VIRTUAL VIEW SYNTHESIS USING MULTI-VIEW VIDEO SEQUENCES
Il-Lyong Jung, Taeyoung Chung, Kwanwoong Song, Chang-Su Kim
School of Electrical Engineering, Korea University, Seoul, Korea
E-mails: {illyong, lovelool17, kwsong71, changsukim}@korea.ac.kr

ABSTRACT

A virtual view synthesis algorithm using multi-view video sequences, which inherits the advantages of both the forward warping and the inverse warping, is proposed in this work. First, we use the inverse warping to synthesize a virtual view without holes from the nearest two views. Second, we detect occluded regions in the synthesized view based on the uniqueness constraint. Then, we refine the occluded regions using the information in the farther views based on the forward warping technique. Simulation results demonstrate that the proposed algorithm provides significantly higher PSNR performances than the conventional inverse warping scheme.

Index Terms—Stereo matching, view synthesis, multi-view video, forward warping, and inverse warping.

1. INTRODUCTION

Recently, the demands for more realistic multimedia data have been increased significantly. Three-dimensional (3-D) video or free-view video technologies have drawn much attention from both academia and industry, since they promise to offer richer experience than traditional video. However, it is a difficult task to capture a scene from a large number of viewpoints, which are necessary to achieve realism. Instead, attempts have been made to synthesize virtual views [1], while capturing real views from a few selected viewpoints only.

A variety of view synthesis techniques have been proposed. According to the coordinate system of view synthesis, these techniques can be classified into forward warping or inverse warping [2]. In the forward warping, as shown in Fig. 1 (a), the stereo matching is performed to estimate disparity vectors based on the coordinate system of one of the reference views (I_L or I_R), and the disparity vectors are scaled to synthesize a virtual view I_V. Since the stereo matching is not one-to-one in general, the forward warping may generate holes in the virtual view I_V. Also, if a scaled disparity vector s · d has non-integer-pixel accuracy, a post processing step is required to interpolate pixel values on the integer-pixel accuracy grid in I_V. However, the forward warping has the advantage that it can detect occlusion regions effectively [3]. On the other hand, in the inverse warping, as shown in Fig. 1 (b), the bi-directional matching is performed directly on the coordinate system of the virtual view to be synthesized [4, 5]. The inverse warping does not generate holes in the synthesized view. It does not require the post processing step either. However, occlusions are harder to detect in the inverse warping, and they tend to yield ghost artifacts in the synthesized view.

In this work, we propose a view synthesis algorithm for multi-view video sequences, which combines the complementary advantages of the forward warping and the inverse warping. First, we employ the inverse warping to estimate disparity vectors directly on the coordinate system of a virtual view using the nearest two reference views. Second, we detect occluded regions based on the uniqueness constraint. Then, we employ the forward warping from farther reference views to conceal the occluded regions. Moreover, we introduce a temporal coherence term to estimate disparity vectors more accurately. Simulation results demonstrate that the proposed algorithm synthesizes virtual views faithfully, while reducing occlusion artifacts.

The rest of this paper is organized as follows. Section 2 describes the proposed algorithm. Section 3 presents experimental results. Finally, concluding remarks are given in Section 4.

2. PROPOSED ALGORITHM

As shown in Fig. 2, the proposed algorithm synthesizes a virtual view I_V using the information in four reference views I_1, I_2, I_3, and I_4 in a multi-view video sequence. We assume that the multi-view cameras are in a 1-D parallel configuration, the distance between two adjacent cameras is 1, and the distance between I_2 and I_V is s.

To synthesize and render the virtual view I_V, we should compute accurate disparity vectors. To this end, we employ the belief propagation (BP) algorithm in [6]. Let q denote a neighboring pixel of pixel p, and d_p denote a disparity label of p. Also, let d denote
two adjacent reference views is 1, the distance between the weighing coefficient is set to $w = \frac{1}{1 - s}$. For example, in Fig. 2, suppose that we match pixels in view 1 to view 2. If more than two pixels in view 1 are matched to the same pixel in view 2, those pixels tend to be occluded in view 2 and tend to gather to the left side of a foreground object in view 1.

We can use the uniqueness constraint in the inverse mapping as well. During the bi-directional matching, if more than two pixels in the matrix that contains the disparity labels of all pixels in an image.

Then, for a candidate matrix $d$, we define the energy function as

$$E(d) = E_{\text{data}}(d) + E_{\text{smooth}}(d)$$

$$= \sum_p D(d_p) + \sum_{p,q} V(d_p,d_q),$$

(1)

where $E_{\text{data}}(d)$ is the data term, which is the sum of matching errors $D(d_p)$ between $p$ and its corresponding point specified by $d_p$. The smoothness term $E_{\text{smooth}}(d)$ is the sum of disparity differences between two neighboring pixels $p$ and $q$. Specifically, the disparity difference is defined as

$$V(d_p,d_q) = \min\{|d_p - d_q|, \tau_d\}$$

to avoid assigning too high costs across real edges. The disparity vector matrix $d$ is selected to minimize the energy function $E(d)$ in Eq. (1).

2.1. Stereo Matching Based on Inverse Warping

We first estimate the disparity vector $d_p$ of pixel $p$ in the virtual view $I_v$ using the left view $I_2$ and the right view $I_3$. We employ the inverse warping, and use the bi-directional matching cost as the data cost $D(d_p)$ in Eq. (1). Specifically,

$$D(d_p) = \lambda_d \cdot \min\{|I_2(x + s^p \cdot d_p,y) - I_3(x - (1 - s) \cdot d_p,y)|, \tau_d\}.$$

After obtaining disparity vectors using the belief propagation algorithm, we render pixel $p = (x,y)$ in the virtual view $I_v$ by

$$I_v(x,y) = w \cdot I_2(x + s^p \cdot d_p,y) + (1 - w) \cdot I_3(x - (1 - s) \cdot d_p,y),$$

where the weighing coefficient $w$ is set to $1 - s$, so that the pixel in a nearer view is assigned a higher weight.

2.2. Occlusion Detection

Although the inverse warping generates a virtual view without holes, it often yields ghost and blurring artifacts in occluded regions. Therefore, we detect the occluded regions using the uniqueness constraint, which was proposed for the forward mapping in [7]. For example, in Fig. 2, suppose that we match pixels in view $I_1$ to view $I_2$. If more than two pixels in $I_1$ are matched to the same pixel in $I_2$, those pixels tend to be occluded in $I_2$ and tend to gather to the left side of a foreground object in $I_1$.

We can use the uniqueness constraint in the inverse mapping as well. During the bi-directional matching, if more than two pixels in $I_v$ are matched to the same pixel in the left view $I_2$, those pixels in $I_v$ are declared to be occluded in the left view $I_2$. Those pixels compose the occluded region $O_L$, which is depicted as the blue region in Fig. 3. Similarly, if more than two pixels in $I_v$ are matched to the same pixel in the right view $I_3$, those pixels compose the occluded region $O_R$, which is depicted as the red region in Fig. 3.

2.3. Occlusion Handling Based on Forward Warping

To refine the pixel values in the occluded regions $O_L$ and $O_R$, we use the information in the farther views $I_4$ and $I_1$, respectively.

Case 1 - Occluded region $O_L$:

The pixels in $O_L$ tend to be not visible in the left view $I_2$ and gather to the right sides of foreground objects. Therefore, to synthesize the pixels in $O_L$, we use the information in the right views $I_3$ and $I_4$. Specifically, we obtain the disparity vector $d_{43}$ that matches a pixel in $I_4$ to $I_3$, using the conventional BP algorithm [6]. Specifically, the matching cost is defined as

$$D(d_{43}) = \lambda_d \cdot \min\{|I_4(x,y) - I_3(x + d_{43},y)|, \tau_d\}.$$

Then, based on the photo-consistency among the matching pixels, a virtual pixel in $O_L$ is refined by

$$I_v(x + (2 - s)d_{43},y) = w' \cdot I_3(x + d_{43},y) + (1 - w') \cdot I_4(x,y),$$

where $w'$ is the weighing coefficient.

Fig. 2. A virtual view $I_v$ is synthesized using the information in four reference views $I_1$, $I_2$, $I_3$, and $I_4$. We assume that the distance between two adjacent reference views is 1, the distance between $I_2$ and $I_v$ is $s$, and the distance between $I_v$ and $I_3$ is $1 - s$.

Fig. 3. Occlusion Detection. The red region $O_R$ contains the pixels that are occluded and not visible in the right view $I_3$. The blue region $O_L$ contains the pixels occluded in the left view $I_2$. The smoothness term is the sum of disparity differences between two neighboring pixels $p$ and $q$. Specifically, the disparity difference is defined as $V(d_p,d_q) = \min\{|d_p - d_q|, \tau_d\}$. to avoid assigning too high costs across real edges. The disparity vector matrix $d$ is selected to minimize the energy function $E(d)$ in Eq. (1).
where the weighting coefficient $w' = \frac{(1-s)^{-1}}{(1-s)^{-1} + (2-s)^{-1}}$, so that the pixels in $I_3$ and $I_4$ are weighted by the inverses of the distances from the virtual view $I_v$.

• Case 2 - Occluded region $O_R$:

Similarly, the pixels in $O_R$ are not visible in the right view $I_3$, and they are rendered using the information in the left views $I_1$ and $I_2$. Based on the BP algorithm, we obtain the disparity vector $d_{12}$ using the matching cost

$$D(d_{12}) = \lambda_d \cdot \min\{|I_1(x, y) - I_2(x + d_{12}, y)|, \tau_d\}$$

Then, a virtual pixel in $O_R$ is synthesized via

$$I_v(x + (1 + s)d_{12}, y) = w''' \cdot I_1(x, y) + (1 - w''') \cdot I_2(x + d_{12}, y),$$

where $w''' = \frac{(1+s)^{-1}}{(1+s)^{-1} + s^{-1}}$.

2.4. Temporal Coherence Term

In typical multi-view sequences, a frame at time instance $t$ is highly correlated with the frame at the previous time instance $t - 1$. Therefore, during the inverse warping, we can improve the reliability of the disparity estimation by introducing a temporal coherence term.

Fig. 4 shows the relationship among the disparity vectors and the motion vectors. Specifically, we have

$$p_v^{t-1} = p_v^t - (1 - s)d^t + m_R + (1 - s)d^{t-1}$$

In general, an object point has similar depths in temporally adjacent frames, i.e., $d^t \approx d^{t-1}$. Therefore, from Eq. (2), we have $m_L \approx m_R$. Intuitively, the motion vector $m_L$ of a point in the left view should be similar to the motion vector $m_R$ in the right view. Therefore, we first compute the motion vector fields of the left view and the right view. Then, for each disparity candidate $d'$ of pixel $p_v^t$, we find the corresponding points in the left view and the right view and their motion vectors $m_L$ and $m_R$. Then, we compute the temporal coherence term by

$$E_{\text{motion}}(d) = \sum_p M(d_p)$$

where

$$M(d_p) = \lambda_m \cdot \min\{\|m_L - m_R\|, \tau_m\}.$$
Fig. 7. Comparison of the estimated depth maps for the “Akko and Kayo” sequence: (a) without and (b) with the temporal coherence term.

sponding synthesized view, obtained by the inverse warping using $I_2$ and $I_3$ only. We see that there are blurring artifacts near the object boundaries, and some details are lost. Fig. 5 (c) shows the result, when the proposed occlusion handling is incorporated with the inverse warping. We see that the details are more faithfully reconstructed. Fig. 6 shows the results on the “Newspaper” sequence, which show similar tendencies to Fig. 5.

Fig. 7 shows how the temporal coherence term improves the accuracy of the disparity estimation. We see that, by using the temporal coherence term, the proposed algorithm effectively suppresses the artifacts due to incorrectly estimated disparity vectors.

Next, we compare the PSNR performances in Fig. 8. Compared with the inverse warping using the two nearest views only, the occlusion handling improves the PSNR performance by more than 1 dB on both sequences. Also, we see that the proposed algorithm provides even better performance, when the temporal coherence term is incorporated into the energy function. Specifically, the temporal coherence term improves the performances by about 0.4 dB on the “Akko and Kayo” sequence and about 0.1 dB on the “Newspaper” sequence, respectively. These simulation results indicate that the proposed view synthesis algorithm is efficient for rendering virtual views.

4. CONCLUSION

In this work, we proposed a virtual view synthesis algorithm for multi-view video sequences, which combines the complementary advantages of the forward warping and the inverse warping. After synthesizing a view without holes using the inverse warping, the proposed algorithm detects occluded regions using the uniqueness constraint and refines the occluded regions using the forward warping from the farther views. Simulation results demonstrated that the proposed algorithm synthesizes virtual views more faithfully than the conventional inverse warping. Moreover, it was shown that the temporal coherence term can improve the accuracy of the disparity estimation.

5. REFERENCES


