30, 2014*

** Regular Paper*

***Abstract*:** This paper presents an efficient motion and disparity prediction method for multi-view video coding based on the high efficient video coding (HEVC) standard. The proposed method exploits inter-view candidates for effective prediction of the motion or disparity vector to be coded. The inter-view candidates include not only the motion vectors of adjacent views, but also global disparities across views. The motion vectors coded earlier in an adjacent view were found to be helpful in predicting the current motion vector to reduce the number of bits used in the motion vector information. In addition, the proposed disparity prediction using the global disparity method was found to be effective for interview predictions. A multi-view version based on HEVC was used to evaluate the proposed algorithm, and the proposed correspondence prediction method was implemented on a multi-view platform based on HEVC. The proposed algorithm yielded a coding gain of approximately 2.9% in a high efficiency configuration random access mode.

***Keywords*:** Multi-view video, HEVC, Inter-view prediction

## 1. Introduction

For 20 years, 3D video compression methods have been developed alongside 2D video compression technology. The MPEG-2 multi-view profile (MVP) and MPEG-4 multiple auxiliary component (MAC) standards deal with stereoscopic video. Even with these available technologies, the 3D market was not large in the past due to the lack of 3D content and infrastructure, such as production, distribution, and consumption. On the other hand, the H.264/AVC multi-view video coding (MVC) was developed not only for stereoscopic video, but also for multi-view video. Along with the development of these 3D technologies, several recent 3D movies have become popular and the level of 3D content is increasing. In addition, the market for 3D terminals, such as stereoscopic televisions is growing rapidly. Despite this, more efficient coding technology is needed to improve the 3D market. Recently, ISO/IEC MPEG and ITU-T VCEG formed a joint collaborative team on video coding (JCT-VC) to develop the High Efficiency Video Coding (HEVC) standard, whose compression performance is twice that of the H.264/AVC [1-3]. HEVC has been extended to scalable and multi-view video [4-8]. Hence, efficient multi-view coding tools based on the HEVC coding structure are needed.

H.264/AVC, HEVC and prior video coding technologies employ motion vector predictions to reduce the number of bits used for motion vectors. They make use of one of (or a combination of) motion vectors of the neighboring blocks as a prediction motion vector (PMV). H.264/AVC employs a median operation of motion vectors of the left, top, and top-left blocks [9]. Jung et al. proposed a motion vector prediction method to indicate which motion vector is used to provide explicit information [10]. Yang et al. conducted motion estimation at the decoder side to reduce the motion vector information to be transmitted [11]. Similarly, Jang et al. reduced the motion vector information using an adaptive motion vector resolution [12]. In addition, Winken et al. employed an interleaved motion vector prediction for motion vector representation [13]. This algorithm efficiently coded the horizontal motion vector components. First, this algorithm employed the median operation of the vertical motion vectors, and the vertical component was coded relative to the median value. One of the neighboring PMVs was selected by choosing the PMV that was most similar to the decoded vertical component. The horizontal component of the PMV is used to predict the horizontal component of the current PMV. McCann et al. proposed a motion prediction method that was similar to Jung's method for HEVC [14]. These conventional motion vector prediction algorithms employ temporal and spatial motion vectors in 2D video. For multi-view video coding, H.264/MVC also employs the median operation on spatially neighboring motion vectors to obtain a prediction. In general, a motion vector for multi-view video tends to have a strong correlation with not only neighboring motion vectors in the same view, but also those in the adjacent views [15]. H.264/AVC MVC, however, does not sufficiently consider the similarity of the motion vectors across the different adjacent views.

This paper proposes a new consolidated motion and disparity prediction method. The proposed algorithm can adaptively set the neighboring motion vectors and disparity vectors in the same view, the motion vectors in the adjacent view and the global disparity between the two views as motion and disparity prediction candidates. The proposed motion and disparity prediction was conducted based on an advanced motion vector prediction (AMVP), which is one of the HEVC tools for motion vector predictions. The candidates for predicting the motion vectors were the motion vectors of the neighboring blocks in the same view and those in the adjacent view. The candidates for predicting the disparity were the disparity vectors in the same view and the global disparity between two views. The proposed motion and disparity prediction was performed adaptively within a consolidated prediction framework. As a result, the proposed algorithm achieved higher utilization of the temporal and spatial redundancy, as well as view redundancy.

This paper is organized as follows. In Section II, the conventional motion prediction is discussed. Section III reports the proposed motion and disparity prediction algorithm. Section IV includes a performance evaluation and a discussion of the proposed algorithm. Section V concludes the paper and provides further research topics.

## 2. Conventional Motion Vector and Disparity Prediction

As mentioned above, in recent video coding standards, the motion vectors are predicted and only the residuals are coded. In addition, disparity estimation among the adjacent views was also employed to improve the coding efficiency for multi-view video coding. This section discusses several conventional motion vector predictions and disparity prediction methods are discussed.

For most of the quantization parameter (QP) range, the number of bits used for motion vectors (MVs) is the second to the amount for residual data. In particular, the MV portion increases significantly for higher QP values. For example, MVs comprise up to 40% of the total bitrate for low bitrate applications with H.264/AVC [16]. Therefore, it is very important to code the MVs efficiently.

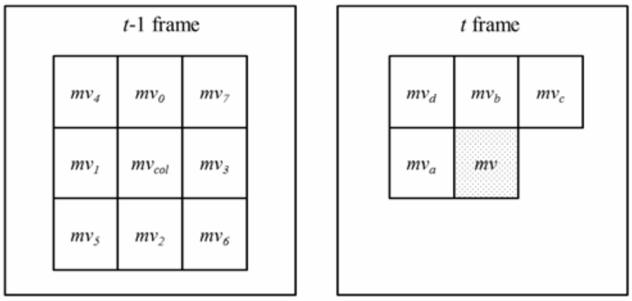



**Fig. 1. Spatial and temporal neighboring motion vectors.**

In the H.264/AVC standard, a MV is not coded by itself. Rather, the difference between the MV and its prediction is coded. The difference is defined as

$$\varepsilon_{mv} = mv - p \tag{1}$$

where $mv$ is the estimated motion vector and $p$ is its predicted vector. For the coding efficiency, it is important to make a prediction that is close to the actual MV. In H.264/AVC, the prediction vector was calculated using the MVs of three neighboring blocks.

Fig. 1 shows the spatial and temporal neighboring MVs. The target motion vector, $mv$, of the $N$-th frame was predicted using the median of the top, left, and top-left vectors, and can be represented as

$$p = \text{Med}\,(mv_a, mv_b, mv_c) \tag{2}$$

Motion vector competition was proposed to improve the coding gain against H.264/AVC. This algorithm sends the indication information that points to the best prediction candidate among the spatial and temporal neighboring ones. This method employs more candidates including not only the spatial MVs, but also a temporally-collocated predictor ($mv_{col}$) and a spatial-temporal predictor ($mv_i$, $i$=0,…,7). The best predictor is selected based on the RD optimization. The cost can be calculated using the bits for the motion difference and additional bits for the indicator value. The bit cost can be defined as

$$R_{mv/mm} = \min\,\{\varsigma(\varepsilon_{mvi}) + \varsigma(i)\},\ i \in \{1, \ldots, n\} \tag{3}$$
$$\varepsilon_{mvi} = mv - p_i$$

where $\varepsilon_{mvi}$ is the motion difference, $i$ represents the indicator value, $n$ is the number of prediction candidates and $\varsigma(\bullet)$ is the number of bits to represent the data. This prediction method was reported to yield an approximately 5% coding gain using SKIP and DIRECT modes.

The advanced motion vector prediction (AMVP), which is similar to the previously mentioned algorithm [9], was adopted as one of the HEVC coding tools to predict the MVs [11]. Instead of sending the MV itself for the current block, this algorithm explicitly transmits a candidate index, the reference index of the current block and the motion vector difference (MVD) for predicting the motion vector, while minimizing the bits required

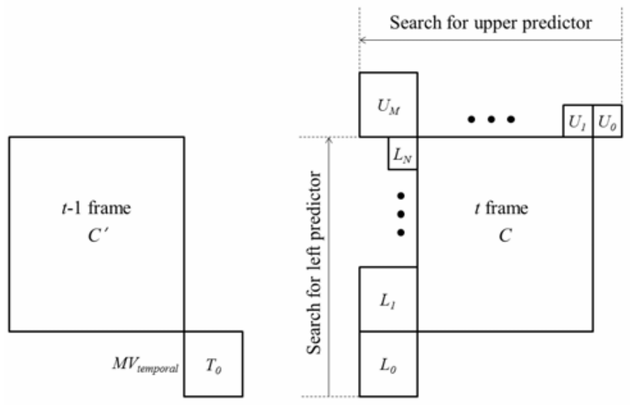



**Fig. 2. Temporal and spatial candidates for HEVC.**

representing them. The AMVP can have up to three candidates: left, upper and temporal motion vectors, as shown in Fig. 2. The candidate index is chosen from one of them. On the decoder side, a candidate vector set for the AMVP is formed by finding the motion vector candidates with the same reference index as the current block, $C$, which is received from the encoder. The candidate vector set, $\Psi_{AMVP}$, is denoted as

$$\Psi_{AMVP} = \{MV(C,d) \mid d \in \{\text{`}L\text{'}, \text{`}U\text{'}, \text{`}T\text{'}\}\} \tag{4}$$

$$MV(C,\text{`}L\text{'}) = \begin{cases} MV(L_{MIN(\Phi)}), \Phi = \{n \mid RI(C) == RI(L_n)\} \\ \varnothing \quad , \text{otherwise} \end{cases} \tag{5}$$

$$MV(C,\text{`}U\text{'}) = \begin{cases} MV(L_{MIN(\Phi)}), \Phi = \{m \mid RI(C) == RI(U_m)\} \\ \varnothing \quad , \text{otherwise} \end{cases} \tag{6}$$

$$MV(C,\text{`}T\text{'}) = \begin{cases} MV(T_0), \text{subject to } \exists MV(T_0) \\ \varnothing \quad , \text{otherwise} \end{cases} \tag{7}$$

where $MV(C, d)$ is one of the three spatial/temporal neighboring MVs of the current block, $C$. $RI(C)$ is the reference index of the current block. The HEVC supports diverse prediction unit (PU) sizes from 4×4 to 64×64. Therefore, the current block can have multiple blocks with the same reference index for each direction. Therefore, AMVP needs to select one candidate MV within the neighboring blocks for each direction. When there are $N$ blocks to the left, as shown Fig. 2, the candidate motion vector for the 'left' direction, $MV(C, \text{`}L\text{'})$, is determined by (5), where $MV(\bullet)$ gives the motion vector for the block with the minimum $n$. If there are no blocks with the same reference index for the 'left' direction, the left candidate vector is unavailable. In the same manner, the candidate motion vector for the 'upper' direction, $MV(C, \text{`}U\text{'})$, is obtained by finding the motion vector of the block with the minimum block index among those with the same reference index from the lower to upper for $m$ blocks, as defined by (6). The candidate vector for the 'temporal' direction, $MV(C, \text{`}T\text{'})$, is determined by the motion vector of the bottom-right block of the one collocated in the previous frame, as shown in Fig. 2. If such a MV is unavailable, the temporal motion vector cannot be used in the candidate set. In addition, AMVP works with the SKIP and merge modes without using a reference index [1].

A multi-view video coding (MVC) standard was released based on H.264/AVC, this standard incorporates an inter-view prediction. The temporal views and neighboring views are used as the reference frames. As a result, the disparity estimation/compensation and prediction can be conducted using the same tools as in H.264/AVC. The estimated disparity is predicted from one of the motion or disparity vectors in the neighboring blocks. For example, the vector is used as a predictor if there is only one disparity vector in the neighboring blocks. If there are no disparity vectors in the neighboring blocks, the median of the neighboring temporal motion vectors is used to predict the current disparity vector. In this case, the prediction from neighboring temporal motion vectors is inappropriate for the disparity predictor.

On the other hand, several disparity estimation algorithms have been proposed to reduce the computational complexity rather than improve the coding efficiency. L. Shen *et al.* proposed fast disparity and motion estimation methods based on the motion homogeneity [17]. The disparity estimation was conducted adaptively depending on the spatial distribution of the motion field. Z. Deng *et al.* proposed a fast motion and disparity estimation algorithm based on the stereo motion consistency [18]. These algorithms could reduce significantly the computational complexity of the disparity and motion estimation. Similarly, W. Zhu *et al.* proposed a fast algorithm to restrict the search areas for a disparity estimation based on spatio-temporal correlation of the disparity field [19]. D. Han *et al.* employed a fast mode decision using a global disparity vector among the views [20]. J. Lu *et al.* also proposed a method to reduce the computational complexity of the disparity estimation based on the epi-polar constraints [21]. To improve the coding efficiency, X. Guo *et al.* proposed the inter-view direct mode that does not send the disparity vector with neighboring disparities [22]. The neighboring disparity was derived based on a global disparity model. This is considered to be an additional prediction mode and inter-view prediction for all cases that are not dealt with effectively.

In previous studies, the motion or disparity prediction methods for multi-view video have been ineffective. To improve the coding efficiency of multi-view video coding, a new disparity prediction method is required, this method should also be effective for predicting the temporal motion.

## 3. The Proposed Motion Vector and Disparity Prediction for Multi-view Video Coding

This section presents motion and disparity prediction method to be applied to a multi-view extension of HEVC. Initially, the HEVC was simply extended to a multi-view video coder for implementing the proposed algorithm. The multi-view extension of HEVC was designed based on the same framework as H.264/MVC. The base view was coded and the reconstructed base view was used as an additional reference frame for auxiliary views. The

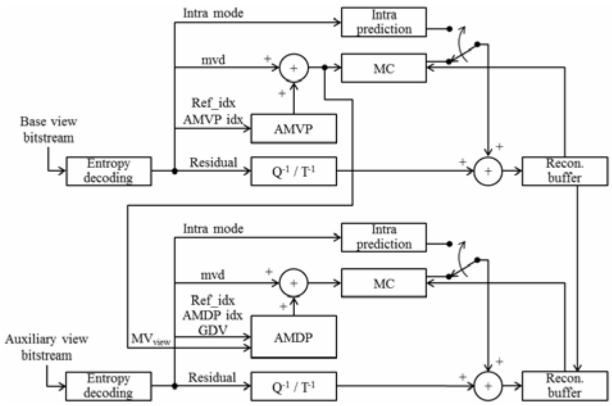



**Fig. 3. Block diagram of the multi-view video encoder with the proposed motion prediction and disparity prediction.**

reconstructed base view was fed into the decoding picture buffer for auxiliary views. The motion and disparity compensation was conducted adaptively using the additional reference frames from the adjacent views. In the proposed multi-view extension, the auxiliary views can be coded by referring to one or two neighboring views along with a spatial/temporal prediction.

Fig. 3 presents a block diagram of the multi-view video decoder with the proposed motion and disparity vector predictions. The two decoding paths (base and auxiliary view) use the same HEVC coding tools. On the other hand, the decoder for the auxiliary view includes the proposed advanced motion and disparity prediction (AMDP) tool. The proposed AMDP employs the additional motion vectors of the corresponding blocks in the previously decoded neighboring views that are captured at the same time at the current coding picture. As mentioned before, the textures of neighboring views are strongly correlated and have a similar temporal motion tendency to the current view. In addition, based on the inter-view correlation, the temporal motion information from the neighboring views can be used as an additional candidate of the motion prediction. The performance of the motion prediction of the auxiliary view can be improved as more motion candidates are employed.

To improve the coding efficiency in the multi-view video configuration, the proposed AMDP was designed to efficiently send motion or disparity vectors for the current block by exploiting the correlation of motion and disparity vectors in the temporal and view directions, respectively. The decoder with the proposed AMDP is supposed to receive a candidate index, reference index (Ref_idx) and vector difference (VD) from the encoder. In addition, the decoder also receives a global disparity vector (GDV) that is defined between the base view and the auxiliary view for each slice header, and is used for the disparity vector prediction. By looking at the reference index, the decoder can determine if the current block to be decoded is coded with the motion or disparity vector. As a result, the proposed AMDP can use the candidate predictor set composed of either motion or disparity vectors. As a result, the motion or disparity prediction is conducted adaptively.

The candidate predictor vector set for the proposed AMDP, $\Psi_{AMDP}$, is denoted as

$$\Psi_{AMDP}=\{CV(C,d)\,|\,d\in\{`L\text{'}, `U\text{'}, `T\text{'}, `V\_L\text{'}, `V\_R\text{'}\}\} \tag{8}$$

$$CV(C,`L\text{'})=\begin{cases} MV(L_{MIN(\Phi)}), & \Phi=\{n\,|\,T(C)==`MV\text{'}\ \&\&\ RI(C)==RI(L_n)\} \\ DV(L_{MIN(\Phi)}), & \Phi=\{n\,|\,T(C)==`DV\text{'}\ \&\&\ RI(C)==RI(L_n)\} \\ \varnothing, & \text{otherwise} \end{cases} \tag{9}$$

$$CV(C,`U\text{'})=\begin{cases} MV(L_{MIN(\Phi)}), & \Phi=\{m\,|\,T(C)==`MV\text{'}\ \&\&\ RI(C)==RI(U_m)\} \\ DV(L_{MIN(\Phi)}), & \Phi=\{m\,|\,T(C)==`DV\text{'}\ \&\&\ RI(C)==RI(U_m)\} \\ \varnothing, & \text{otherwise} \end{cases} \tag{10}$$

$$CV(C,`T\text{'})=\begin{cases} MV(T_0), & \text{subject to } T(C)==`MV\text{'}\ \&\&\ \exists MV(T_0) \\ DV(T_0), & \text{subject to } T(C)==`DV\text{'}\ \&\&\ \exists DV(T_0) \\ \varnothing, & \text{otherwise} \end{cases} \tag{11}$$

$$CV(C,`V\_L\text{'})=\begin{cases} MV(V_{LRG(\Phi)}), & \Phi=\{k\,|\,T(C)==`MV\text{'}\ \&\&\ RI(C)==RI(V_k)\} \\ GV(`V\_L\text{'}), & \text{subject to } T(C)==`DV\text{'} \\ \varnothing, & \text{otherwise} \end{cases} \tag{12}$$

$$CV(C,`V\_R\text{'})=\begin{cases} MV(V_{LRG(\Phi)}), & \Phi=\{k\,|\,T(C)==`MV\text{'}\ \&\&\ RI(C)==RI(V_k)\} \\ GV(`V\_R\text{'}), & \text{subject to } T(C)==`DV\text{'} \\ \varnothing, & \text{otherwise} \end{cases} \tag{13}$$

where the candidate vector, $CV(C, d)$, represents a candidate vector for each direction, $d$. The proposed AMDP set includes view directions (‘$V\_L$’ and ‘$V\_R$’) in addition to the three directions allowed in AMVP. Note that $T(C)$ represents the prediction type, either ‘$MV$’(motion vector) or ‘$DV$’(disparity vector), which is identified by the reference index of the current block. The candidate vector for the ‘left’ direction, $CV(C, `L\text{'})$, was determined by (9), where $DV(\bullet)$ gives a disparity vector for the input block. If the prediction type for the current block is ‘$MV$,’ the candidate vector is selected from the blocks with motion vectors. On the other hand, if the prediction type is ‘$DV$’, it is selected from those with disparity vectors. Depending on the prediction type, the candidate vector for the ‘left’ direction is the motion or disparity vector with a minimum block index. In a similar manner, the candidate vector for the ‘upper’ direction, $CV(C, `U\text{'})$, was obtained using (10). The candidate vector for the ‘temporal’ direction, $CV(C, `T\text{'})$, comes from one of the bottom-right blocks of the collocated block in the previous frame. The candidate can be either a motion or disparity vector depending on the prediction type of the bottom-right block.

For the ‘view’ direction, an auxiliary view can have one or two adjacent views. The candidate vectors for the ‘view’ direction are determined according to the reference structure. Two candidates are used when both left and right reference views are available. Otherwise, one of them would be employed for the ‘view’ direction candidate. Note that the reference structure is defined in the sequence

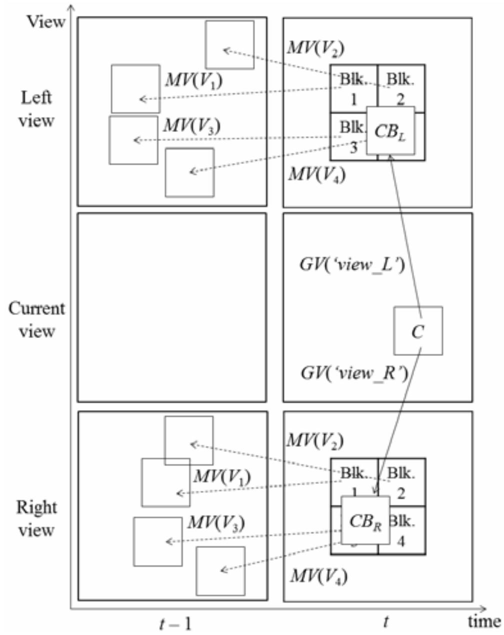



**Fig. 4. View-direction candidates from the overlapped corresponding blocks.**

parameter set (SPS) for each view. The candidate vector for the 'left view' direction, *CV*(*C*, '*V_L*'), and the candidate vector for the 'right view' direction, *CV*(*C*, '*V_R*'), are given by (12) and (13), respectively. When the prediction type of the current block is '*MV*,' the view candidate vector becomes one of the motion vectors of multiple blocks that overlap with the corresponding block (*CB*) in the adjacent view, as shown in Fig. 4. The corresponding block is determined using the received global disparity. The corresponding block might not be aligned with the prediction units in the adjacent view. Note that there can be a maximum of four overlapped blocks. Hence, $k$ can range from 1 to 4. One of them is selected depending on the largest overlapped area, which is found using the operation, *LRG*(•). This operation provides the index of the block with the largest overlapped area among all the overlapped blocks with the same reference index [23]. As an illustration, assume that the reference index of all the overlapped blocks for two adjacent views is the same as that of the current block, as shown in Fig. 4. The fourth block of the left view and the third block of the right view have a large overlap with the corresponding blocks. Therefore, the motion vector of the fourth block of the left view and the motion vector of the third block of the right view are included in the set of view candidate predictors.

When the prediction type of the current block is '*DV*', the global disparity vector (GDV) is used as the view-direction candidate. *GV*('*V_L*') represents the global disparity vector between the current view and the adjacent view. Regardless of the reference index and prediction type, the GDVs are always available for predicting the disparity. The GDV for a slice is computed at the encoder side that is then transferred to the decoder side. In this paper, GDVs are calculated based on the perspective motion model [24-25]. Assume that each adjacent view has one GDV per frame. The perspective model has eight degrees of freedom and can be defined as

$$\begin{pmatrix} x' \\ y' \\ 1 \end{pmatrix} = S\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} + R\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} + T\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} m_1 & m_2 & m_3 \\ m_4 & m_5 & m_6 \\ m_7 & m_8 & 1 \end{pmatrix}\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \quad (14)$$

where $(x', y')$ and $(x, y)$ represent the pixel positions of two views. $S$, $R$, and $T$ are the scale, rotation, and translation parameters, respectively. This paper reports the translation parameters ($m_3$, $m_6$) for the disparity prediction. On the encoder side, eight parameters are extracted but only two translational parameters are transmitted to the decoder side. In general, the global disparity vectors do not vary over short periods of time unless a large camera zoom or scene change occurs. An initial global disparity vector is transmitted in the slice header for the IDR slices and the prediction vectors between consecutive disparity vectors are then transmitted for consecutive slices. Table 1 lists all the candidate possibilities depending on the neighboring and temporal prediction types.

On the encoder side, the best motion vector for each reference frame is calculated at the motion estimation stage. At a given motion vector and reference index, the best AMDP index is determined by finding the minimum number of bits required to encode both the vector difference and the AMDP index to be transmitted. The best AMDP index is obtained by

$$D_{AMDP} = MIN_d[\varsigma(MV(C) - CV(C, d)) + \varsigma(d)] \quad (15)$$
$$d \in \{\text{'}L\text{'}, \text{'}U\text{'}, \text{'}T\text{'}, \text{'}V\_L\text{'}, \text{'}V\_R\text{'}\}$$

where *MV*(*C*) represents the motion or disparity vector for the current block and ς(•) gives the number of bits required to code the input symbol.

## 4. Experimental Results

The proposed AMDP for a multi-view video was implemented on a simple multi-view coder based on HEVC. In these experiments, HM2.0 [26], which was released by JCT-VC as the reference software, was used as the inter-view prediction structure that is the same as that of H.264/MVC. For the evaluation, several multi-view sequences, whose resolutions are 1024×768p (Class C) at 30fps and 1920×1088p (Class A) at 25fps, were used [27]. Table 2 lists view identifiers and QP values for the two-view and three-view cases.

The coding parameters were set to those specified for the random access mode for a high efficiency configuration in HEVC [28]. The coding unit (CU) sizes were set from 8×8 to 64×64, and transform unit (TU) sizes ranged from 4×4 to 32×32. Context-based adaptive binary arithmetic coding (CABAC) was used for the entropy coding and the adaptive loop filter (ALF) tool was enabled. A hierarchical coding structure was used and its GOP size was set to 8. Fig. 5 shows the coding structure for the three-view case. The outer view can be coded using the

**Table 1. Candidate vector set according to coding mode of the neighboring blocks and target block.**

| Left block | Upper block | Temporal block | Candidate sets for the current block | |
|---|---|---|---|---|
| | | | Motion prediction | Disparity prediction |
| Motion | Motion | Motion | $MV_{left}$, $MV_{up}$, $MV_{temporal}$, $MV_{view}$ | $DV_{global}$ |
| Motion | Motion | Inter-view | $MV_{left}$, $MV_{up}$, $MV_{view}$ | $DV_{temporal}$, $DV_{global}$ |
| Motion | Motion | Intra | $MV_{left}$, $MV_{up}$, $MV_{view}$ | $DV_{global}$ |
| Motion | Inter-view | Motion | $MV_{left}$, $MV_{temporal}$, $MV_{view}$ | $DV_{up}$, $DV_{global}$ |
| Motion | Inter-view | Inter-view | $MV_{left}$, $MV_{view}$ | $DV_{up}$, $DV_{temporal}$, $DV_{global}$ |
| Motion | Inter-view | Intra | $MV_{left}$, $MV_{view}$ | $DV_{up}$, $DV_{global}$ |
| Motion | Intra | Motion | $MV_{left}$, $MV_{temporal}$, $MV_{view}$ | $DV_{global}$ |
| Motion | Intra | Inter-view | $MV_{left}$, $MV_{view}$ | $DV_{temporal}$, $DV_{global}$ |
| Motion | Intra | Intra | $MV_{left}$, $MV_{view}$ | $DV_{global}$ |
| Inter-view | Motion | Motion | $MV_{up}$, $MV_{temporal}$, $MV_{view}$ | $DV_{left}$, $DV_{global}$ |
| Inter-view | Motion | Inter-view | $MV_{up}$, $MV_{view}$ | $DV_{left}$, $DV_{temporal}$, $DV_{global}$ |
| Inter-view | Motion | Intra | $MV_{up}$, $MV_{view}$ | $DV_{left}$, $DV_{global}$ |
| Inter-view | Inter-view | Motion | $MV_{temporal}$, $MV_{view}$ | $DV_{left}$, $DV_{up}$, $DV_{global}$ |
| Inter-view | Inter-view | Inter-view | $MV_{view}$ | $DV_{left}$, $DV_{up}$, $DV_{temporal}$, $DV_{global}$ |
| Inter-view | Inter-view | Intra | $MV_{view}$ | $DV_{left}$, $DV_{up}$, $DV_{global}$ |
| Inter-view | Intra | Motion | $MV_{temporal}$, $MV_{view}$ | $DV_{left}$, $DV_{global}$ |
| Inter-view | Intra | Inter-view | $MV_{view}$ | $DV_{left}$, $DV_{temporal}$, $DV_{global}$ |
| Inter-view | Intra | Intra | $MV_{view}$ | $DV_{left}$, $DV_{global}$ |
| Intra | Motion | Motion | $MV_{up}$, $MV_{temporal}$, $MV_{view}$ | $DV_{global}$ |
| Intra | Motion | Inter-view | $MV_{up}$, $MV_{view}$ | $DV_{temporal}$, $DV_{global}$ |
| Intra | Motion | Intra | $MV_{up}$, $MV_{view}$ | $DV_{global}$ |
| Intra | Inter-view | Motion | $MV_{temporal}$, $MV_{view}$ | $DV_{up}$, $DV_{global}$ |
| Intra | Inter-view | Inter-view | $MV_{view}$ | $DV_{up}$, $DV_{temporal}$, $DV_{global}$ |
| Intra | Inter-view | Intra | $MV_{view}$ | $DV_{up}$, $DV_{global}$ |
| Intra | Intra | Motion | $MV_{temporal}$, $MV_{view}$ | $DV_{global}$ |
| Intra | Intra | Inter-view | $MV_{view}$ | $DV_{temporal}$, $DV_{global}$ |
| Intra | Intra | Intra | $MV_{view}$ | $DV_{global}$ |

**Table 2. Test sets for two view and three view.**

| Test Sequence | | two-view | | three-view | |
|---|---|---|---|---|---|
| | | View IDs | QP | View IDs | QP |
| Class A | Poznan_Hall2 | 7-6 | 49, 45, 41, 37 | 7-6-5 | 49, 45, 41, 37 |
| | Poznan_Street | 4-3 | 51, 47, 43, 39 | 5-4-3 | 51, 47, 43, 39 |
| | Undo_Dancer | 2-5 | 51, 49, 45, 43 | 1-5-9 | 51, 47, 45, 41 |
| | GT_Fly | 5-2 | 51, 47, 43, 41 | 9-5-1 | 51, 47, 43, 41 |
| Class C | Kendo | 3-5 | 49, 45, 43, 39 | 1-3-5 | 51, 47, 43, 39 |
| | Balloons | 3-5 | 49, 47, 43, 39 | 1-3-5 | 51, 47, 43, 39 |
| | Lovebird1 | 6-8 | 49, 47, 43, 39 | 4-6-8 | 51, 47, 43, 39 |
| | Newspaper | 4-6 | 51, 47, 45, 41 | 2-4-6 | 51, 49, 45, 43 |

inter-view prediction from the base view, as shown in Fig. 5. In addition, the center view was coded with the bi-directional inter-view prediction from both the base and outer views. Except for the first frame of the base view, all other frames were coded as B-frames using the generalized P and B-picture (GPB) tool in HEVC.

Table 3 lists the percentage of selected vector predictor types for the AMVP and the proposed AMDP. The percentages of the selected vectors for 'temporal,' 'left,' and 'upper' directions for the AMVP were approximately 5.3%, 54.6%, and 40.1%, respectively. For the proposed AMDP, the selection percentages for 'temporal,' 'left,' 'upper,' and 'view' directions were approximately 1.2%, 50.6%, 35.5%, and 12.7%, respectively. As shown in the table, the selection percentage for the view direction was relatively high, which means that the proposed view candidates can offer good predictors for the motion and disparity prediction.

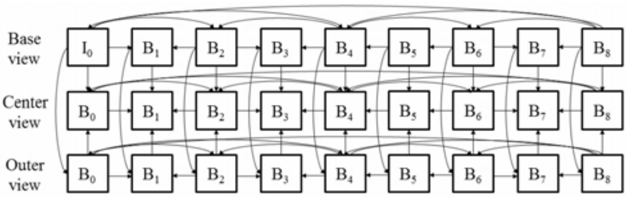
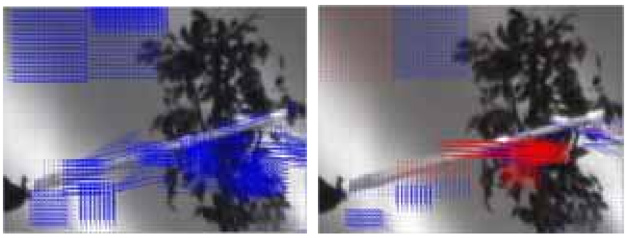



**Fig. 5. Coding structure for the proposed method in the three-view case.**



(a) (b)

**Fig. 6. Motion fields of motion vector difference (a) AMVP, (b) AMDP.**

**Table 3. Selection ratios of the candidate vectors for the AMVP and the AMDP.**

| Test Sequence | Conventional AMVP | | | Proposed AMDP | | | |
|---|---|---|---|---|---|---|---|
| | Temporal (%) | Left (%) | Upper (%) | Temporal (%) | Left (%) | Upper (%) | View (%) |
| Kendo | 9.2 | 58.4 | 32.4 | 2.0 | 47.7 | 24.7 | 25.6 |
| Balloons | 4.2 | 48.6 | 47.2 | 0.4 | 43.8 | 43.3 | 12.5 |
| Lovebird1 | 4.4 | 56.3 | 39.3 | 1.4 | 55.7 | 35.9 | 7.0 |
| Newspaper | 3.5 | 55.0 | 41.5 | 0.9 | 55.1 | 38.2 | 5.8 |
| Average | 5.3 | 54.6 | 40.1 | 1.2 | 50.6 | 35.5 | 12.7 |

The proposed AMDP can reduce the number of bits required to encode the prediction error between the actual vector to be coded for the current block and the predictor vector. For example, Fig. 6 shows the motion fields of prediction error vectors for the conventional AMVP and the proposed AMDP for frame 0 of the 'Kendo' sequence. Regardless of the prediction block (PU) sizes, Fig. 6 depicts the motion vector difference for each 4×4 block. The blue arrows represents the prediction error vectors for conventional AMVP. The red arrows represent the prediction error vectors selected by the proposed 'view' candidate. Fig. 6 illustrates the ability of the proposed AMDP to reduce the magnitude of the prediction error vectors.

Tables 4-6 show the coding performance of the proposed method for two-view and three-view cases, respectively. Simulcast coding for each view was conducted as an anchor. The BD-bitrate and BDPSNR of the inter-view HEVC coders with a conventional AMVP and the proposed AMDP were evaluated against the simulcast HEVC [29]. For the two-view case, the inter-view HEVC yielded a bit reduction of approximately 50.3%. The proposed method also provided an additional coding gain of 3.8% over the inter-view HEVC with

**Table 4. Comparisons of the RD performance for two-view case.**

| Sequence | Multi-view system | | The proposed AMVP | |
|---|---|---|---|---|
| | BD PSNR | BD Bitrate | BD PSNR | BD Bitrate |
| Poznan_Hall2 | 1.62 | -35.24 | 1.87 | -40.59 |
| Poznan_Street | 3.36 | -61.57 | 3.79 | -66.35 |
| Undo_Dancer | 2.48 | -58.48 | 2.63 | -60.88 |
| GT_Fly | 3.02 | -65.92 | 3.06 | -67.58 |
| Average | 2.62 | -55.30 | 2.84 | -58.85 |
| Kendo | 2.47 | -37.85 | 2.70 | -40.84 |
| Balloons | 2.55 | -40.17 | 2.87 | -43.99 |
| Lovebird1 | 3.70 | -59.98 | 4.17 | -63.83 |
| Newspaper | 2.78 | -42.91 | 3.29 | -49.05 |
| Average | 2.87 | -45.23 | 3.26 | -49.43 |
| Total average | 2.75 | -50.27 | 3.05 | -54.14 |

**Table 5. Comparisons of RD performance for three-view case (outer view).**

| Sequence | Multi-view system | | The proposed AMVP | |
|---|---|---|---|---|
| | BD PSNR | BD Bitrate | BD PSNR | BD Bitrate |
| Poznan_Hall2 | 2.05 | -41.6 | 2.27 | -46.09 |
| Poznan_Street | 4.57 | -70.7 | 4.98 | -73.28 |
| Undo_Dancer | 2.79 | -61.9 | 2.93 | -63.8 |
| GT_Fly | 3.22 | -67.53 | 3.4 | -70.64 |
| Average | 3.16 | -60.43 | 3.39 | -63.45 |
| Kendo | 4.21 | -53.19 | 4.48 | -55.5 |
| Balloons | 3.72 | -52.33 | 4.21 | -56.61 |
| Lovebird1 | 4.5 | -65.66 | 5.03 | -69.55 |
| Newspaper | 3.79 | -51.85 | 4.3 | -57.05 |
| Average | 4.05 | -55.76 | 4.51 | -59.68 |
| Total average | 3.61 | -58.1 | 3.95 | -61.57 |

**Table 6. Comparisons of RD performance for three-view case (center view).**

| Sequence | Multi-view system | | The proposed AMVP | |
|---|---|---|---|---|
| | BD PSNR | BD Bitrate | BD PSNR | BD Bitrate |
| Poznan_Hall2 | 0.98 | -23.63 | 1.21 | -28.69 |
| Poznan_Street | 2.47 | -51.08 | 2.83 | -55.95 |
| Undo_Dancer | 1.62 | -43.8 | 1.78 | -46.85 |
| GT_Fly | 1.65 | -46.15 | 1.91 | -52.15 |
| Average | 1.68 | -41.16 | 1.93 | -45.91 |
| Kendo | 1.42 | -23.68 | 1.62 | -26.64 |
| Balloons | 1.81 | -30.58 | 2.07 | -34.08 |
| Lovebird1 | 2.84 | -50.45 | 3.32 | -55.64 |
| Newspaper | 1.07 | -20.59 | 1.41 | -25.9 |
| Average | 1.78 | -31.32 | 2.1 | -35.57 |
| Total average | 1.73 | -36.25 | 2.02 | -40.74 |

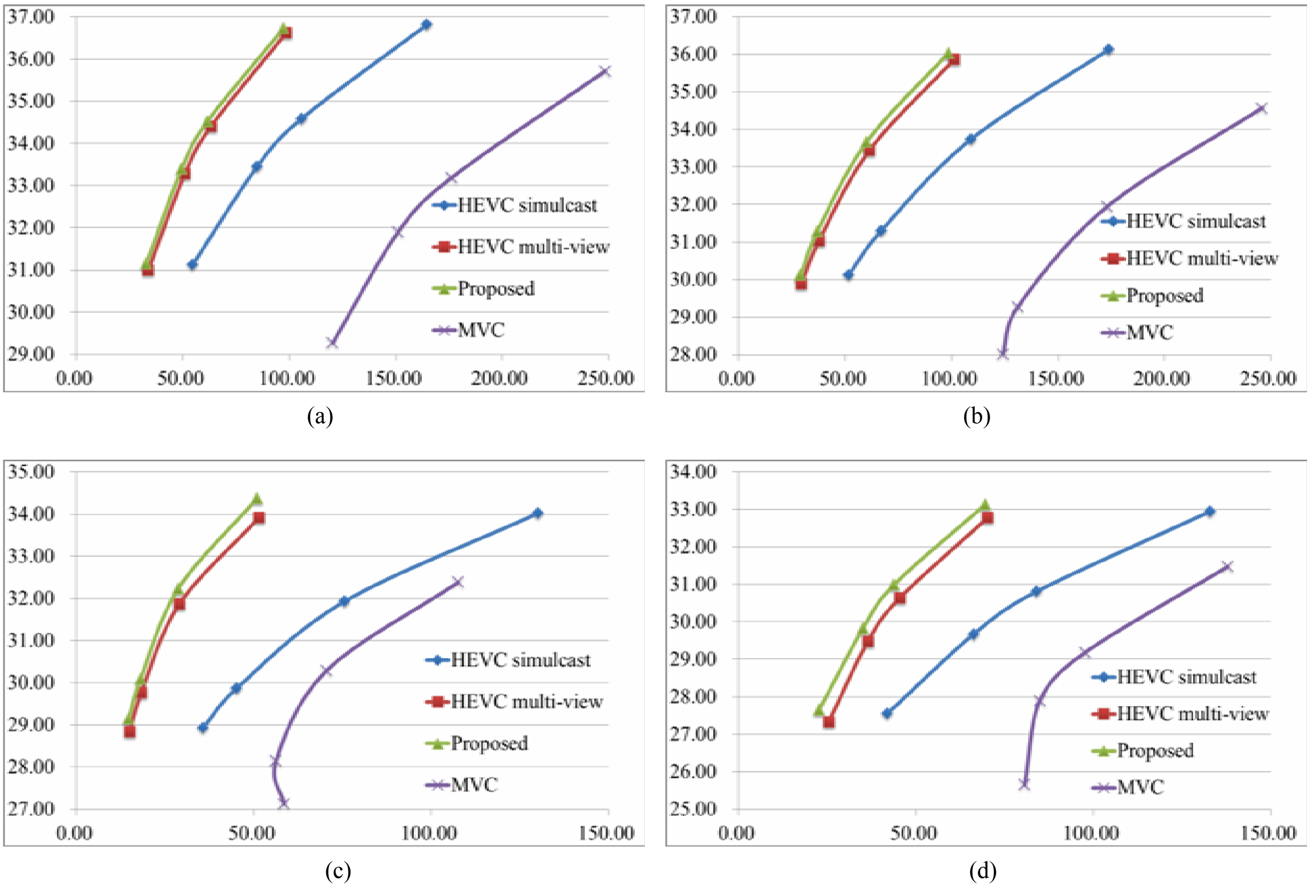



**Fig. 7. Comparison of the RD performance (a) Kendo, (b) Balloons, (c) Lovebird1, (d) Newspaper.**

conventional AMVP. For the three-view configuration, significant coding gains for the outer and center views were achieved, which are approximately 36.3% and 58.1% in BD-Bitrate for the outer and center views, respectively. In addition, the proposed AMDP yielded an additional coding gain of 4.4% for the outer view and 3.5% for the center view. The coding gain for the center was approximately 7.5% higher than that for the two-view cases in terms of bit-rate reduction. For the center views of the three-view cases, two view candidates for motion or disparity prediction can be employed, whereas only one view candidate can be used in the two-view cases.

Fig. 7 shows the operational RD graphs for H.264/AVC MVC, HEVC simulcast, inter-view HEVC and the proposed method on the inter-view HEVC for the Class C test sequences. As shown in Fig. 7, the HEVC simulcast yielded coding gains of 1 to 4dB compared to MVC. On the other hand, multi-view HEVC with AMVP can achieve RD gains of 2 to 3dB compared to the HEVC simulcast case. In addition, the proposed method (multi-view HEVC + AMDP) clearly achieved some additional coding gain due to the better motion vector prediction. In particular, the proposed method provided better coding efficiency at low bitrates.

## 5. Conclusion

An advanced motion and disparity prediction method was proposed for multi-view video coding based on HEVC. The proposed method uses the candidate predictor vectors of 'view' direction to exploit the inter-view correlation. The candidates were used to effectively predict the motion or disparity vectors to be coded. To evaluate the proposed Advanced Motion and Disparity Prediction (AMDP) method, a basic multi-view HEVC was implemented and evaluated. The experimental results showed that the proposed method generates coding gains of approximately 3.9% and 4.0% for the two-view and three-view cases, respectively, compared to the conventional Advanced Motion Vector Prediction (AMVP) used in the multi-view extension of H.264/AVC.

## References

[1] T. Wiegand, W. Han, B. Bross, J. Ohm, and G. Sullivan, "WD2: Working Draft 2 of High-Efficiency Video Coding," *Document of Joint Collaborative Team on Video Coding*, JCTVC-D503, Daegu, KR, Jan. 2011.

[2] K. McCann, B. Bross, S. Sekiguchi, and W. Han, "High efficiency video coding (HEVC) test model 2 (HM 2) encoder description," *Document of Joint Collaborative Team on Video Coding*, JCTVC-D502, Daegu, KR, Jan. 2011.

[3] T. Wiegand, J. Ohm, G. Sullivan, W. Han, R. Joshi, T.

Tan, and K. Ugur, "Special section on the joint call for proposals on high efficiency video coding (HEVC) standardization," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 12, pp. 1661-1666, Dec. 2010. Article (CrossRef Link)

[4] A. Luthra, "Scalable enhancement requirements for HEVC," *Document of Joint Collaborative Team on Video Coding,* JCTVC-E502, Geneva, CH, Mar. 2011.

[5] A. Luthra, J. Ohm, and J. Ostermann, "Draft call for proposals on the scalable video coding extensions of HEVC," *The 98th MPEG meeting*, N12398, Geneva, CH, Nov. 2011.

[6] ISO/IEC JTC1/SC29/WG11, "Applications and requirements on 3D video coding," *The 96th MPEG meeting*, N12035, Geneva, CH, Mar. 2011.

[7] H. Choi, J. Nam, D. Sim, and I. V. Bajić, "Scalable video coding based on high efficiency video coding (HEVC)," *IEEE Pacific Rim Conference on Communications, Computers, and Signal Processing* (*PacRim2011*), pp. 346-351, Victoria, CA, Aug. 2011. Article (CrossRef Link)

[8] J. Nam, W. Lim, D. Sim, and I. V. Bajić, "Multi-view video coding based on high efficiency video coding (HEVC)," *The 26th International Technical Conference on Circuits/Systems, Computers, and Communications*, pp. 169-172, Gyeongju, KR, Jun. 2011.

[9] T. Wiegand, G. Sullivan, and A. Luthra, "Draft ITU-T recommendation and final draft international standard of joint video specification," *Joint Video Team* (*JVT*) *of ITU-T VCEG and ISO/IEC MPEG*, JVT-G050r1, Geneva, CH, May 2003.

[10] J. Jung and G. Laroche, "Competition-based scheme for motion vector selection and coding," *ITU-T SG16/Q.6 VCEG-AC06*, Klagenfurt, DE, Jul. 2006.

[11] J. Yang, K. Won, and B. Jeon, "Motion vector coding with selection of an optimal predictive motion vector," *SPIE Optical Engineering*, vol. 48, no. 1, pp. 010501:1-3, Jan. 2009. Article (CrossRef Link)

[12] M. Jang, C. Seo, and J. Han, "Motion estimation using region-based motion vector precision," *The 26th International Technical Conference on Circuits/Systems, Computers, and Communications*, pp. 823-825, Gyeongju, KR, Jun. 2011.

[13] M. Winken, S. Bobe, B. Bross, P. Helle, T. Hinz, H. Kirchhoffer, H. Lakshman, D. Marpe, S. Oudin, M. Preib, H. Schwarz, M. Siekmann, K. Suhring, and T. Wiegand, "Description of video technology proposal by Fraunhofer HHI," *Document of Joint Collaborative Team on Video Coding, JCTVC-A116*, Dresden, DE, Apr. 2010.

[14] Ken McCann, Woo-Jin Han, Il-Koo Kim, Jung-Hye Min, Elena Alshina, Alexander Alshin, Tammy Lee, Jianle Chen, Vadim Seregin, Sunil Lee, Yoon-Mi Hong, Min-Su Cheon, and Nikolay Shlyakhov, "Samsung's Response to the Call for Proposals on Video Compression Technology," *Document of Joint Collaborative Team on Video Coding,* JCTVC-A124, Dresden, DE, Apr. 2010.

[15] D. Kim, D. Min, and K. Sohn, "A stereoscopic video generation method using stereoscopic disparity characterization and motion," *IEEE Transactions on broadcasting*, vol. 54, no. 2, pp. 188-197, Jun. 2008. Article (CrossRef Link)

[16] G. Laroche, J. Jung, and B. Pesquet-Popescu, "RD optimized coding for motion vector predictor selection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 9, pp. 1247-1257, Sept. 2008. Article (CrossRef Link)

[17] L. Shen, Z. Liu, S. Liu, Z. Zhang, and P. An, "Selective disparity estimation and variable size motion estimation based on motion homogeneity for multi-view coding," *IEEE Transactions on broadcasting*, vol. 55, no. 4, pp. 761-766, Dec. 2009. Article (CrossRef Link)

[18] Z. Deng, Y. Chan, K. Jia, C. Fu, and W. Siu, "Fast motion and disparity estimation with adaptive search range adjustment in stereoscopic video coding," *IEEE Transactions on broadcasting*, vol. 58, no. 1, pp. 24-33, Mar. 2012. Article (CrossRef Link)

[19] W. Zhu, X. Tian, F. Zhou, and Y. Chen, "Fast disparity estimation using spatio-temporal correlation of disparity field for multiview video coding," *IEEE Transactions on Consumer Electronics*, vol. 56, no. 2, pp. 957-964, May 2010. Article (CrossRef Link)

[20] D. Han, Y. Lee, and Y. Lee, "Fast mode decision using global disparity vector for multi-view video coding," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E92.A, no. 12, pp. 3407-3411, Dec. 2009. Article (CrossRef Link)

[21] J. Lu, H. Cai, J.-G. Lou, and J. Li, "An epipolar geometry-based fast disparity estimation algorithm for multiview image and video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 17, no. 6, pp. 737-750, Jun. 2007. Article (CrossRef Link)

[22] X. Guo, Y. Lu, F. Wu, and W. Gao, "Inter-view direct mode for multi-view video coding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 12, pp. 1527-1532, Dec. 2006. Article (CrossRef Link)

[23] J. Youn, M.-T. Sun, and C.-W. Lin, "Motion vector refinement for high performance transcoding," *IEEE Transactions on Multimedia*, vol. 1, no. 1, pp. 30-40, Mar. 1999. Article (CrossRef Link)

[24] Y. Su, M.-T. Sun, and V. Hsu, "Global motion estimation from coarsely sampled motion vector field and the applications," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 15, no. 2, pp. 232-242, Feb. 2005. Article (CrossRef Link)

[25] Y.-M Chen and I. V. Bajić, "A joint approach to global motion estimation and motion segmentation from a coarsely sampled motion vector field," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 21, no. 9, pp. 1316-1328, Sep. 2011. Article (CrossRef Link)

[26] http://hevc.kw.bbc.co.uk/trac/ browser/ tags/HM-2.0, HM2.0 software.

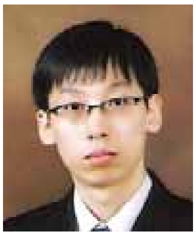
[27] ISO/IEC JTC1/SC29/WG11, "Call for proposals on

3D video coding technology," *The 96th MPEG meeting*, N12036, Geneva, CH, Mar. 2011. Article (CrossRef Link)

[28] F. Bossen, "Common Test Conditions and Software Reference Configurations," *Document of Joint Collaborative Team on Video Coding*, JCTVC-D600, Daegu, KR, Jan. 2011. Article (CrossRef Link)

[29] G. Bjøntegaard, "Calculation of average PSNR differences between RD-Curves," *ITU-T SG16/Q.6,* VCEG-M33, Austin, TX, Apr. 2001. Article (CrossRef Link)

**Woong Lim** received his B.S. and M.S. degree in computer engineering from Kwangwoon University, Seoul, Korea, in 2008 and 2010, respectively. Currently, he is a Ph.D student at the same university. His current interests are video coding, multi-view video, computer vision and multimedia systems.



**Junghak Nam** received his M.S. and Ph.D. degree in computer engineering from Kwangwoon University (KWU), Seoul, Korea, in 2008 and 2013, respectively. He is now with the Department of Advanced Technology and System Team Convergence R&D Lab, LG Electronics Inc. His research interests include next-generation video coding, 3D video processing and scalable video coding.

**Donggyu Sim** received his BS and MS degrees in electronics engineering from Sogang University, Seoul, Korea, in 1993 and 1995, respectively. He also received his PhD at the Image processing lab of the same University in 1999. He was with the Hyundai Electronics Co., Ltd from 1999 to 2000, where was involved in MPEG-7 standardization. He was a senior research engineer at Varo Vision Co., Ltd., working on MPEG-4 wireless applications from 2000 to 2002. He worked for the Image Computing Systems Lab. (ICSL) at the University of Washington as a senior research engineer from 2002 to 2005. He researched ultrasound image analysis and parametric video coding. He joined the Department of Computer Engineering at Kwangwoon University, Seoul, Korea, in 2005 as an associate professor. He was elevated to an IEEE Senior Member in 2004. His current research interests are image processing, computer vision, and video communication.