Virtual View Synthesis for Free Viewpoint Video and Multiview Video Compression using Gaussian Mixture Modelling

D M Motiur Rahaman, Member, IEEE and Manoranjan Paul, Senior Member, IEEE

Abstract—High quality virtual views need to be synthesized from adjacent available views for free viewpoint video (FVV) and multiview video coding (MVC) to provide users with a more realistic 3D viewing experience of a scene. View synthesis techniques have poor rendering quality due to the occlusion and rounding integer error through warping by creating holes. To remove the holes in the virtual view the existing techniques use spatial and temporal correlation in intra/inter-view images and depth maps. However, they still suffer quality degradation in the boundary region of foreground and background areas due to the low spatial correlation in texture images and low correspondence in inter-view depth maps. In the proposed technique, we use the number of models in Gaussian mixture modelling (GMM) to separate background and foreground pixels to overcome the limitations of the above mentioned techniques. Then, the missing pixels are recovered from the adaptive weighted average of the pixel intensities from the corresponding model(s) of the GMM and the warped image to overcome the error introduced in the warping process. The weights vary with the time to accommodate the changes due to dynamic background and motions of the moving objects for view synthesis. We also introduce an adaptive strategy to reset GMM modelling if the contributions of the pixel intensities from the models drop significantly. Our experimental results indicate that the proposed approach provides 5.40–6.60 dB PSNR improvement compared with relevant methods. To verify the effectiveness of the proposed view synthesis technique, we use it as an extra reference frame in the motion estimation for MVC. The experimental results confirm that the proposed view synthesis is able to improve PSNR by 3.15–5.13 dB compared to the conventional three reference frames.

Index Terms—View synthesis, free viewpoint video, depth image based rendering, multiview video compression, moving picture experts group (MPEG), international telecommunication union (ITU)

I. INTRODUCTION

FVV has attracted considerable attention in recent years as it provides freedom to the user to observe a scene from different angles or viewpoints [1][2][3]. A large number of views with a small baseline are required to facilitate this luxury, which increases transmission bandwidth and storage data significantly. Depth image based rendering (DIBR) is a practical way to reduce storage and transmission bandwidth for multiview videos from color textures and their corresponding depth maps [1][2]. However, in the DIBR technique, portions of regions are not visible in the virtual position due to the front objects termed as occlusion, which create some holes in the video synthesis [3]-[6]. Moreover, warping process from different views cause another source of error due to rounding integer.

Generally, there are two types of methods to fill missing pixels or holes. One is to exploit spatial correlation of the video to fill the missing pixels. In the spatial domain, view blending approaches can reduce the number of holes as two adjacent cameras can cover a relatively wider viewing angle [6]. In this technique, adjacent warped views are combined into a single view, which can reduce the holes. However, only a small number of views are transmitted due to the bandwidth constraint. Therefore, the rendered view would miss some pixel information [7]-[10]. Inpainting technique is normally popular to recover these missing pixels without introducing significant blur artifacts. Inpainting technique in [10][11], after computing the priority of holes’ boundary pixels, the most relevant patch is copied from the source patch by exploiting spatial correlation. However, this process can deteriorate the quality of the view synthesis by being unable to differentiate foreground and background pixels properly. This is due to the low spatial correlation in the perimeter between foreground and background pixels [2][5][6][10][13]. In [14], blocks with missing pixels in terms of decreasing difficulty for inpainting were sorted out. In this technique, explicit instructions called auxiliary information (AI) of the most difficult blocks is transmitted to guide the decoder in the reconstruction process. The decoder can independently fill up missing pixels in the blocks that are easy to inpaint via a template-matching algorithm. In [15], depth information was used to the priority computation and patch distance calculation of the algorithm in [11]. In this technique, the patch whose depth variance is low gives higher priority. However, this may produce distorted synthesized results around the foreground object boundaries in case the boundaries of objects in the depth map are mismatched with that of the color image. Inverse mapping is another popular technique for hole-filling. This technique re-maps the missing
pixel locations in the original view based on the column-shifts of the neighbourhood. In this way, holes can be mapped backward to one of the original views to identify the missing pixel values [7][12]. As this technique also exploits spatial correlation for the column part, it also suffers the hole filling problem in the foreground-background boundary areas.

The other methods use temporal correlation to fill missing pixels of the view synthesis. They are popularly known as background update techniques and they are based on the assumption that an occluded background in the one frame may become visible in the other frames when the foreground objects move away. The techniques in [13][16] generated a static background frame by exploiting temporal correlation and then removing any foreground object with conventional inpainting and clustering techniques depending on the depth map. The experimental results reveal that these techniques improve the quality of the view synthesis significantly compared to other techniques including inpainting techniques [2]. However, this technique suffers quality degradation, due to the dependency on inpainting, warping of a background image and clustering methods. Inaccuracy of any these steps deteriorates the quality of the view synthesis. Moreover, an imperfect depth map may lead to some artefacts of the foreground. In addition, if the background frame is generated from the model sorted by ratio of weight and standard deviation, it does not represent the recent changes of the pixel, and it causes a poor background frame [2][6][17].

Unlike the existing GMM-based techniques, the proposed technique uses the number of models in the GMM to separate background and foreground pixels and modify pixel intensities accordingly. We use an adaptive weighted average to generate a pixel intensity to overcome the error introduced in the warping process. We also use an adaptive reset mechanism to keep the relevancy of the modelling system.

View synthesis techniques are recognized as a promising tool for rendering new views from multiview video plus depth (MVD) for supporting advanced 3D video coding [1][18]. Recently, international organization for standardization (ISO), moving picture experts group (MPEG) and international telecommunication union (ITU) video coding experts group have jointly developed efficient coding tools such as 3D-HEVC [1][3][19][20]. The main focus of the technique in [21] is to integrate a synthesized or disparity-adjusted view into the block-based rate-distortion (RD) optimization framework to improve prediction in MVD. For this, they generate a virtual view and introduce new skip and direct modes using the synthesized view. However, they did not include any explicit hole filling technique to improve the quality of the synthesized view to address occlusion and error due to rounding integer problems. Therefore, the view synthesis prediction in [21] does not provide significant compression ratio improvement compared to 3D-HEVC.

3D-HEVC provides the best compression ratio for MVD data by exploiting the view synthesis optimization (VSO) coding tool [3]. A VSO scheme for the exact view synthesis distortion calculation was proposed by employing a measure called synthesis view distortion change (SVDC). The view rendering was performed iteratively in the encoding process to compute the RD cost. This technique achieves high compression efficiency, but it incurs heavy computation burden to the encoder. Later, view synthesis distortion and depth distortion models without performing view rendering were proposed to reduce computational complexity. However, the accuracy may not be high [22]. In [23], view synthesis distortion estimation for AVC- and HEVC-compatible 3-DV coding technique is proposed with a solution for each of the problems in which the view synthesis distortion function consistently achieves positive coding gains. To enable auto stereoscopy additional views, the receiver side generates the current frames from already encoded adjacent frames and the previous frame of the current view [24]. Moreover, DIBR techniques provide an extra reference by exploiting disparity among adjacent views. Due to the high similarity of the proposed view synthesis with the current view, this technique provides better prediction compared to the conventional three references (i.e. two frames from adjacent views and the previous frame of the current view) system. To verify the effectiveness of the proposed view synthesis, we use it as an additional reference frame in the motion estimation for MVC. The experimental results confirm that the proposed view synthesis is able to improve PSNR significantly compared to the conventional three reference frames. As we do not need any motion estimation for the virtual view, the computational time of using four reference frames is comparable to the three reference frames. We also use two reference frames using the proposed view synthesis view and the previous frame. The results show that we can improve the PSNR compared to the three reference scheme in multiview compression. This proposed scheme also reduces computational time significantly.

The preliminary concept is published in [2]. The new contributions in this paper are (i) adaptive weighting, (ii) adaptive reset strategy for modelling, (iii) a new way to generate pixel intensity of the virtual view, (iv) view synthesis using synthesized images, (v) introducing four and two references technique instead of standard three references MVC.

The rest of this paper is organized as follows: Section II describes the proposed view synthesis approach with adaptive weighting hole filling technique, Section III focuses on view synthesis for MVC, while Section IV presents experimental results. The conclusions are given in Section V.

II. PROPOSED VIEW SYNTHESIS TECHNIQUE

In a number of occasions, GMM technique is used for view synthesis using background frame. However, in the proposed technique, the number of models in the GMM is used to separate background and foreground pixels and modify pixel intensities using the corresponding model-pixel intensities not only for the background model but also other models available in the GMM. The missing pixels of the background are recovered using the adaptive weighted average of the pixel intensities from the model(s) and the warped image to overcome the error introduced in the warping process. In this technique, the inherent characteristics of Gaussian mathematical models are capitalized to recover occluded areas.
It is true that GMM technique is more effective for static background scenarios, however, it is also useful to address pixel intensity problem for the event of occlusion. Moreover, it can handle dynamic (with minor changes) background scenario. To handle more dynamic background and foreground scenario, we have used an adaptive reset mechanism in the proposed method when the current models lose their relevancy. Moreover, unlike in general scenarios, static cameras are normally used in the free viewpoint and multi-view video scenarios.

In the GMM technique, if a pixel position experiences similar intensities over the period, there should be only one model, which indicates that the pixel is a background. On the contrary, if a pixel position experiences different pixel intensities and it is needed to represent with multiple Gaussian models. This indicates that the pixel has both background and foreground in different times. Therefore, the hypothesis is that the number of GMM models would be a good indicator to identify background/foreground pixels. In this technique, the GMM applies on the interpolated view instead of the adjacent view assuming that synthesized previous images of the interpolated view are already available. This technique provides a better pixel correspondence, which leads to better quality compared to both inpainting and background update methods. However, if a pixel position experiences once foreground together with background in other moments, it considered foreground throughout the technique in [2] after experiencing the foreground. Even so, after experiencing foreground pixel intensities, it can experience background pixel intensities again. Based on this hypothesis, we find appropriate background and foreground pixels for filling missing pixel intensities of the virtual view.

Moreover, the setting weight is crucial as the PSNR of the view synthesis may vary up to 1.0~6.0dB by using different weights to balance the contributions between warping image and the learned foreground model. In this paper, we propose an adapting weighting technique to fill up foreground pixels of the view synthesis. In the experiment, we have observed that if a video has more moving regions, the tendency is that it has more pixels which use two or more Gaussian models. In this situation, the relatively large contribution from warped image provides a better quality of the view synthesis. It is due to the less relevancy of the learned foreground with the view synthesis for the rapid changes of foreground within a short period of time. In this paper, we have first established a relationship between the weight and the percentage of multiple Gaussian models using a number of videos. Then, we apply the relationship in the view generation. The experimental results show that the proposed technique does not sacrifice any significant quality degradation compared to the maximum achievable quality through setting the weights.

In the proposed technique, $n$-th texture images from two adjacent views are warped into a virtual position by using their corresponding depth maps and camera parameters to generate $n$-th image of the intermediate view. But warped images contain holes due to the occlusion and rounding integer error. Two warped images are blended to reduce these missing pixels to make a warped image. This procedure reduces the number of holes but does not help to recover all missing pixel intensities specially occluded regions. To recover these missing pixels, we use GMM technique to model each pixel with available previous frames of the virtual view as shown in Fig. 1. In our experiment, we assume that we have already 1 to $(n-1)$-th frames for the GMM when we generate $n$-th frame of a virtual view. In this technique, parameter $i$ is used to reset the modelling after a certain interval, where $i=2, 3, 4...n$. The resetting of the modelling depends on weighting factor (details see in Section III (C)). Initially, we use original frame for GMM. Moreover, we also use synthesized frame for GMM. Then, based on the number of GMM models, each pixel is classified as a foreground or background pixel. After that, the missing pixel intensities of the background and foreground areas are filled from the adaptive weighted intensities between the blended image and the learned background and foreground model(s) of the GMM. The subsequent section describes interpolating virtual view, GMM technique, adaptive hole filling technique, and choosing the value of weighting factor.

A. Interpolating Virtual View

In our experiment, we assume that the sender usually transmits two texture images and their correspondence depth maps of a same scene captured by two cameras at the same instant. Generally, depth maps represent the distance of objects from the camera which is quantized into 256 different values where 0 and 255 represent the farthest and nearest distance respectively. The true depth values $Z$ are converted from the encoded depth map $\Omega$ by using (1) [2][7]:

$$Z = \frac{Z_{\text{near}}Z_{\text{far}}}{(\frac{255}{255})^2(Z_{\text{near}} - Z_{\text{far}}) + Z_{\text{near}}} \tag{1}$$
where \( Z_{\text{far}} \) and \( Z_{\text{near}} \) are the farthest and nearest depth in a scene respectively.

Then the disparity \( d \) between the reference view (i.e. adjacent views) and the virtual view is determined from the camera parameters such as camera focal length \( f \) and baseline distances \( l \) by using (2)

\[
d = \frac{fl}{Z}. \tag{2}
\]

After that, texture images are aligned in the virtual position based on the calculated disparity values [7][25][25]. However, this aligned texture contains many holes due to rounding integer error and the occlusion problem. Warped images \((\Gamma_i \text{ and } \Gamma_f)\) are blended based on four conditions to minimize the holes problems as follows:

Case I: When there are no holes into the warped texture \( \Gamma_i \) and warped texture \( \Gamma_f \), we are taking the average of the corresponding pixels.

Case II: If there are no holes into the warped texture \( \Gamma_i \), but holes into the warped texture \( \Gamma_f \), we are taking the pixel intensity from the warped texture \( \Gamma_f \).

Case III: If there are holes into the warped texture \( \Gamma_i \), but no holes into the warped texture \( \Gamma_f \), we are taking the pixel intensity from the warped texture \( \Gamma_f \).

Case IV: If there are holes into both warped textures, we are considering pixel intensity is equal to zero.

This procedure reduces the number of holes but does not help to recover all missing pixel intensities. To recover missing pixels, we model each pixel by using the GMM technique using the previously generated images in the virtual view.

### B. GMM Technique

The GMM technique is usually used for separating background and foreground pixels at pixel level from dynamic environment, where each pixel is modeled independently by a mixture of \( K \)-th Gaussian distributions (usual setting \( K=3 \) [27][28]. In our proposed technique, let us assume at time \( t \), the value of \( k \)-th Gaussian intensity \( n_{k,t} \), mean \( \mu_{k,t} \), variance \( \sigma_{k,t}^2 \), and weight in the mixture \( \omega_{k,t} \), so that \( \sum_{k=1}^{K} \omega_{k,t} = 1 \). In our proposed technique, we set the initial parameters from literature as follows [27][29]: standard deviation \( \sigma_k \) = 2.5, weight \( \omega_k \) = 0.001 and learning rate, \( \alpha = 0.1 \). A learning parameter \( 0 < \alpha < 1 \) is used for balancing the contribution between present and previous values of aforementioned parameters.

After initialization, the current pixels are used to match with \( k \)-th Gaussian for every new observation if the condition

\[
|X_t - \mu_{k,t}| \leq 2.5\sigma_{k,t}
\]

is satisfied against existing models, where \( X_t \) is the new pixel intensity at time \( t \). If a model matches, the Gaussian model will be updated as follows:

\[
\mu_{k,t} \leftarrow (1-\alpha)\mu_{k,t-1} + \alpha X_t;
\tag{3}
\]

\[
\sigma_{k,t}^2 \leftarrow (1-\alpha)\sigma_{k,t-1}^2 + \alpha (X_t - \mu_{k,t})^T (X_t - \mu_{k,t});
\tag{4}
\]

\[
\omega_{k,t} \leftarrow (1-\alpha)\omega_{k,t-1} + \alpha,
\tag{5}
\]

and the weights of other Gaussians models are updated as

\[
\omega_{k,t} \leftarrow (1-\alpha)\omega_{k,t-1}.
\tag{6}
\]

Then, the value of weights is normalized among all models in such a way that \( \sum_{k=1}^{K} \omega_{k,t} = 1 \). Conversely, if a model fails to match, then a new model is introduced with initial parameter values. If it is already crossed the maximum allowable number of models, based on the value of weight/standard deviation, the new model substitutes an existing model. If a pixel intensity of a color \( c \) satisfies a model \( (k) \), we store the pixel intensity as the recent value \( B_{k,t}^c \) of the corresponding model and color. After that, we use this value to recover missing pixel values.

### C. Hole Filling

If a pixel experiences only one model over the time in different frames, it represents static background pixels, conversely, if a pixel experiences more than one models, it represents foreground and background pixels, where the highest value of weight/standard deviation represents the most stable background [2]. As the GMM has inherent capacity to capture background and foreground pixel intensities by exploiting temporal correlation, missing pixel intensities of an occluded area are successfully recovered. In the proposed technique, if a pixel has only one model, the pixel intensity of the synthesized final image \( \Psi_i \) is taken from the recent value i.e. \( B_{k,t}^c \) of the model and warped image. However, a video with larger moving objects and high motions changes the content frequently, as a result, the models lose relevancy with the past frames. In this scenario, learned foreground using GMM does not provide adequate pixel intensity for a virtual view. Therefore, we need to reset the models after a certain interval. Otherwise, error propagates through the whole systems. On the other hand, the pixel intensity of the synthesized final image is taken as a weighted average from the blended image and the recent value of the model, which provides the lowest value in terms of weight/standard deviation. The detail of the interpolated image recovering technique using GMM is described below:

Case I: If a pixel experiences only one model over the whole duration for a given colour, we store the recent value \( B_{k,t}^c \) of the colour for the final image synthesis by using

\[
\Psi_i^c = (1 - \xi) \Phi_i^c + (1.0 + 0.5842 - \xi) B_{k,t}^c.
\tag{7}
\]
where $\xi$ is weighting factor (see details calculation of constants used in (7) in the subsection D of section II) and $\Phi_i^c$ is the outcome of inverse mapping.

Case 2: If a pixel experiences more than one model for a given colour over the duration, whether it would be the foreground or background pixels, initially, we use the inverse mapping technique [7] to fill the holes. Then, we find the smallest distances between pixel intensities of $\Phi_i^c$ and the recent values $B_{i,t}^c$, $B_{2,t}^c$ and $B_{3,t}^c$ as follows:

$$
\Delta_1 = |\Phi_i^c - B_{1,t}^c| \\
\Delta_2 = |\Phi_i^c - B_{2,t}^c| \\
\Delta_3 = |\Phi_i^c - B_{3,t}^c| \\
\Delta = \min(\Delta_1, \Delta_2, \Delta_3)
$$

(8)

If $\Delta = \Delta_1$, the pixel represents the background at that moment for a given colour and we store the recent value $B_{i,t}^c$ of the colour for the final synthesis image by using (7).

If $\Delta = \Delta_2$ or $\Delta = \Delta_3$ the pixel represents foreground at that moment, therefore, we choose a weight factor ($\xi$) for selecting the fraction of the $\Phi_i^c$ and the recent value of the second or third model ($B_{2,t}^c$ or $B_{3,t}^c$) as follows:

$$
\Psi_i^c = \xi \Phi_i^c + (1 - \xi) B_{k,t}^c.
$$

(9)

where the value of $k$ is either 2 or 3.

### D. Adaptive Weighting Factor

In the proposed technique, we have observed that different values of weighting factor $\xi$ provide different qualities of virtual view (see Fig. 6). Thus, it is essential to determine the value of $\xi$ in different frames and videos. To determine the value of $\xi$, we have tried to learn the factor which influences the value to provide a better virtual view. Our theory is that if a video has larger foreground areas with high motion, the video should have larger number of pixels with multiple models in GMM. In this case, the video should show a tendency to take more pixel intensities from the warped image compared to the background image as the background image loses its relevance more frequently over the time. Thus, the value of $\xi$ would be proportionate with the number of multiple models in GMM. Through experiments, we have observed that there is a positive relationship of the value of $\xi$ with the number of pixels with multiple number of models of a frame. In this scenario, learned foreground using GMM does not provide adequate pixel intensity for a virtual view, thus the value of $\xi$ should be higher for those cases as the warped image has more contribution compared to the learned foreground. We derive a relationship between multiple models and a weighting factor ($\xi$) for a number of videos and then we use the relation in each frame of a video to adaptively set the value of the weighting factor. The weighting factor is formulated as a ratio of the pixel number with multiple models and the total number of pixels of a frame in GMM, which is given below:

$$
\xi = \frac{\text{number of pixels with multiple models}}{\text{total number of pixels of a frame}}
$$

(10)

where $A_1$, $A_2$, $A_3$, and $A_2,3$ are the number of pixels with one model, two models, three models and two/three models respectively. From (10) we fit a third order polynomial $(0.0004(A_2,3)^3 - 0.0116 (A_2,3)^2 + 0.1214 A_2,3 + 0.5842)$ to derive the relationship as shown in Fig. 2. Note that we have calculated the weighting factor in $(n-1)^{th}$ frame and used it to generate $n^{th}$ virtual frame.

The main idea of the adaptive weighting factor determination is that if a video has larger moving objects, the large contribution comes from warped images ensure better view synthesis. Moreover, it helps to reset modelling after certain interval depending on number of multiple models. With increasing multiple models the contribution of learned background/foreground reduces to ensure better view synthesis. In our experiment, when the value of $\xi$ is 0.9 or higher we reset the modelling. The figure shows that the value of $\xi$ is very close to 1 when the number of multiple models is close to 13% or more i.e., there is no or little contribution of learned foreground to form a virtual view if a frame has a higher moving object. However, the learned background of GMM still has a great contribution to form the virtual frame in static and uncovered background areas. When we compare the adaptive weighting factor to generate virtual view, we do not sacrifice any significant quality degradation compared to the maximum achievable quality by setting the weights from 0 to 1 (see Fig. 7).

### III. View Synthesis for MVC

Adjacent views of multiview video sequences are captured by multiple cameras with slightly different angles. Therefore, there are disparities among the different views. Moreover, co-located pixels/blocks at different instances of the same views are predicted by the motion estimation technique. However, finding co-located pixels/blocks on different frames by using motion estimation and disparity estimation is time consuming [30][31]. Therefore, a reduction of computation for searching motion parameters such as motion vector is an important aspect of current research [32]-[34]. Thus, the best policy is reducing the

![Fig. 2: Trend of weighting factor ($\xi$).](image)
number of reference views. Traditionally, three references such as already encoded frames of adjacent views (reference frames 1 and 2 in Fig. 3) and the previous frame of the current view (reference frame 3 in Fig. 3) are used to encode each frame of dependent views [3][24]. In this technique, a disparity \( d \) is used to find a current block \((X_c, Y_c)\) on adjacent reference views \(((X_{r1}, Y_{r1})\) and \((X_{r2}, Y_{r2})\)) where \(X_{r1} = X_c \pm d\) and \(X_{r2} = X_c \mp d\). This method only considers the horizontal component as multiview video sequences are rectified [3]. Furthermore, motion vectors are predicted to find a current block on the previous frame of the current view i.e. \((X_{r3}, Y_{r3})\) [29][35]. Instead of typical approaches, we use the proposed view synthesis technique to generate a synthesized current frame, which is used as the fourth reference frame. This synthesized frame is almost similar in terms of object position and its motion to the expected current frame. Therefore, we have four candidates for choosing each block to encode the current frame of the middle view i.e. view 2 as shown in Fig. 3. As the fourth reference frame has more similar content with the current frame compared to the other three reference frames, it is expected that encoding the current frame using four reference frames provides better quality. Moreover, using four reference frames does not require significant extra computational time compared to three reference frames as it does not require any disparity or motion estimation.

To see the effectiveness of the proposed virtual frame, we also consider two references such as reference three and reference four as shown in Fig. 3, it also provides better prediction compared to the traditional approaches. Obviously, two reference scheme provides better computational time compared to three reference frame scheme.

### IV. EXPERIMENTAL RESULTS

In our experiment, PSNR is used to measure the squared intensity differences of synthesized and original image pixels. Then based on average PSNR performance, we compare the outcome of the proposed method with the state-of-the-art methods namely View Synthesis Reference Software (VSRS) [36], inpainting [11], and the background update technique [16]. Four standard multiview video sequences are selected for testing the performance of the proposed technique. The input reference viewpoints, the virtual viewpoint and the baseline are listed in Table 1. Fig. 4 demonstrates the results. We use the same warping and blending techniques for all techniques for 100 frames using adjacent views, then we apply an inpainting, background update and the proposed method for refining the blended image. The figure shows that the proposed technique provides better performance compared to the existing hole filling techniques for all video sequences. The improvement range varies from 7.85dB to 11.69dB with average improvement 9.72dB for VSRS, 7.32dB to 8.85dB with average improvement 8.25dB for inpainting and 5.40dB to 7.65dB with average improvement 6.51dB for the background update technique respectively. Furthermore, we compare the outcome of our preliminary paper [2] with the proposed technique. Fig. 5 shows that the proposed technique outperforms the preliminary technique in [2] for all video sequences. In [2], only a single view was taken for warping, whereas the proposed technique uses two adjacent views for warping. Moreover, the proposed method identifies both foreground and background pixel intensities to refine a virtual view when multiple models are used to model a pixel intensities. The model which provides the minimum difference in pixel intensities between the blended image and the different models in GMM for a given moment, represents foreground or background. However, in the previous method, it was considered as a foreground pixel intensity throughout. That’s why the PSNR of the virtual view using the proposed method increases with number of frames compared to the technique in [2] (see Fig. 5).

![Fig. 4: Average PSNR comparison for 100 frames.](image)

| Table 1: Test sequences, synthesized viewpoints and baseline |
|-----------------|-----------------|-----------------|-----------------|
| Sequences       | Input Reference Viewpoints | Target Viewpoint | Baseline        |
| Newspaper       | 6,2              | 4               | 185.37          |
| Lovebird1       | 8,4              | 6               | 148.12          |
| Poznan Street   | 5,3              | 4               | 3.19            |
| Book Arrival    | 10.6             | 8               | 2.31            |
We analyzed PSNR against different values of $\xi$ i.e. 0 to 1. The contribution of the GMM models and the blended images in the proposed approach to reconstruct the final synthesized image is shown in Fig. 6 for 30th frame of each video sequence. The figure reveals that the learned foreground of GMM has some contribution to generate a final image for each video. Note that if we get the maximum PSNR value of a given image where the value of $\xi$ is 0.6 (Lovebird1), it means that the pixel intensities of the 60% and 40% foreground are taken from blended image and learned foreground respectively. A fixed threshold may work for some frames, however, we observed that the threshold is varied for other frames. Thus, it is crucial to use adaptive threshold rather than a fixed threshold.

We also compared the maximum PSNR and adaptive PSNR against predicted frames for Newspaper, Lovebird1, Poznan Street (up to 100 frames) and Book Arrival (full sequences), which is shown in Fig. 7. More specifically, Newspaper, Lovebird1, Poznan Street and Book Arrival video sequences sacrifice 0.10dB, 0.08dB, 0.16dB and 0.23dB PSNR on average. The slope of this curve is controllable by changing the values of $\xi$. If the value of $\xi$ is greater than or equal to 0.9, we reset the modelling in this experiment as the contributions from the GMM models reduce (i.e. 0.1 or less). In our experiment, the modelling is reset 10, 5, 9 and 14 times for 100 frames for Newspaper, Lovebird1, Poznan Street and Book Arrival video sequences respectively. Normally, due to the content of the video sequence, occasionally the PSNR of the view synthesis drops or rises over the time. The proposed reset strategy can handle this trend of changing PSNR as the contributions of the pixel intensities from the model(s) reduces over the time due to the dynamic background and motions of the moving objects.

Fig. 8 illustrates the subjective quality for Newspaper video sequence. Fig. 8 (a) shows the original images, i.e. 10th original frame of the virtual view and the green rectangular boxes are used to mark the cropped and zoomed portion which is shown in Fig. 8 (b) and (c). Similarly, Fig. 8 (d), (g), (j) and (m) shows the view synthesis by VSRS, inpainting, background update and the proposed technique and Fig. 8 (e), (f), (h), (i), (k), (l), (n) and (o) shows corresponding cropped and zoomed images. These figures demonstrate that the proposed technique is able to generate a better view synthesis compared to the three standard methods.

To see the strength of the proposed method, we have applied the proposed scheme in its current form to the synthesized images for modelling and generated view synthesis. We have used synthesized images generated by the inverse mapping technique for GMM. Fig. 9 shows that the proposed technique improves the quality of the synthesized view. This technique improves 0.57dB, 0.15dB, 0.27dB and 0.32dB PSNR on average for Newspaper, Lovebird1, Poznan Street and Book Arrival video sequences compared to the inverse mapping technique respectively. If we get good quality images for learning GMM it should provide better synthesized image. As the parameters of the proposed technique are not optimized for the synthesized views, it gives us moderate improvement. Fig. 10 demonstrates the subjective quality of the proposed technique when we learned the output of the inverse mapping technique [7]. It shows that the proposed technique provide better synthesis images.

To understand the effectiveness of the proposed view synthesized technique in the moving background sequences, we have conducted experiments using Balloons, Kendo, Poznan Hall2 and Undo Dancer video sequences with 50 frames. Fig. 11 shows the performance of the proposed method compared to other recent and relevant method [7]. The figure reveals that the proposed method performs better for Balloons and Kendo videos sequences but not for other two sequences as the first two sequences have relatively less moving background compared to other two. Thus, the proposed techniques still provide better results if the moving background is not too strong.
Fig. 8: Original image (a), synthesis images (d, g, j and m), crop and zoom images (b, c, e, f, h, i, k, l, n and o) for Newspaper video sequence by the proposed method three standard methods.
To encode different resolutions and a wide range of video content for different views in 3D-HEVC, each frame is divided into a number of blocks with various sizes such as 8x8, 16x16, 32x32 and 64x64 pixels and the search length become 8, 16, 32, 64 and 128 pixels. In our experiment, we have considered 32x32 and 64x64 pixel block sizes and 64 pixel search lengths to demonstrate the performance of the proposed four and two reference schemes compared to the existing three reference scheme. Due to the better prediction of the synthesized view, the proposed technique provides better PSNR compared to the conventional approaches, which are shown in Fig. 12(a) for 32x32 and Fig. 12(b) for 64x64. It reveals that the PSNR improvements for two references and four references are varied from 3.18 to 4.95dB, when block sizes are 32x32 pixels. Similarly, when the block sizes are 64x64 pixels, the PSNR improvements for two references and four references are varied from 3.15dB to 5.14dB. The four reference scheme provides better PSNR compared to the two reference and three reference schemes.

Moreover, for the performance justification of the proposed two references and four references methods for eight video sequences, we have generated the RD performance curve using different QPs i.e. 22, 27, 32 and 37 in the scenario of MVC as shown in Fig. 13. We have used 3D-HEVC structure where the first view and third view are encoded using HEVC coding framework. The middle view is encoded using the two adjacent inter-view images, immediate previous intra-view image, and synthesized image. All reference frames are generated from the reconstructed (i.e. decoded) reference frames, so that, both encoder and decoder have the same reference frames. Fig. 13 shows the results for the middle view. We compared the strength of the proposed method in terms of generating better synthesized view which is used as one of reference frames for coding purpose. The experimental results illustrate that the proposed techniques improve RD performance significantly in all video sequences by improving the quality of the synthesized views.

Furthermore, the performance of the four references and two references techniques against the three references technique are evaluated based on the Bjøntegaard-Delta Bit Rate (BD-BR) and Bjøntegaard-Delta PSNR (BD-PSNR) [37] in Table 2, where ‘+’ and ‘-’ sign indicate the increment and decrement respectively. Over eight different video sequences, the four references and two references techniques provide gains 1.07dB and 0.88dB BD-PSNR, while decreasing 29.68% BD-BR and 26.06% BD-BR on average respectively compared to the
The proposed technique outperforms for all video sequences in terms of both improving the BD-PSNR and reducing the BD-BR.

MVC leads to high computational complexity, which limits its application on low power consumption electronic devices such as smart phones [29]. Total encoding time heavily depends on motion and disparity estimation. Research shows that there are no significant time differences for estimating motion and disparity [30][38]. Moreover, MVC exhaustively checks a number of inter/intra modes for a coding unit to select a best mode to encode the coding unit. This procedure increases complexity multiple times compared to the uni-mode technique [39]. Therefore, any technique which skips disparity estimation and/or motion estimation should reduce time complexity. The proposed two reference technique skips disparity estimation and improves PSNR compared to the three reference technique.

Although the proposed technique needs extra computational time for synthesis virtual view, it reduces overall time complexity for MVC by 28.95% whereas the four reference scheme requires an extra 18.42% time on average compared to the existing three reference scheme (see Fig. 14). In both cases, the proposed technique outperforms the three reference scheme in terms of image quality for a given bit rate. All experiments are conducted by a dedicated desktop machine DELL OPTIPLEX 9020 (with Intel core i5-4690 CPU @ 3.50 GHz, 8 GB RAM and 250 GB HDD) running 64-bit Windows 7 operating system. According to the rate-distortion performance (see Table 2 and Fig. 13), the proposed four-reference technique outperforms the proposed two-reference technique for all video sequences. The significant performance gains are observed for the video sequences with camera motions e.g. Undo Dancer, Kendo, Balloons etc. However, the proposed four-reference technique requires around 47% extra computational time compared to the two-reference technique. Thus, our recommendation is to use four-reference technique for the scenarios with video sequences with camera motions and no concern of the computational time requirements, otherwise, use two-reference technique.

<table>
<thead>
<tr>
<th>Sequences</th>
<th>Four References</th>
<th>Two Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>BD-PSNR (dB)</td>
<td>BD-BR (%)</td>
</tr>
<tr>
<td>Newspaper</td>
<td>+1.92</td>
<td>-33.03</td>
</tr>
<tr>
<td>Lovebird1</td>
<td>-0.77</td>
<td>-23.35</td>
</tr>
<tr>
<td>Poznan Street</td>
<td>+0.48</td>
<td>-14.43</td>
</tr>
<tr>
<td>Book Arrival</td>
<td>+1.7</td>
<td>-39.12</td>
</tr>
<tr>
<td>Balloons</td>
<td>+0.75</td>
<td>-38.33</td>
</tr>
<tr>
<td>Kendo</td>
<td>+0.80</td>
<td>-23.88</td>
</tr>
<tr>
<td>Poznan Hall2</td>
<td>+0.80</td>
<td>-28.99</td>
</tr>
<tr>
<td>Undo Dancer</td>
<td>+1.32</td>
<td>-36.28</td>
</tr>
<tr>
<td>Average</td>
<td>+1.07</td>
<td>-29.68</td>
</tr>
</tbody>
</table>
frames as an extra reference frame for MVC, which improves the quality of the encoded frame on average 0.73dB compared to the standard technique. Another version of the proposed technique provides 0.68dB image quality improvement with reduced computational time compared to the existing MVC technique.

REFERENCES


Manoranjan Paul (M’03 SM’13) received Ph.D. from Monash University in 2005. He has worked as a Research Fellow at the University of New South Wales, Monash University and Nanyang Technological University during 2005 to 2011. He is currently working as an Associate Professor in the School of Computing and Mathematics, Charles Sturt University. His major research interests are in the fields of image/video coding/compression, EEG signal analysis, and computer vision. He has published more than 150 refereed papers in international journals and conferences. A/Professor Paul regularly published journal articles in the IEEE Transactions, which are considered as the top ranked journals in the relevant fields. He was a keynote speaker at IEEE DICTA 2017, WOWMOM Workshop 2014, DICTA 2013, and ICCT 2010. A/Professor Paul is a Senior Member of the IEEE and ACS. He has served as a Guest Editor of the Journal of Multimedia and Journal of Computers for five special issues. Currently A/Professor Paul is an Associate Editor of the EURASIP Journal on Advances in Signal Processing. He is selected as an ICT Researcher of the Year 2017 (Finalist) by Australian Computer Society.

M Motiur Rahman has completed his B.Sc. degree from the Department of Electrical and Electronic Engineering, Rajshahi University of Engineering & Technology, Rajshahi, Bangladesh. After accomplishing his degree, he joined as a lecturer in the Electrical, Electronics and Telecommunication Department, Dhaka International University, Dhaka, Bangladesh. He is a student member of the IEEE. Currently, he is working as a Ph.D. student at Charles Sturt University, Australia. He has published several papers in the area of image processing and video coding.