Horizontal Scaling and Shearing-Based Disparity-Compensated Prediction for Stereo Video Coding

Ka-Man Wong, *Member, IEEE,* Lai-Man Po, *Senior Member, IEEE,* Kwok-Wai Cheung, *Member, IEEE,* Chi-Wang Ting, Ka-Ho Ng, *Student Member, IEEE,* and Xuyuan Xu, *Member, IEEE*

***Abstract*—In multiview video coding (MVC), disparity-compensated prediction (DCP) exploits the correlation among different views. A common approach is to use block-based motion-compensated prediction (MCP) tools to predict the disparity effect among different views. However, some regions in different views may have various deformations due to nonconstant depth. Thus, performance of DCP is not satisfactory with the simple translational model assumed in conventional block-based MCP tools. Previous attempts to achieve better disparity prediction were usually too complex for practical use. In this paper, horizontal scaling and shearing (HSS) effects are investigated to increase interview prediction accuracy for stereo video. HSS deformations are common among images of horizontally aligned views, due to horizontal and vertical flat surfaces that are not parallel with projection image planes. To achieve HSS-based DCP with minimal complexity, an efficient subsampled block-matching technique is adopted and integrated into MVC extension of H.264/AVC in stereo profile. Affine parameters estimation and additional frame buffers are not required and the overall increase of computational complexity and memory requirements are moderate. Experimental results show that the new technique can achieve up to 5.25% bitrate reduction in interview prediction using JM17.0 reference software implementation.**

***Index Terms*—Compression, disparity-compensated prediction, horizontal scaling, horizontal shearing, multiview video coding, stereo video coding, stretching, subsampled block matching.**

I. Introduction

STEREO VIDEO has left and right views to provide depth perception to emulate human stereoscopic vision. Multiview video coding (MVC) is a key technology to enable efficient coding, storage, and transmission of video data composed of multiple views [1], including stereo video. Although MPEG-2 [2] and H.264/AVC [3] can support multiple views by interleaving the views temporarily or spatially, the coding efficiency is not satisfactory. MVC extension of H.264/AVC extends the current framework of H.264/AVC to exploit the correlations among views. Block-based disparity-compensated prediction (DCP) is adopted for interview prediction so as to utilize the existing block-based motion-compensated prediction (MCP) techniques such as multiple reference frames (MRFs) [4], variable block size (VBS) [5], subpixel MCP [6], hierarchical prediction structure [7], and fast motion estimation. The differences between views are considered as camera panning from one position to another and the prediction error is encoded by residue coding. The similarity among views is exploited by group-of-picture (GOP) structure in MVC extension for DCP [8], [9]. The block-based MCP used in the DCP scheme assumes translational motion for objects across different views. However, for a wide baseline camera distance or regions with changing depths, such translational motion assumption is not valid. Thus, high interview prediction accuracy cannot be achieved. Fig. 1 shows an example of stereo image pair. Some regions such as the ceiling and the wall with gradually changing depths are deformed between these two views due to disparity effects. Block-based translational prediction is not accurate in such cases. Sufficient coding gain for stereo image or video compression can be obtained if these deformations are considered in the prediction.

To tackle the deformations between views, higher order models such as affine [11], [12] and mesh [13]–[15] methods have been proposed for providing spatial transformed block matching. Affine multipicture motion-compensated prediction (AMMCP) [11] is a well-known higher order model-based MCP technique using affine transformation. It was proposed as an extension of MRFs to provide wrapped versions of the reference frames, and the MCP is performed on these wrapped frames. A set of affine parameters is determined by detecting the affine transforms between the current frame and the reference frame. AMMCP consists of three major steps: cluster initialization, affine refinement, and frame wrapping. First, the reference frame is partitioned with a certain number of clusters based on the number of affine-transformed frames used. Second, a translational motion vector (TMV) is determined for each cluster by the MV(s) found in conventional motion estimation. With the MV, affine parameter estimation is performed on each cluster such that the affine parameter set

Manuscript received August 19, 2011; revised November 21, 2011 and February 7, 2012; accepted February 14, 2012. Date of publication June 1, 2012; date of current version September 28, 2012. This work was supported by the City University of Hong Kong, Hong Kong, under a GRF grant with project 9041501 (CityU 119909). This paper was recommended by Associate Editor P. Salembier.

K.-M. Wong, L.-M. Po, C.-W. Ting, K.-H. Ng, and X. Xu are with the Department of Electronic Engineering, City University of Hong Kong, Kowloon, Hong Kong (e-mail: w.carlos@alumni.cityu.edu.hk; eelmpo@cityu.edu.hk; cwting@cityu.edu.hk; kahomike@gmail.com; xuyuanxu2@gmail.com).

K.-W. Cheung is with the Department of Computer Science, Chu Hai College of Higher Education, Tsuen Wan, Hong Kong (e-mail: kwcheung@chuhai.edu.hk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSVT.2012.2202073

(a) (b)

Fig. 1. Stereo image pair example. (a) Left view. (b) Right view. The 3-D effect can be viewed by the parallel eyes method.

can be obtained. Finally, the wrapped frames are generated by affine transform and the noninteger displacements are computed by cubic spline interpolation and these frames are added into the reference frame buffer. The affine motions in the video content are then estimated by applying block-matching technique with the wrapped frames. In the frame selection procedure, if a block in a wrapped frame is selected as the best matched candidate, the affine parameters of that wrapped frame will be encoded.

Apart from affine model, other types of spatial transforms can be used as the motion model for DCP if they can provide predictions that fit the current block caused by disparity effects. Mesh-based methods [13]–[15] are proposed to provide a transformed frame by building a mesh to compensate the spatial transformations caused by motion. Mesh-based DCP consists of three major components: 1) mesh nodes determination; 2) frame transformation; and 3) mesh nodes encoding. It can be applied to DCP since there are strong correlations between views and the deformations can be modeled if the mesh points are located at the correspondent points between views. To determine the mesh nodes, the most intuitive method is to use the TMVs obtained from the conventional block-matching technique [13]. With the mesh nodes, the transformed frame is generated as the prediction frame by affine transform and interpolation. Since three nodes are sufficient to solve all the affine transform parameters in a six-parameter model, the quadrilateral regions further split into triangular regions. Finally, the MVs of all mesh nodes will be encoded into the bitstream with lossless compression algorithms.

Affine and mesh methods show the possibility of making use of higher order models for improving DCP, but they have two major drawbacks limiting their practical applications. First, they both require much higher computational complexity and memory requirement for estimating the transformation parameters and storing additional transformed reference frames. Their complex encoder structures make them difficult to be integrated into modern video coding standards such as H.264/AVC. Second, coding efficiency improvements are limited by additional and complex parameters coding representations. AMMCP requires six to eight parameters to represent the affine transformations and mesh method requires a MV for each node. To tackle this problem, one potential direction is to use only a few popular deformations among interview frames in DCP, thus higher coding efficiency can be achieved for practical applications.

In this paper, horizontal scaling and shearing (HSS) effects among interview pictures of stereo video are studied. They are common deformations for objects with gradual change of depths. This approach was not very attractive in the past since it usually requires interpolation operation to obtain the deformed blocks or frames. Recently, a subsampled block-matching technique [18] demonstrated an approximation of zoom motion-compensated prediction in a low complexity way. By further generalizing the subsampled block matching technique, various types of deformations can be achieved by specially designed subsampling grid. In this paper, HSS-based DCP (HSS-DCP) realized by subsampled block matching is proposed.

This paper is organized as follows. A brief review of techniques for MVC tools is provided in Section II. Section III presents the study of HSS effects as common deformations among interview frames for exploiting the interview correlation. In the same section, HSS-DCP using subsampled block-matching technique is proposed as a practical higher order approach to obtain coding gain on exploiting interview dependency. Section IV describes the integration of the HSS-DCP into an H.264/AVC encoder and experimental results. This paper concludes with Section V.

II. Stereo and MVC Tools

To support stereo and multiview video coding, new techniques have been developed and adopted in current video coding standards such as H.264/AVC. In this section, some special techniques for supporting MVC are reviewed, which are mainly used for supporting the presentation of the proposed new HSS-DCP.

A. MVC Extension of H.264/AVC

H.264/AVC has MVC extension supporting large number of views with arbitrary camera positions. Stereo video is supported by using two views assuming two horizontally positioned cameras. Hierarchical coding is used to form an efficient prediction structure for stereo and multiview video. In stereo case, I frame is available only in the left view, which provide interview prediction reference to the right view and its prediction structure is shown in Fig. 2(a). In the MVC case, all frames in B view can be predicted by biprediction such that the bit rate can be further reduced as shown in Fig. 2(b). P frames are normally predicted by interview prediction only such that the P frame can be decoded along the views to prevent the decoding delay. Interview prediction is achieved by rearranging the encoding order such that the frames from different views can be referenced efficiently. To improve the coding efficiency, some methods are proposed [19]–[21] without modifying the bitstream format. They usually make use of the interview correlation for the disparity estimation (DE) such that the disparity vectors (DVs) can be encoded more efficiently.

B. Block-Based DCP

In stereo and MVC, the frames capture the same scene at the same time with different camera locations. The correlation

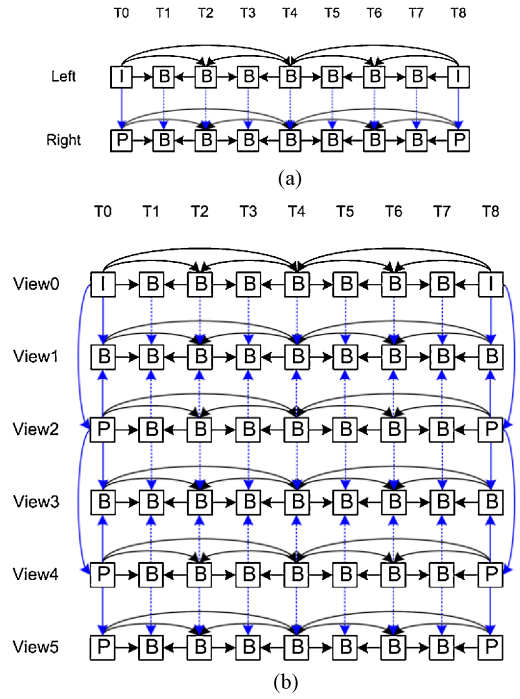


Fig. 2. Prediction structures. (a) Stereo video. (b) MVC with six views. View 0 is the base view, views 2, 4, and 5 are P views, and views 1 and 3 are B views.

between views is very similar to single view video sequence with camera panning effect. The difference between views depends on disparity effects. If the disparity information can be exploited such as motions in MCP, the coding efficiency can be improved. H.264/AVC MVC extension handles DCP using the same set of coding tools for single view encoding. In MCP, the current nth frame $\mathbf{F}_n$ of size $W \times H$ with pixel value $F_n(\mathbf{s})$ for each pixel location $\mathbf{s} = (x, y)$, where $0 \leq x < W$ and $0 \leq y < H$ is divided into K nonoverlapping blocks, $\{B_{i,n}\}_{i=1}^{K}$. Motion estimation is applied to find a prediction for each block based on the pixel value $\tilde{F}_m(\mathbf{s})$ in the mth reconstructed reference frame $\tilde{\mathbf{F}}_m$. Pixel value at noninteger location is obtained by interpolation. A residue block is created by subtracting the prediction from the current block. Only the residue block and the MV are required to be encoded. The reference frame number m and MV, $\mathbf{v}_{i,n} = (u, v)_{i,n}$, specifying a spatial displacement for motion compensation of ith block in $\mathbf{F}_n$, is determined by the block-matching process as

$$\left(m, \mathbf{v}_{i,n}\right) = \arg\min_{m,v} \mathrm{BDM}_{B_{i,n}}\left(F_n(\mathbf{s}), \tilde{F}_m\left(\mathbf{s} - \mathbf{v}\right)\right) \quad (1)$$

where block distortion metric (BDM) measures the difference between two blocks. A commonly used BDM is sum of absolute difference (SAD), which is defined as

$$\mathrm{SAD}_B\left(F(\mathbf{s}), G(\mathbf{s})\right) = \sum_{\mathbf{s}\in B} \left|F(\mathbf{s}) - G(\mathbf{s})\right|. \quad (2)$$

In DCP, the reconstructed reference frame $\tilde{\mathbf{G}}_m$ from the frame $\mathbf{G}_m$ of other view is used instead of previous frames from the same view. Thus, the DE becomes the problem of finding the reference frame and DV $\mathbf{d}_{i,n} = (u, v)_{i,n}$, which can be defined as

$$\left(m, \mathbf{d}_{i,n}\right) = \arg\min_{m,\mathbf{d}} \mathrm{BDM}_{B_{i,n}}\left(F_n(\mathbf{s}), \tilde{G}_m\left(\mathbf{s} - \mathbf{d}\right)\right) \quad (3)$$

where $\tilde{G}_m(\mathbf{s})$ is the pixel value of the frame $\tilde{\mathbf{G}}_m$ and $\mathbf{d}_{i,n}$ specifies a disparity for DCP of ith block in $\mathbf{G}_n$. Since the MVC extension supports more than two views, the frame selection process is still needed. Practically, there is no additional parameter in the encoded bitstream. The reference frame parameter indicates the interview frame and the MV parameter holds the DV, and the reference frame from the other view is put into the same buffer as the temporal reference frames. The conventional DCP is based on block matching that assumes a translation motion model in which the DVs of all pixels within a block are the same. As shown in Fig. 1, the projected shapes of the objects in two views may be deformed due to different depths within the objects. The prediction accuracy of conventional DCP is therefore highly degraded in these regions. In the following section, we will demonstrate that HSS effects are very common deformations among stereo pictures and show how to effectively realize the HSS-based DCP using subsampling block-matching technique.

III. HSS-Based Interview Prediction

As mentioned in Section I, the coding efficiency improvements of conventional higher order model-based DCPs are limited by their complex parameters coding representations. The higher order models should be simplified to exclude unnecessary parts and reduce the overhead for indicating different transforms. If DCP can focus on only a few popular deformations among interview frames, then higher coding efficiency could be achieved for practical MVC applications. Let us use stereovision to reveal some common deformations.

A. HSS Effects

In stereovision system, two eyes are horizontally displaced by a few centimeters from each other. As the viewing angle to an object from each eye is different, the projected images will be different. Three-dimensional reconstruction from multiview pictures depends on matching the correspondent points between the views and estimating the depth of the correspondent points. Fig. 3 shows a simple pinhole camera model-based two-view geometry, where *PSRQ* is a flat surface object to be observed and this surface is parallel to the projection image planes. C_L and C_R are the centers of projection (optical centers) and f is the common focal length. The line perpendicular to the projection plane passing through the optical center is the optical axis. $P_L S_L R_L Q_L$ and $P_R S_R R_R Q_R$ are the projected surfaces of the object *PSRQ* on the projection planes for left and right views, respectively. The displacement difference between the projected locations of P_L and P_R from the optical axis of each view is known as disparity and it depends on the depth Z. To study different types of deformation in stereovision, let us consider the stereo image pair as shown in Fig. 1 and focus on three regions: 1) the door on the left side; 2) the wall on the right side; and 3) the ceiling on the top.

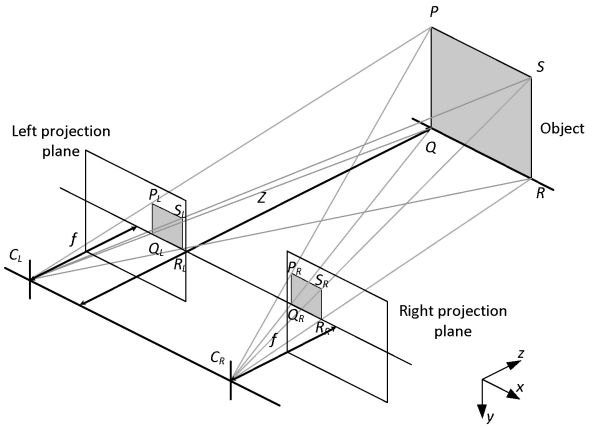


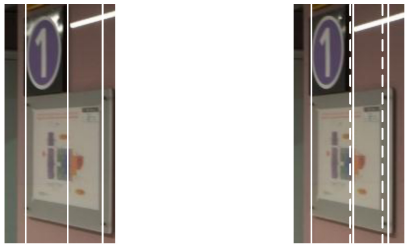
Fig. 3. Pinhole camera model-based two-view geometry.

B. Translational Effect

An object with flat surface parallel to the projection plane, such as the surface *PSRQ* as shown in Fig. 3, will appear on the two views with only translational displacement difference with same shape. This is similar to the door's surface on the left side of Fig. 1. It is because both the depth Z on the flat surface and the disparity is constant. The projected objects on these two views have equal size with same shape but in different locations. Thus, the conventional block-based DCP perfectly fits this situation when neglecting the occlusion problem. In real situations, many object surfaces may not be exactly parallel with the projection plane, which will create different types of deformations.

C. Horizontal Scaling Effect

Let us consider the region of the wall on the right side of Fig. 1. In addition, Fig. 4 shows the magnified parts of the wall. We can easily discover that the wall appears to be horizontally stretched in the left-view image as compared with the right-view image. Or we can consider the wall is horizontally compressed in the right-view image as compared with the left-view image. In this paper, both horizontal stretching and compressing deformations are considered as horizontal scaling deformation. To verify this phenomenon, the solid lines in the left-view image are marked some reference points and these lines are copied to the right-view image. At the right-view image, it can be seen that the solid lines do not matched the same point as in the left-view image. To match the corresponding points in the right-view image, dotted lines are added onto the right-view image of Fig. 4. While comparing the region bounded by these lines, it is obvious that the corresponding regions are horizontally compressed. These horizontal stretching and compression effects are mainly due to the gradually increasing or decreasing depth of a vertical surface as shown in Fig. 5. The surface of *PSRQ* as shown in Fig. 5 is not parallel to the projection plane but perpendicular to the zx-plane. As a result, the projected objects on these two views are unequal along the x-axis, which appeared as horizontal scaling deformations between the two views. This type of horizontal scaling deformation is very common in certain contents of real-world stereo images such as walls and standing objects.



(a) (b)

Fig. 4. Magnified regions on the wall with horizontal scaling deformation between (a) left and (b) right views.

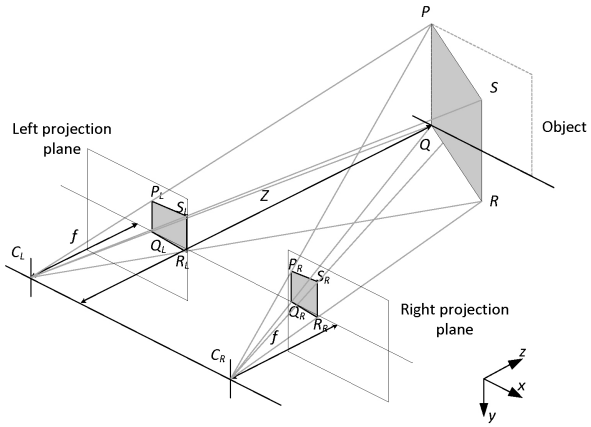
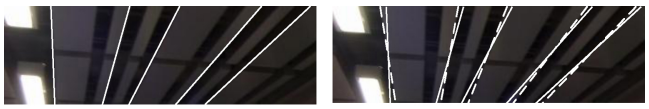


Fig. 5. Horizontal scaling deformation between two views caused by a vertical flat surface.



(a) (b)

Fig. 6. Magnification on the ceiling with shearing deformation between (a) left and (b) right views.

D. Horizontal Shearing Effect

Another common situation is a horizontal surface similar to the ceiling regions as shown in Fig. 1. Fig. 6 shows the magnified parts of this region, which appear to be horizontally sheared between two views. To verify such shearing effect, solid lines are added on the left-view image according to the texture of the ceiling and these lines are copied to the right-view image. Dotted lines are drawn according the texture of the ceiling in the right-view image. Comparing the regions bounded by these lines again, it can be seen that the corresponding regions in the right-view image are sheared left by the angle between the lines. This is caused by a horizontal surface parallel to zx-plane and perpendicular to the projection plane such as the PSQR surface in Fig. 7. As a result, horizontal shearing deformation occurs between the projected objects on these two views as shown in Figs. 6 and 7, which are common for regions of ceiling and floor in real-world stereo images.

Based on these observations, it can be deduced that the HSS effect is very common among stereo and multiview images. Although the actual disparity effects may not be completely

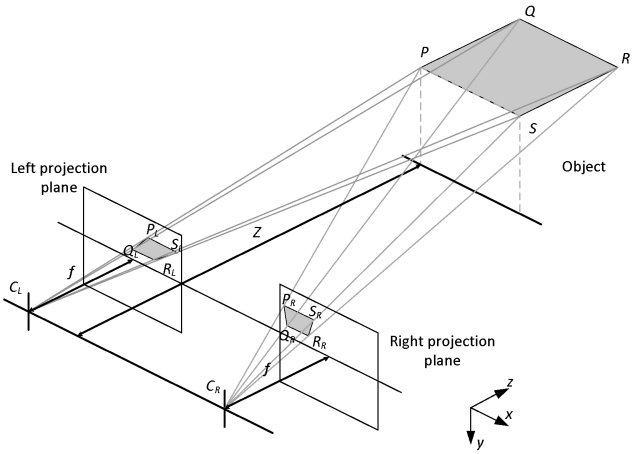


Fig. 7. Horizontal shearing deformation between the two views caused by a horizontal flat surface.

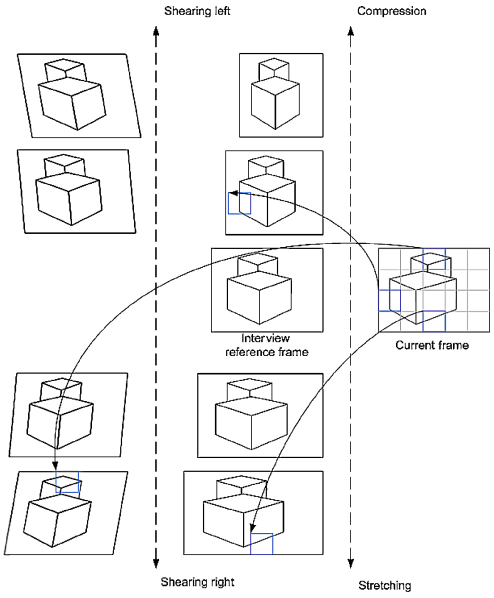


Fig. 8. Frame-based HSS DE.

characterized by using only HSS, they still provide good approximations for different deformations especially in the block-matching situation. It is possible to provide spatially transformed blocks with HSS effects to better model the disparity effects and extend the motion model to improve the coding efficiency of MVC. Fig. 8 shows an extended model to provide HSS frames in DCP where the transformed frames are derived from the interview reference frame.

E. Subsampled Block Matching for HSS-DCP

Although HSS effects can be achieved by applying affine transforms or by providing reference frames with HSS effects as shown in Fig. 8, the computational complexity and the memory requirement are significant. An efficient approach is to use subsampled block matching [18] to provide HSS transformed blocks. The interpolated frame for subpixel MCP is subsampled with various subsampling rates to obtain candidate prediction blocks with different HSS effects. It does not require additional operation to obtain a transformed block nor additional memory space for storing transformed blocks or frames. With subsampled block-matching technique, the DE in (3) can be modified as

$$\left(h, m, \mathbf{d}_{i,n}\right) = \arg\min_{h,m,\mathbf{d}} \mathrm{BDM}_{B_{i,n}}\left(F_n(\mathbf{s}), \tilde{G}_m\left(t_h\left(\mathbf{s}\right) - \mathbf{d}\right)\right) \quad (4)$$

where $t_h(\mathbf{s})$ is the transformed pixel position with deformation parameter h that indicates the transformations defined in (5)–(7). The detail of this parameter mapping with transformation parameters is provided in the implementation section. The parameter set may include only the common deformation effects in DCP. Different subsampling grids or patterns defined by $t_h(\mathbf{s})$ are used to achieve various affine transformation effects. Three types of transformation are considered to provide prediction candidates for DCP in this paper. To provide a zoomed in candidate block, the subsampling grid used is

$$w_0(\mathbf{s}) = \frac{\mathbf{s}}{\alpha} \quad (5)$$

where $\alpha \in \{4/1, 4/2, 4/3, 1, 4/5, \ldots\}$ is the zoom factors associated with the zoom levels for 1/4 subpixel reference frame. The zoom factors larger than 1 correspond to zoom-out effect and zoom factors smaller than 1 correspond to zoom-in effect. In HSS-DCP for interview prediction, the subsampling grids are asymmetric. Horizontal scaling has only the horizontal subsampling rate changed. The subsampling grid of horizontal scaling is defined as

$$w_1(\mathbf{s}) = \mathbf{s} \cdot \mathbf{H}_1, \qquad \text{where } \mathbf{H}_1 = \begin{bmatrix} 1/\beta & 0 \\ 0 & 1 \end{bmatrix} \quad (6)$$

where $\beta \in \{4/1, 4/2, 4/3, 1, 4/5, \ldots\}$ is the horizontal scaling factor associated with the horizontal scaling levels for 1/4 subpixel reference frame. For example, as shown in Fig. 9, when the original subsampling grid $\mathbf{s} \in \{(0,0), (4,0), (8,0), (12,0), (0,4), (4,4), (8,4), (12,4), \ldots, (12,12)\}$ of a block in subpixel resolution is substituted into (6) with $\beta = 4/3$, the subsampling gird $w_1(\mathbf{s})$ will become $\{(0,0), (3,0), (6,0), (9,0), (0,4), (3,4), (6,4), (9,4), \ldots, (9,12)\}$. As these pixels (triangular dot as shown in Fig. 9) exist in the interpolated reference frame, the subsampled block matching can be performed for horizontally scaled blocks. In this example with $\beta = 4/3$, a narrower region is subsampled in the reference frame to match with the current block, it is a horizontally stretched block for block matching. The horizontal scaling factor can be larger or smaller than 1. Horizontal scaling factors larger than 1 correspond to horizontal stretching deformation, while horizontal scaling factors smaller than 1 correspond to horizontal compression deformation. Fig. 10 shows an example with $\beta = 4/5$, a wider region is subsampled in the reference frame to match with the current block such that it is a horizontally compressed block for block matching.

Similarly, horizontal shearing can be obtained by the following transformation:

$$w_2(\mathbf{s}) = \mathbf{s} \cdot \mathbf{H}_2, \qquad \text{where } \mathbf{H}_2 = \begin{bmatrix} 1 & \gamma \\ 0 & 1 \end{bmatrix} \quad (7)$$

where $\gamma \in \{0, \pm 1/4, \pm 2/4, \pm 3/4, \ldots\}$ is the shearing factor that shifts the x coordinate depending on y for 1/4 subpixel reference frame. An example with

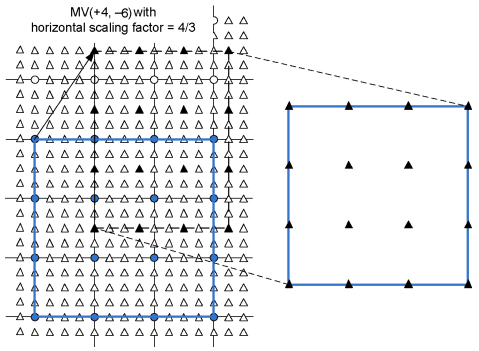


Fig. 9. Block matching on a reference frame of horizontal scaling factor of 4/3 (stretching).

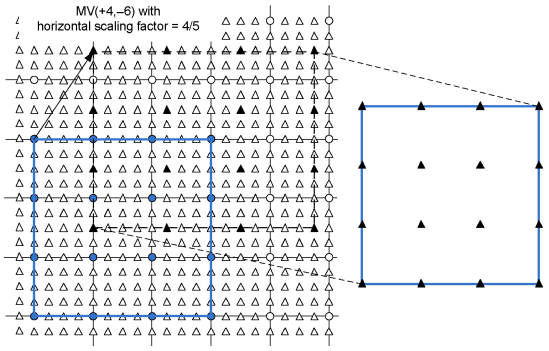


Fig. 10. Block matching on a reference frame of horizontal scaling factor of 4/5 (compression).

TABLE I
HSS PARAMETERS

Reference Frame Resolution	Scaling Factors (β)	Shearing Factors (γ)
1/4 subpixel	4, 2, 4/3, 1, 4/5, 4/6, 4/7, ..., 4/17	$-2, -7/4, \ldots, -1/4$, 0, 1/4, ..., 7/4, 2
1/8 subpixel	8, 4, 8/3, 2, 8/5, 8/6, 8/7, 1, 8/9, ..., 8/17	$-1, -7/8, \ldots, -1/8$, 0, 1/8, ..., 7/8, 1
1/16 subpixel	2, 16/9, 16/10, ..., 16/15, 1, 16/17, ..., 16/22, 16/23, 16/24	$-1/2, -7/16, \ldots,$ $-1/16$, 0, 1/16, ..., 7/16, 1/2

shearing factor $\gamma = 1/4$ is shown in Fig. 11. By substituting the coordinates (x, y) of the original subsampling grid $\mathbf{s} \in \{(0,0), (4,0), (8,0), (12,0), (0,4), (4,4), (8,4), (12,4), \ldots, (12,12)\}$ of a block in subpixel resolution into (7), the subsampling gird $w_2(s)$ will become $\{(0,0), (4,0), (8,0), (12,0), (1,4), (5,4), (9,4), (13,4), \ldots, (15,12)\}$. The parallelogramic region in the reference frame is subsampled for horizontally sheared block matching. The shearing factors can be positive or negative. Positive shearing factors correspond to right shearing deformation and negative shearing factors correspond to left shearing deformation. Thus, the example in Fig. 11 is a right shearing deformation. With these subsampling patterns, HSS effects can be achieved without extra computation since the pixels exist in the interpolated reference frames for subpixel MCP and DCP.

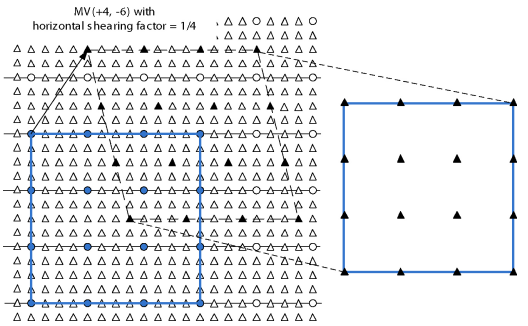


Fig. 11. Block matching on a reference frame of horizontal shearing factor of 1/4.

(a) (b)

Fig. 12. Example stereo image pair of the sequence *Akko_Kayo*. (a) Left-view image. (b) Right-view image.

F. Analysis on HSS-DCP

The prediction gain of HSS-DCP is investigated by applying HSS-DCP using subsampled block-matching technique to the sequences *Akko_Kayo*, *Rena*, *Balloons*, and *Kendo* from Nagoya University [25] with 640×480 or 1024×768 resolution. These sequences are generated by 1-D camera array from 7 to 100 cameras for MVC study. Fifty frames are used in this paper and two views at the middle of the camera array are selected as a stereo pair. Views 46 and 48 are selected for *Akko_Kayo*, views 38 and 40 are selected for *Rena*, views 3 and 5 are selected for *Balloons* and *Kendo*. Sample stereo image pairs are shown in Figs. 12–15. According to the camera parameters, the displacement between the left and the right cameras is about 5–10 cm, which is a popular setting for stereovision modeling human eyes. The HSS blocks are obtained by subsampled block-matching technique. The macroblock size is fixed at 16×16. The search windows size is set at ± 48 pixels and full search is used within the search window. SAD is used for the block distortion measure. The effects of using reference frames of different resolutions on HSS-DCP are investigated by using 1/4, 1/8, and 1/16 subpixel reference frames (interpolated reference frames for subpixel motion estimation). The 1/4 subpixel reference frame is already available in H.264/AVC. The 1/8 and 1/16 subpixel reference frames are obtained by bilinear interpolation from the 1/4 subpixel reference frame. In each case, 17 levels of horizontal scaling and 17 levels of horizontal shearing, including the original reference frame without HSS effects, are investigated. The HSS transformation parameters are listed in Table I, where $\beta = 1$ and $\gamma = 0$ are the original scale without HSS effect. The purpose of this section is to study the possible prediction gain can be obtained from HSS-DCP without considering encoding overheads.

(a) (b)

Fig. 13. Example stereo image pair of the sequence *Rena*. (a) Left-view image. (b) Right-view image.

(a) (b)

Fig. 14. Example stereo image pair of the sequence *Balloons*. (a) Left-view image. (b) Right-view image.

(a) (b)

Fig. 15. Example stereo image pair of the sequence *Kendo*. (a) Left-view image. (b) Right-view image.

The prediction gain of HSS-DCP is given in Fig. 16. The improvement is based on the comparison with translation-only DCP. Improvement is observed in all cases. The gain is between 0.83 dB and 1.46 dB. The maximum improvement is achieved for *Akko_Kayo* sequence using 1/8 subpixel reference frame. The prediction gain of HSS-DCP has a significant drop from the maximum when using 1/16 subpixel reference frame. This drop may be due to the range of HSS scales. As shown in Table I, the range of horizontal stretching scales using 1/16 subpixel reference frame is much narrower than the other two cases with the same number of HSS levels. Thus, some HSS effects in the picture may not be covered using the set of scale factors used in 1/16 subpixel reference frame case.

As the computation loading increases with the number of HSS scales used, the number of scaling and shearing levels used in practical implementation should be determined. Figs. 17 and 18 show the prediction gains of HSS. Table II lists the numerical results. The prediction gain of horizontal scaling is up to 1.10 dB for various sequences. As expected, the prediction gain increases with the number of levels. Although shearing provides less gain as compared to scaling, it can provide addition gain when both HSS are used. However, the incremental gain becomes insignificant when the number of level increases from 9 to 17 in both horizontal scaling and

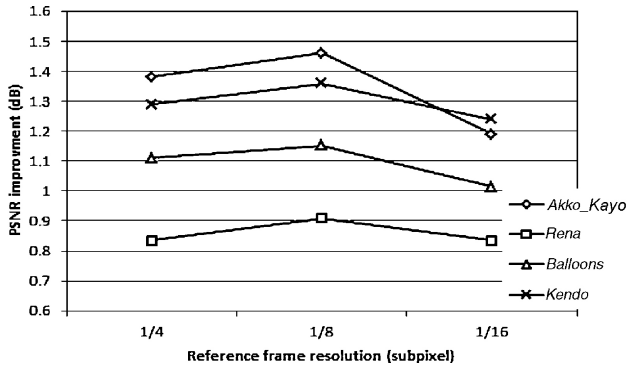


Fig. 16. PSNR improvement of HSS-DCP with reference frame at various resolutions.

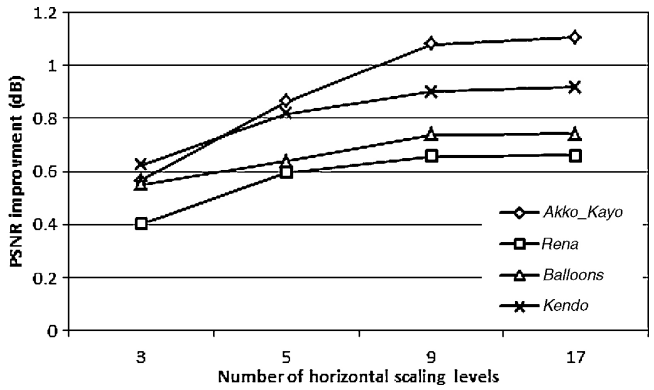


Fig. 17. PSNR improvement of various numbers of stretching levels with 1/4 subpixel reference frame.

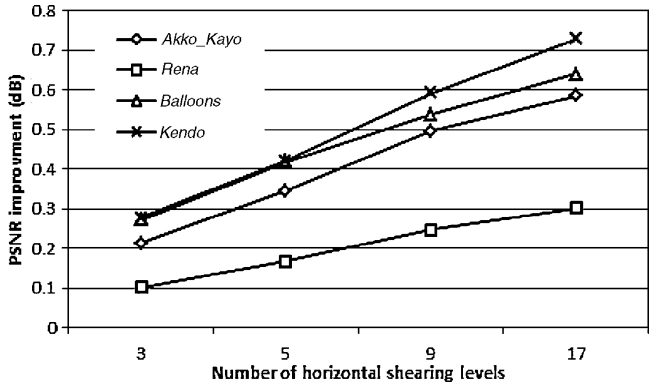


Fig. 18. PSNR improvement of various numbers of shearing levels with 1/4 subpixel reference frame.

horizontal shearing cases for all sequences under investigation. The result of HSS with nine horizontal scaling and nine horizontal shearing levels is also provided. The result shows that the improvement is very close to 17 levels and the difference of prediction gain is less than 0.1 dB. Thus, nine levels are selected for both HSS in this paper. From these results, it can be seen that the prediction gain given by HSS-DCP is quite significant. In addition, in order to demonstrate the effects of HSS-DCP based on our assumed contents, Fig. 19(a) and (b) shows a residue image comparison between translation-only DCP and HSS-DCP with 17 horizontal shearing and 17 horizontal scaling levels using the example stereo image as

Fig. 19. Residue image comparison of predicted right view from Fig. 1. (a) Using translation-only-based DCP. (b) Using HSS-based DCP. The error has been amplified for better visibility. (c) Block-wise selection of (b) indicating translation-only DCP (by black blocks) and HSS-based DCP (by white blocks).

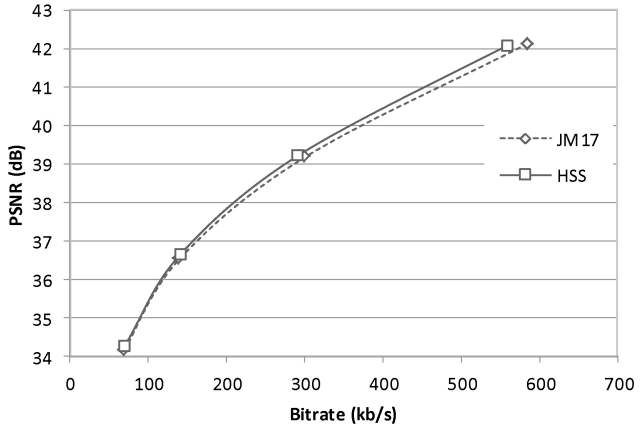


Fig. 20. RD curve comparison of the sequence *Rena*.

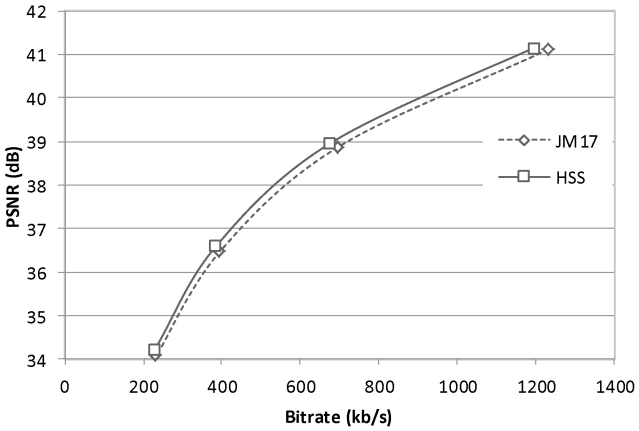


Fig. 22. RD curve comparison of the sequence *Balloons*.

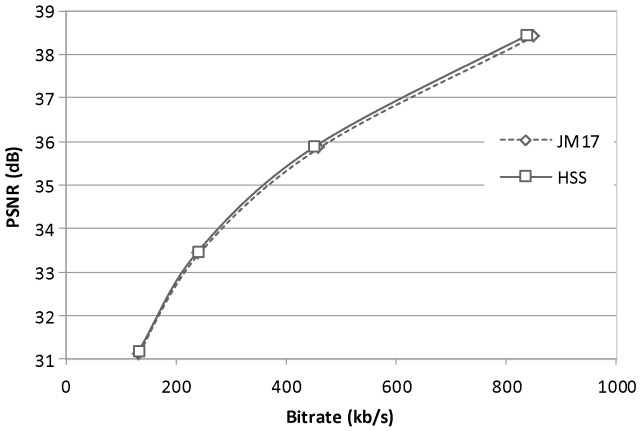


Fig. 21. RD curve comparison of the sequence *Akko_Kayo*.

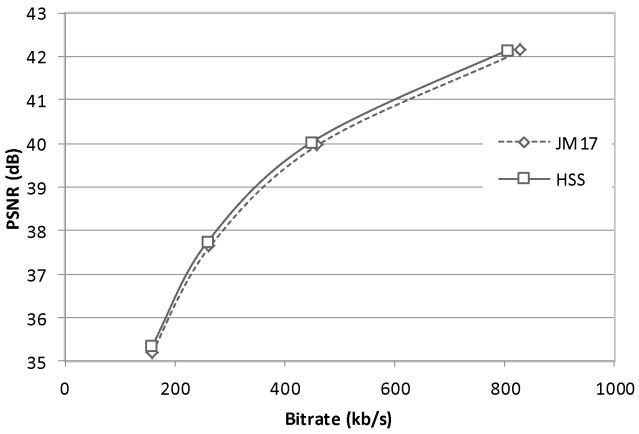


Fig. 23. RD curve comparison of the sequence *Kendo*.

shown in Fig. 1. It shows that, with HSS-DCP, the prediction errors for the regions of the ceiling and the wall are significantly lower. Fig. 19(c) shows the block-wise selection of translation-only DCP (by black blocks) and HSS-based DCP (by white blocks), which indicates that HSS-DCP is highly used to provide the prediction improvement. Thus, HSS can better model the deformation among views for these regions.

IV. Experimental Results

HSS-DCP using subsampled block-matching technique is implemented into JM version 17.0 [26] (reference software of H.264/AVC with support of MVC) to evaluate the performance. The implementation details, experimental settings, and results are presented in this section.

A. Integration of HSS-DCP in Practical Video Codecs

To integrate HSS-DCP into modern video codec such as H.264/AVC, the cost of DV in determining the prediction should also be considered. The DVs constituting the spatial displacement, frame reference, and HSS parameter are required to be coded and then transmitted to the decoder. In H.264/AVC, rate-distortion (RD)-optimized motion search is

TABLE II
SIMULATION RESULT OF HSS-DCP WITH VARIOUS HSS LEVELS WITH 1/4 SUBPIXEL REFERENCE FRAME

Sequence	DCP	Horizontal Scaling				Horizontal Shearing				HSS-DCP	
	PSNR (dB)	3	5	9	17	3	5	9	17	17+17	9+9
Rena	37.57	37.97	38.17	38.23	38.23	37.67	37.74	37.82	37.87	38.41	38.37
Improvement (dB)		0.40	0.60	0.66	0.66	0.10	0.17	0.25	0.30	0.83	0.80
Akko_Kayo	31.20	31.77	32.07	32.28	32.31	31.42	31.55	31.70	31.79	32.58	32.52
Improvement (dB)		0.57	0.86	1.08	1.10	0.21	0.34	0.50	0.58	1.38	1.32
Balloons	33.55	34.10	34.19	34.29	34.29	33.82	33.97	34.09	34.19	34.66	34.60
Improvement (dB)		0.55	0.64	0.74	0.74	0.27	0.42	0.54	0.64	1.11	1.05
Kendo	36.54	37.17	37.36	37.44	37.46	36.82	36.96	37.13	37.27	37.83	37.74
Improvement (dB)		0.62	0.82	0.90	0.92	0.28	0.42	0.59	0.73	1.29	1.20

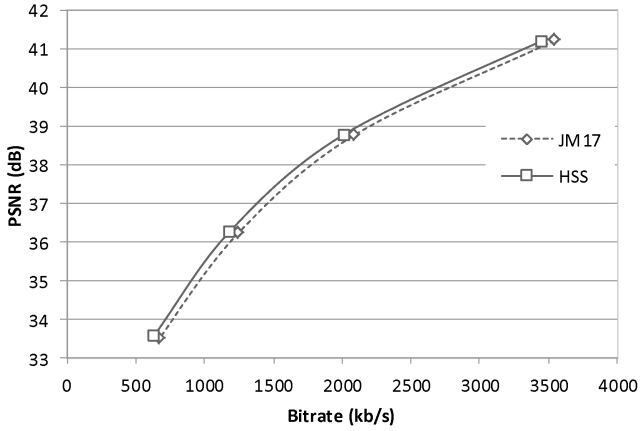


Fig. 24. RD curve comparison of the sequence *Champagne*.

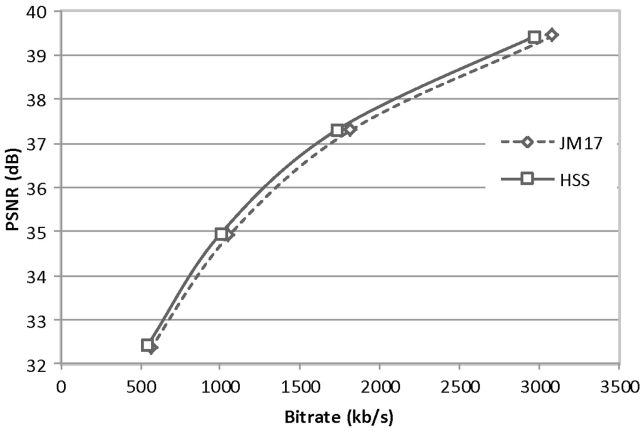


Fig. 25. RD curve comparison of the sequence *Dog*.

implemented by minimization of the Lagrangian cost function. Thus, (4) is modified as

$$\left(h, m, \mathbf{d}_{i,n}\right) = \arg\min_{h,m,\mathbf{d}} \left(\mathrm{BDM}_{B_{i,n}}\left(F_n(\mathbf{s}), \tilde{G}_m\left(t_h\left(\mathbf{s}\right) - \mathbf{d}\right)\right) + \lambda_{\mathrm{motion}} R\left(h, m, \mathbf{d}\right)\right) \quad (8)$$

where $\lambda_{\mathrm{motion}}$ is the Lagrangian multiplier for motion search used in H.264/AVC and $R(h, m, \mathbf{d})$ is the bitrate associated with a particular choice of DV. The DV $(h, m, \mathbf{d})$ for each block mode (intermodes of various subblock sizes) is the one that gives minimum Lagrangian cost.

The HSS effects are achieved by subsampled block matching with the DCP reference frame. Although the nature and generation of HSS effects are quite different from conventional DCP and MCP, HSS-DCP can be considered as providing additional reference frames for prediction. Thus, by considering the additional HSS frames as an increase of reference frame buffer, HSS parameter h is used to represent both β and γ with predefined values and this parameter is embedded into the reference frame parameter provided by H.264/AVC bitstream.

In this paper, we only consider to apply HSS-DCP on the P frames in stereo-prediction structure as shown in Fig. 2(a). Therefore, the parameter of m in (8) is always equal to 0, which corresponds to the only interview reference frame (I frame). In practical implementation of (8) in reference software, both of the parameters h and m are embedded into the reference frame number m' of the reference software as shown in Table III. For $m' = 0$, it is referring to the interview prediction without HSS effect such that both m and h are equal to 0. The reference frame numbers m' between 1 and 15 with $m = m'$ and $h = 0$ are reserved for other interview reference frames without HSS effect. For the m' between 16 and 31, they are corresponding to HSS-DCP with nonzero h parameters but $m = 0$. The mapping of h parameter with predefined transformation parameters β and γ are embedded into reference frame number m', which making the horizontal scales and shearing factors can be determined by table lookup as shown in Table III. According to Section III-C, scaling can provide more improvement than shearing. Further improvement can be achieved when both scaling and shearing are used. The additional improvement of using both scaling and shearing deformations is larger than the improvement due to each type of deformation. In Table III, therefore, we allocate the reference frame numbers by interlacing scaling and shearing levels and begin with two scaling levels and then two shearing levels for better entropy encoding performance. This method of encoding HSS-DCP parameters can be extended to multiple interview reference frames for MVC. In addition, the second interview reference frame of HSS-DCP with $m = 1$ with nonzero h parameters can be mapped to the range of m' between 32 and 47 similar to the arrangement in Table III.

Although Fig. 16 show that 1/8 subpixel reference frame provides the highest prediction gain for HSS-DCP, the best prediction gain is very close to that using 1/4 subpixel reference frame with a difference of less than 0.08 dB. With considerations of the computational complexity and integration on practical video codec such as H.264/AVC reference software, 1/4 subpixel reference frames is adopted. It makes the direct

TABLE III
DEFINITION OF HSS PARAMETERS AND ASSOCIATED VALUES

Reference frame number (m')	0-15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Interview frame number (m)	0-15	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
HSS parameter (h)	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
HSS Effect	None	Scaling		Shearing		Scaling		Shearing		Scaling		Shearing		Scaling		Shearing	
Transformation function of $t_h(\mathbf{s})$	$\mathbf{s}$	$w_1(\mathbf{s})$		$w_2(\mathbf{s})$		$w_1(\mathbf{s})$		$w_2(\mathbf{s})$		$w_1(\mathbf{s})$		$w_2(\mathbf{s})$		$w_1(\mathbf{s})$		$w_2(\mathbf{s})$	
Horizontal scaling factor (β)		4/3	4/5			2	4/6			4	4/7			4/8	4/9		
Horizontal shearing factor (γ)				$\frac{1}{4}$	$-\frac{1}{4}$			$\frac{2}{4}$	$-\frac{2}{4}$			$\frac{3}{4}$	$-\frac{3}{4}$			1	-1

implementation of HSS-DCP in H.264/AVC, which supports 1/4 pixel MV, possible without significant modification.

The main cost of subsampled block-matching technique is the larger pool for prediction searching. With different subsampling grid, HSS blocks can be obtained from the subpixel reference frames without computation. The extra computation is mainly due to the block distortion measure for the candidate blocks. From Section III-C, it is found that using nine horizontal scaling and nine horizontal shearing levels can achieve prediction improvement close to that of 17 levels. Thus, nine horizontal scaling and nine horizontal shearing levels can be considered as an effective range of HSS and are used in this implementation to minimize the computational requirement and overhead.

In the case of nine horizontal scaling and nine horizontal shearing levels, total 17 reference frames are involved. Intuitively, the computation of block matching will grow to 17 times and it will be a heavy loading if full search DE is used. However, in DCP case, the computation due to the additional HSS reference frames is not likely to be that significant. The reason is that, due to the nature of multiview video, the relative locations between cameras are usually known such that the information can be exploited and the computation can be reduced substantially. For example, in stereo video, if the cameras are correctly calibrated with horizontal displacement without rotation, it is possible to use horizontal DV for DCP without significant performance impact. Horizontal block DE or other fast algorithms can be applied to find the DV such that the computation can be dramatically reduced. For example, without fast block-matching algorithms, with a ± 64-pixel searching range, the difference of computation loading between 2-D and 1-D search is 129 times. In such case, the computation for searching 17 HSS frames is relatively insignificant. In our experiments, effect of using horizontal DV search is also investigated. On the other hand, the decoder complexity of HSS-DCP is very close to without HSS-DCP, as the numbers of pixels to be calculated for a prediction block in the decoder are just the same.

B. Experiment Setup

The proposed HSS-DCP is implemented on JM version 17.0 [26], which is the first version of JM with MVC support. HSS-DCP is applied on large block modes (16×16, 16×8, and 8×16) of P frames only. In the experiments, nine horizontal scaling and nine horizontal shearing levels are used in the DE. It means that total 17 reference frames are involved. Apart from the four sequences used in the previous section, two higher resolution (1280×960) sequences, namely, *Champagne* and *Dog* are also used. Each sequence has many views and two of the views are taken as a stereo pair. Views 39 and 41 are selected for *Champagne* sequence; views 37 and 39 are selected for *Dog* sequence. The first 100 frames from each view will be used. The H.264/AVC coding tools such as VBS and rate-distortion optimization (RDO) are turned on. Search window is set at ± 64 and exhaustive search is used within the search window. The left view is used as the base view and the right view is the alternate view predicted by interview prediction or inter prediction. Due to the special coding structure of stereo video coding, P frames in the right view use interview prediction only and B frames use inter prediction only. GOP structures without B frames and with seven hierarchical B frames are tested. Quantization parameter (QP) factors 28, 32, 36, and 40 are used for RD performance evaluation. The average bitrate reduction and average peak signal-to-noise ratio (PSNR) improvement are calculated using Bjøntegaard delta (BD)-bitrate and BD-PSNR [24].

C. Direct Improvement of HSS Interview Prediction

To investigate the direct improvement, GOP structure IIII is used for base view and PPPP is used for the alternate view. Since the P frames only use interview prediction, performance of HSS-DCP and conventional block-matching method can be compared directly. Table IV shows the RD performance comparison of the alternate view from each sequence and Figs. 20–25 provide RD curves comparison of each sequence. The improvement is quite significant and the BD-bitrate reduction is around 2.33%–5.25% and the BD-PSNR improvement is around 0.09–0.23 dB. In addition, the overhead bits for indicating HSS parameter is about 0.24%–1.79%, which is relatively small. Furthermore, in HSS-DCP, the mode selection distribution has more inter prediction modes instead of skip mode and intra modes. As in RDO, mode is selected based on the Lagrangian function. While the translation-only prediction does not provide accurate prediction, the residue coding cost might be even higher than skip mode or intra modes. Table V shows the comparisons of mode distribution for QP of 28 and 40. For all sequences with different QP, there are significant increases in the percentage of inter 16×16, 16×8, and

TABLE IV

RD COMPARISON OF INTERVIEW PREDICTION BETWEEN JM17 AND HSS-DCP

Akko_Kayo	JM17 Full Search		HSS Full Search			*Rena*	JM17 Full Search		HSS Full Search		
QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)	HSS Overhead	QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)	HSS Overhead
28	850.95	38.41	837.99	38.43	0.26%	28	584.24	42.12	559.28	42.08	0.39%
32	459.75	35.84	452.55	35.88	0.45%	32	300.29	39.18	290.98	39.21	0.69%
36	241.92	33.41	240.23	33.47	0.68%	36	139.68	36.53	140.94	36.65	1.15%
40	132.43	31.1	132.07	31.17	1.08%	40	69.15	34.15	69.47	34.28	1.71%
		BD-PSNR (dB)			0.0919			BD-PSNR (dB)			0.1254
		BD-bitrate (%)			−2.3222			BD-bitrate (%)			−3.3744
Balloons	JM17 Full Search		HSS Full Search			*Kendo*	JM17 Full Search		HSS Full Search		
QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)	HSS Overhead	QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)	HSS Overhead
28	1232.23	41.11	1194.48	41.15	0.57%	28	830.03	42.12	806.17	42.15	0.59%
32	696.1	38.87	674.32	38.95	0.85%	32	461.03	39.97	450.25	40.04	0.92%
36	395.29	36.47	384.86	36.58	1.20%	36	263.72	37.62	259.74	37.74	1.27%
40	233.45	34.06	228.93	34.21	1.69%	40	159.51	35.19	158.84	35.34	1.50%
		BD-PSNR (dB)			0.2133			BD-PSNR (dB)			0.1683
		BD-bitrate (%)			−4.9157			BD-bitrate (%)			−3.9417
Champagne	JM17 Full Search		HSS Full Search			*Dog*	JM17 Full Search		HSS Full Search		
QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)	HSS Overhead	QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)	HSS Overhead
28	3546.34	41.21	3447.04	41.19	0.24%	28	3087.19	39.43	2967.74	39.42	0.31%
32	2093.17	38.77	2011.1	38.77	0.41%	32	1821.93	37.3	1731.82	37.3	0.49%
36	1242.87	36.22	1184.43	36.27	0.67%	36	1054	34.91	1001.21	34.94	0.71%
40	671.82	33.5	633.62	33.58	1.11%	40	566.28	32.35	545.76	32.43	0.72%
		BD-PSNR (dB)			0.2309			BD-PSNR (dB)			0.2258
		BD-bitrate (%)			−4.9247			BD-bitrate (%)			−5.2466

TABLE V

MODE DISTRIBUTION COMPARISON BETWEEN JM17 AND HSS-DCP

	Rena				*Akko_Kayo*			
	QP = 28		QP = 40		QP = 28		QP = 40	
Block Mode	JM17	HSS	JM17	HSS	JM17	HSS	JM17	HSS
Skip	2800	2208	73 373	69 920	11 731	10 832	81 281	77 110
Inter 16 × 16	34 724	37 650	9747	14 642	60 357	58 274	17 835	22 158
Inter 16 × 8	4886	5738	960	1092	8767	9461	3417	3382
Inter 8 × 16	9422	12 617	1311	2166	17 047	21 977	4708	6339
Inter P8 × 8	1095	652	122	107	6044	4743	875	619
Intra 4 × 4	2805	2596	328	202	1947	1683	333	300
Intra 8 × 8	39 287	33 287	12 538	11 621	8989	8358	5402	4904
Intra 16 × 16	24 981	25 252	21 621	20 250	5118	4672	6149	5188
	Balloons				*Kendo*			
	QP = 28		QP = 40		QP = 28		QP = 40	
Block Mode	JM17	HSS	JM17	HSS	JM17	HSS	JM17	HSS
Skip	109 965	96 725	231 964	221 672	85 067	71 623	208 321	197 996
Inter 16 × 16	84 370	103 355	34 735	51 862	69 436	86 385	26 142	40 317
Inter 16 × 8	19 458	20 277	6011	5405	11 380	11 985	2701	3006
Inter 8 × 16	21 212	23 911	9266	10 563	15 382	18 346	4142	5317
Inter P8 × 8	6458	5020	1563	962	2515	1871	495	262
Intra 4 × 4	2221	1837	1102	845	4379	3826	804	671
Intra 8 × 8	23 339	19 103	9936	7551	31 174	27 843	17 545	19 051
Intra 16 × 16	40 177	36 972	12 623	8340	87 867	85 321	47 050	40 580
	Champagne				*Dog*			
	QP = 28		QP = 40		QP = 28		QP = 40	
Block Mode	JM17	HSS	JM17	HSS	JM17	HSS	JM17	HSS
Skip	153 550	150 691	343 986	335 997	53 938	47 819	228 231	210 422
Inter 16 × 16	76 546	91 734	52 942	72 675	138 129	150 230	45 478	64 171
Inter 16 × 8	22 910	19 229	11 457	9335	18 234	20 068	5256	8367
Inter 8 × 16	24 719	28 481	14 949	17 249	41 632	61 944	13 303	25 530
Inter P8 × 8	9067	6373	3217	1580	5526	3447	622	392
Intra 4 × 4	10 382	9798	1062	744	6899	5530	2877	3244
Intra 8 × 8	65 688	59 329	23 299	17 217	168 380	146 528	80 211	68 881
Intra 16 × 16	117 138	114 365	29 088	25 203	47 262	44 434	104 022	98 993

TABLE VI
COMPUTATIONAL COMPLEXITY ANALYSIS OF HSS-DCP

	JM17 Full Search		HSS Full Search		HSS Horizontal Full Search	
Sequence	Runtime (s)	DE Time (s)	Runtime (s)	DE Time (s)	Runtime (s)	DE Time (s)
Akko_Kayo	2400	1003	13 982	13 081	1629	730
Rena	2110	888	12 783	11 939	1587	745
Balloons	5208	1881	23 504	21 235	4056	1797
Kendo	4859	1691	21 060	18 867	3972	1790
Champagne	10 763	5419	58 041	54 464	6318	2933
Dog	13 264	7874	103 270	99 693	6572	3231

TABLE VII
RD PERFORMANCE COMPARISON OF HSS-DCP WITH FAST DE ALGORITHM COMPARED TO JM17

	HSS Full Search		HSS Horizontal Full Search	
Sequence	BD-PSNR (dB)	BD-Bitrate (%)	BD-PSNR (dB)	BD-Bitrate (%)
Akko_Kayo	0.0919	−2.3222	0.0643	−1.6380
Rena	0.1254	−3.3744	0.1321	−3.5124
Balloons	0.2133	−4.9157	0.1408	−3.2845
Kendo	0.1683	−3.9417	0.1271	−2.9987
Champagne	0.2309	−4.9247	0.1356	−2.8903
Dog	0.2258	−5.2466	0.1817	−4.1830

TABLE VIII
COMPARISON OF RD PERFORMANCE BETWEEN JM17 AND HSS-DCP WITH PRACTICAL GOP STRUCTURE

rena	JM17		HSS		*Akko_Kayo*	JM17		HSS	
QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)	QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)
28	99.21	42.62	95.42	42.61	28	138.1	38.82	135.44	38.83
32	57.81	40.01	55.46	40.03	32	80.92	36.42	79.58	36.45
36	32 37	37 55	31 86	37 66	36	47 77	34 13	47 29	34 2
40	17.03	35.25	17.59	35.43	40	27.05	31.87	27.11	31.95
	BD-bitrate (%)			−4.0534		BD-bitrate (%)			−2.4176
	BD-PSNR (dB)			0.175		BD-PSNR (dB)			0.1048
Balloons	JM17		HSS		*Kendo*	JM17		HSS	
QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)	QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)
28	210.78	41.57	204.22	41.6	28	146.53	42.51	143.32	42.52
32	127.18	39.51	123.44	39.58	32	89.17	40.6	86.7	40.65
36	77.32	37.25	75.37	37.36	36	54.63	38.48	53.69	38.6
40	47.64	34.88	46.8	35.04	40	33.74	36.24	34.01	36.4
	BD-bitrate (%)			−4.6198		BD-bitrate (%)			−3.7267
	BD-PSNR (dB)			0.2124		BD-PSNR (dB)			0.1616
Champagne	JM17		HSS		*Dog*	JM17		HSS	
QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)	QP	Bitrate (kb/s)	PSNR (dB)	Bitrate (kb/s)	PSNR (dB)
28	516.46	41.53	502.25	41.52	28	472.97	39.72	457.61	39.72
32	311.67	39.25	300.33	39.24	32	292.45	37.78	279.9	37.79
36	193.01	36.81	184.41	36.84	36	182.6	35.64	175.34	35.68
40	113.31	34.28	107.23	34.33	40	110.14	33.31	107.46	33.4
	BD-bitrate (%)			−4.3591		BD-bitrate (%)			−4.4252
	BD-PSNR (dB)			0.21		BD-PSNR (dB)			0.1997

8×16 modes being selected in HSS-DCP case. With low-quality configuration (QP = 40), the use of skip mode is largely reduced. With high-quality configuration (QP = 28), the use of intra modes is largely reduced. As HSS-DCP is only applied on these inter modes, it prevents a significant amount of intra and skip mode selection by providing better predictions.

D. Computation Complexity of HSS Interview Prediction

To investigate the computation complexity of HSS-DCP, a horizontal DV search is applied. For the integer pixel locations, the search range is limited to horizontal only. For subpixel locations, the search range is set at 3/4 pixel for both directions. All other configurations remain the same as previous section. Table VI shows the runtime comparison of the proposed HSS-DCP. Without fast DE algorithm, the time spend on DE is more than 10 times of translational DCP and the total encoding time is about six times longer. With the fast algorithm, the DE runtime is only about 41%–95% of full search without HSS-DCP and the total encoding time is also shortened. Since fast DE algorithms might have drawbacks of lower accuracy, RD analysis is also investigated for the fast DE algorithm. From Table VII, it can be seen that the impact

on the RD performance is small. The improvement is slightly reduced to 1.64%–4.13% of BD-bitrate reduction and 0.06–0.18 dB of BD-PSNR improvement, which is still significant for just using single technique for improving RD performance.

E. Overall Improvement of HSS Interview Prediction

In practical stereo video coding, biprediction using hierarchical B frames, as shown in Fig. 2(a), is commonly used. As biprediction can provide very good predictions, interview prediction is usually not applied to B frames. Thus, interview prediction using HSS-DCP is also not applied to B frames in this part as the improvement may not be significant. However, the HSS-DCP improvement on P frames can be propagated to B frames if the quality of reconstructed P frames is higher. In this part, the GOP structure as shown in Fig. 2(a) is used, i.e., seven hierarchical B frames between I and P frames. Table VIII shows the RD performance of the alternate view. Although the improvement is diluted in some of the sequences, the improvement of other sequences becomes larger. The overall performance is about 2.42%–4.62% of BD-bitrate reduction and 0.10–0.21 dB of BD-PSNR improvement. The improvement of PPPP coding structure is reasonably maintained so that HSS-DCP is suitable for practical use in stereo video coding. That should also be applied to MVC using the prediction structure as shown in Fig. 2(b) for applying HSS-DCP to all P frames.

F. Indirect Comparison With Full Affine Model DCP

Since there is no existing method that utilizes the full affine model for DCP yet, we can only compare HSS-DCP and full affine model indirectly. It is true that the affine model can adopt more deformations that can provide more accurate predicted blocks. However, the affine model usually associates with six to eight parameters. The overhead can be up to six to eight times of HSS-DCP and that would be difficult to encode them efficiently. For the computational cost, if the full affine model is applied to DCP similar to the AMMCP, there will be a process of affine parameter estimation to determine the parameter set to be used in DCP, which is very computationally complex and the memory requirement is higher. HSS-DCP does not have such process since the HSS effects are predefined and the transformation effects are achieved by subsampled block matching. The complexity of HSS-DCP is obviously much lower than that of the full affine model approach.

V. Conclusion

HSS deformations between the two views in stereo images are very common in vertical and horizontal flat surfaces that are not parallel to the projection plane. These deformations relation among the views can be utilized in DCP. In this paper, HSS deformations were used for improving interview prediction using subsampled block-matching technique. The interview prediction accuracy was improved by introducing additional HSS deformation-based prediction candidates blocks generated by specially selected sets of subsampled pixels from the interpolated interview reference frames. This HSS-DCP was implemented in JM reference software with MVC support for improving the compression efficiency. Experimental results showed that bitrate reduction up to 5.25% can be achieved in interview prediction for stereo video coding. The improvement on the hierarchical prediction structure was up to 4.62%. As HSS-DCP was performed based on the interpolated interview reference frames for subpixel DE, no significant extra storage was required in both encoder and decoder implementation. Thus, it was suitable for practical stereo video coding with the possibility of generalized MVC applications.

References

[1] *Introduction to Multiview Video Coding*, document N9580, ISO/IEC JTC 1/SC 29/WG 11, Antalya, Turkey, Jan. 2008.

[2] *Generic Coding of Moving Pictures and Associated Audio Information, Part 2: Video*, document ITU-T Rec. H.262 and ISO/IEC 13818-2 (MPEG-2), ITU-T and ISO/IEC JTC-1, 1995.

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

[4] T. Wiegand, X. Zhang, and B. Girod, "Long-term memory motion compensated prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 2, pp. 70–84, Feb. 1999.

[5] G. J. Sullivan and R. L. Baker, "Rate-distortion optimized motion compensation for video compressionusing fixed or variable size blocks," in *Proc. Global Telecommun. Conf.*, 1991, pp. 85–90.

[6] T. Wedi and H. G. Musmann, "Motion- and aliasing-compensated prediction for hybrid video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 577–586, Jul. 2003.

[7] H. Schwarz, D. Marpe, and T. Wiegand, "Analysis of hierarchical B pictures and MCTF," in *Proc. IEEE ICME*, Jul. 2006, pp. 1929–1932.

[8] P. Merkle, A. Smolic, K. Muller, and T. Wiegand, "Efficient prediction structures for multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1461–1473, Nov. 2007.

[9] M. Kitahara, H. Kimata, S. Shimizu, K. Kamikura, Y. Yashimata, K. Yamamoto, T. Yendo, T. Fujii, and M. Tanimoto, "Multi-view video coding using view interpolation and reference picture selection," presented at the IEEE International Conference Multimedia and Exposition (ICME), Toronto, ON, Canada, Jul. 2006.

[10] Y. J. Jeon, J. Lim, and B. M. Jeon, "Report of MVC performance under stereo condition," document JVT-AE016, Joint Video Team, London, U.K., Jun. 2009.

[11] T. Wiegand, E. Steinbach, and B. Girod, "Affine multipicture motion-compensated prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 15, no. 2, pp. 197–209, Feb. 2005.

[12] R. C. Kordasiewicz, M. D. Gallant, and S. Shirani, "Affine motion prediction based on translational motion vectors," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 10, pp. 1388–1394, Oct. 2007.

[13] R. S. Wang and Y. Wang, "Multiview video sequence analysis, compression, and virtual viewpoint synthesis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 10, no. 3, pp. 397–410, Apr. 2000.

[14] M. Sayed and W. Badawy, "An affine-based algorithm and SIMD architecture for video compression with low bit-rate applications," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 4, pp. 457–471, Apr. 2006.

[15] S. R. Han, T. Yamasaki, and K. Aizawa, "Time-varying mesh compression using an extended block matching algorithm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1506–1518, Nov. 2007.

[16] M. R. Pickering, M. R. Frater, and J. F. Arnold, "Enhanced motion compensation using elastic image registration," in *Proc. IEEE Int. Conf. Image Process.*, Oct. 2006, pp. 1061–1064.

[17] A. A. Muhit, M. R. Pickering, M. R. Frater, and J. F. Arnold, "Video coding using elastic motion model and larger blocks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 5, pp. 661–672, May 2010.

[18] L. M. Po, K. M. Wong, K. W. Cheung, and K. H. Ng, "Subsampled block-matching for zoom motion compensated prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1625–1637, Nov. 2010.

[19] T. Frajka and K. Zeger, "Residual image coding for stereo image compression," *Optic. Eng.*, vol. 42, no. 1, pp. 182–189, Jan. 2003.

[20] J. Kim, Y. Kim, and K. Sohn, "Stereoscopic video coding and disparity estimation for low bitrate applications based on MPEG-4 multiple auxiliary components," *Signal Process.: Image Commun.*, vol. 23, no. 6, pp. 405–416, Jul. 2008.
[21] X. M. Li, D. B. Zhao, X. Y. Ji, Q. Wang, and W. Gao, "A fast inter frame prediction algorithm for multiview video coding," in *Proc. IEEE ICIP*, vol. 3. Sep. 2007, pp. 417–420.
[22] *Joint Video Team (JVT) Reference Software Version 17.0* [Online]. Available: http://iphome.hhi.de/suehring/tml/download/old_jm
[23] *Draft Call for Proposals on 3D Video Coding Technology*, document N11830, ISO/IEC JTC1/SC29/WG11, Daegu, Korea, Jan. 2011.
[24] G. Bjøntegaard, "Calculation of average PSNR differences between RD-curves," document VCEG-M33, ISO/IEC JTC1/SC29/WG11 and ITU-T SG16 Q.6, VCEG 13th Meeting, Austin, TX, Apr. 2001.
[25] M. Tanimoto, "FTV (free viewpoint television) creating ray-based image engineering," in *Proc. IEEE Int. Conf. Image Process.*, vol. 2. Sep. 2005, pp. 25–28.
[26] *Joint Video Team (JVT) Reference Software Version 17.0* [Online]. Available: http://iphome.hhi.de/suehring/tml/download/old_jm

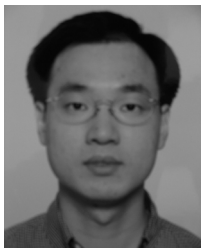
Ka-Man Wong (M'11) received the B.E. degree in computer engineering and the Ph.D. degree in electronic engineering from the City University of Hong Kong, Kowloon, Hong Kong, in 2001 and 2011, respectively.

He was a Senior Research Assistant with the Department of Electronic Engineering, City University of Hong Kong, in 2011. Currently, he is an Engineer with Tencent, Shenzhen, China. His current research interests include motion and disparity-compensated prediction for video coding and content-based image retrieval.

Lai-Man Po (M'92–SM'09) received the B.S. and Ph.D. degrees in electronic engineering from the City University of Hong Kong, Kowloon, Hong Kong, in 1988 and 1991, respectively.

Since 1991, he has been with the Department of Electronic Engineering, City University of Hong Kong, where he is currently an Associate Professor and Program Leader of the Information Engineering Program. He has published over 130 technical journal and conference papers with more than 1500 citations. His current research interests include image and video coding with emphasis on fast encoding algorithms, new motion-compensated prediction techniques, and 3-D video processing.

Dr. Po is currently the Chairman of the IEEE SIGNAL PROCESSING Hong Kong Chapter. He is a member of the Technical Committee on Multimedia Systems and Applications and the IEEE Circuits and Systems Society. He is an Associate Editor of the *Hong Kong Institution of Engineers Transactions*. He was an Organizing Committee Member of the IEEE International Conference on Acoustics, Speech, and Signal Processing in 2003, the IEEE International Conference on Image Processing in 2010, and other conferences.

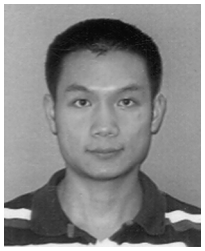


Kwok-Wai Cheung (M'10) received the B.E., M.S., and Ph.D. degrees from the City University of Hong Kong, Kowloon, Hong Kong, in 1990, 1994, and 2001, all in electronic engineering.

From 1990 to 1995, he was with Hong Kong Telecom as Trainee Engineer or Project Engineer. From 1996 to 2002, he was a Research Student or Research Assistant with the Department of Electronic Engineering, City University of Hong Kong. He joined Chu Hai College of Higher Education, Hong Kong, in 2002, where he is currently an Associate Professor with the Department of Computer Science. His current research interests include image or video coding and multimedia database.

Chi-Wang Ting received the B.E. degree in computer engineering and the M.Phil. degree in electronic engineering from the City University of Hong Kong, Kowloon, Hong Kong, in 2002 and 2005, respectively.

Since February 2007, he has been with the Department of Electronic Engineering, City University of Hong Kong, where he is currently an Instructor. His current research interests include fast motion estimation and fast-mode decision algorithms for H.264/AVC.



Ka-Ho Ng (S'10) received the B.E. and M.Phil. degrees in electronic engineering from the City University of Hong Kong, Kowloon, Hong Kong, in 2005 and 2008, respectively.

He is currently pursuing the Ph.D. degree from the Department of Electronic Engineering, City University of Hong Kong. His current research interests include video coding and motion estimation.

Xuyuan Xu (S'11) received the B.E. degree in information engineering from the City University of Hong Kong, Kowloon, Hong Kong, in 2010, where he is currently pursuing the Ph.D. degree with the Department of Electronic Engineering.

His current research interests include 3-D video coding and 3-D view synthesis.

Mr. Xu received the Best Tertiary Student Project of Asia Pacific International and Communication Award (APICTA) in 2010 for his final year project "Stereoscopic video generation from monoscopic video."