Towards automatic Stereoscopic Video Synthesis from a Casual Monocular video

Lin Zhong
CBIM, Computer Science Department
Rutgers University
Piscataway, Nj, USA
linzhong@cs.rutgers.com

Sen wang
Corporate Research and Engineering
Eastman Kodak Company
Rochester, NY, USA
sen.wang@kodak.com

Minwoo Park
Corporate Research and Engineering
Eastman Kodak Company
Rochester, NY, USA
minwoo.park@kodak.com

Rodney Miller
Corporate Research and Engineering
Eastman Kodak Company
Rochester, NY, USA
minwoo.park@kodak.com

Dimitris Metaxas
CBIM, Computer Science Department
Rutgers University
Piscataway, Nj, USA
dnm@cs.rutgers.com

***Abstract*—Automatically synthesizing 3D content from a causal monocular video has become an important problem. Previous works either use no geometry information, or rely on precise 3D geometry information. Therefore, they cannot obtain reasonable results if the 3D structure in the scene is complex, or noisy 3D geometry information is estimated from monocular videos. In this paper, we present an automatic and robust framework to synthesize stereoscopic videos from casual 2D monocular videos. First, 3D geometry information (e.g., camera parameters, depth map) are extracted from the 2D input video. Then a Bayesian-based View Synthesis (BVS) approach is proposed to render high-quality new virtual views for stereoscopic video to deal with noisy 3D geometry information. Extensive experiments on various videos demonstrate that BVS can synthesize more accurate views than other methods, and our proposed framework also be able to generate high-quality 3D videos.**

***Keywords*-Stereoscopic video synthesis, automatic, Bayesian-based View Synthesis.**

I. Introduction

Advances in display technologies and the commercial success of 3D motion pictures in recent years have spurred interest to enable consumers to create 3D content. While 3D content can be created by stereoscopic cameras, the vast majority of captured images and videos exist only in 2D. Therefore, many interactive 2D-to-3D conversion methods have been proposed. Stereoscopic videos are generated by synthesizing proper sequence of left and right views based on given monocular view sequence with human interactions. Unfortunately, these interactive 2D-to-3D conversion processes are not accessible to the general public.

Although there are many automatic approaches, existing methods unavoidably confront three key challenges. *First*, the 3D model of the scene is required to render new virtual views, which is quite difficult to be reconstructed from casual videos. *Second*, 3D geometry information estimated from a monocular video is still error-prone, e.g., camera parameters, which cannot meet the requirements of current image-based rendering (IBR) methods [1], [2], [3]. *Third*, image quality of synthesized virtual views degrades greatly due to occlusions. While several methods have been proposed to address one or two of these challenges, to the best of our knowledge, none of them can deal with all of these challenges simultaneously. Fig. 1(a) and Fig. 1(b) show some of the poor results by [1], [2] due to these challenges.

Figure 1. (a) Interpolation [1] (b) Blending [2] (c) Ground truth (green box) (d) Our method (red box). Most IBR methods suffer from inaccurate estimated 3D geometry information, especially at object boundaries. For zoom out version, refer to Figure 6.

In this paper, we present a framework to automatically convert casual monocular videos[1] to stereoscopic videos while addressing the aforementioned challenges (some results see Fig. 1(d)).

The main contributions of our work are twofold: 1) To the best of our knowledge, we are the first to put forward an automatic stereoscopic video synthesis framework to convert casually captured 2D videos[1] of complex scenes into 3D videos without ruining actual 3D geometry scene structures. 2) A Bayesian-based view synthesis (BVS) approach is proposed to address the challenges to synthesize high-quality virtual views with inaccurate 3D geometry information. Extensive experiments on various videos, including two public videos, two consumer videos, and one professional movie clip, demonstrate the effectiveness of our approach.

II. Relevant Work

Depending on the degree of human interaction, 2D-to-3D approaches can be categorized into three techniques: *manual*, *semi-automatic*, and *automatic*.

Manual approaches manually assign different disparity values to pixels of different objects, and then shift these pixels horizontally by their disparities to produce a sense of parallax. The holes produced by shifting are also filled with proper pixels manually [4].

Semi-automatic approaches require the users to label partial 3D information (e.g., scribbles, stroke) for certain frames (first or key-frames of every shot) to obtain the dense disparity or depth map [5], [6]. The 3D information from these labeled frames is propagated to other frames. However, the results may degrade significantly if the frames in one shot are not so similar.

Automatic approaches can be classified into two categories: non-geometric and geometric methods. *Non-geometric* methods directly render new virtual views from one nearby image in the monocular frame sequence, e.g., time-shifting [7]. However, it is not guaranteed that the nearby image frame and the target virtual view are similar enough to render the desired target virtual view. They also cannot preserve the 3D geometry structure of the scene. *Geometric* methods basically consist of two main steps: exploration of underlying 3D geometry information and synthesis of a new virtual view. For some simple scenes captured under controlled conditions, the full and accurate 3D geometry information (e.g., 3D model) can be recovered [8]. A new view can be easily rendered using conventional graphics techniques.

Our proposed framework is also an automatic method. However, different from the previous work [8], our framework does not rely on the 3D model of the scene to render virtual views. Unlike the previous works [9], [10], [11], our proposed Bayesian-based view synthesis (BVS) approach uses available noisy 3D geometry information and it is not so sensitive to noise, which is also different from the previous works [2], [3].

[1]None of the existing automatic methods including ours can deal with independent moving objects, stationary camera, and zoom in/zoom out cameras. It is a matter of our future work.

III. Proposed Framework

Our framework includes three main parts: 1) Structure from Motion [12] is employed to estimate the camera parameters for each frame and the sparse point clouds of the scene; 2) An efficient dense disparity/depth map recovery approach is implemented based on fast mean-shift belief propagation proposed in [13]; and 3) A Bayesian-based view synthesis algorithm (BVS) is proposed to synthesize new virtual views for left-/right- sequences. The flow is illustrated in Fig. 2.

A. Structure from Motion (SfM)

We modify *Bundler* from Photo Tourism [12] to extract positions and rotations of a camera from a 2D video. First, sets of keypoints [14] within certain temporal neighborhoods are matched to save computational cost, since our inputs are ordered 2D frames. Second, only some key-frames are selected to guarantee enough baselines and reduce the numerical errors in solving camera parameters. Then we rerun *Bundler* for the entire frame sequence with the interpolated camera parameters estimated from these key-frames as initial values for numerical stability.

B. Depth Recovery

We first form a stereo pair for each frame in a way that the pair has enough baseline and enough shared coverage [15]. Then the selected stereo pair is rectified using a fundamental matrix estimated from keypoints matching [14] between the pair. Next, we perform 1D search along epipolar lines to estimate disparity using a 3×3 block matching by a normalized cross correlation (NCC). The block matching generates much noise in general, especially for the texture-less and object boundaries areas (see Fig. 3(a)). Therefore we formulate the estimation of disparity as a maximization problem on a 4-connected Markov Random Field (MRF) [13] to get a more robust result (see Fig. 3(b)). After we estimate the dense disparity map, we combined it with the camera parameters of the stereo pair to obtain the depth map by the triangulation method.

C. Stereoscopic Video Synthesis

Basically, one stereoscopic video is a combination of two synchronized left and right view sequences. To synthesize the virtual view sequences for left-/right- eye, we need to calculate the camera parameters (e.g., position, orientation) for these virtual views. As shown in Fig. 2 (red circles), one left and right virtual views are estimated for each original frame. The distance between one pair of left-/right- views is constant, d_{eye}, and their directions are perpendicular to the line connecting them. With the estimated 3D geometry information (e.g., camera parameters, depth map), the virtual views are synthesized using our proposed BVS approach (Section IV).

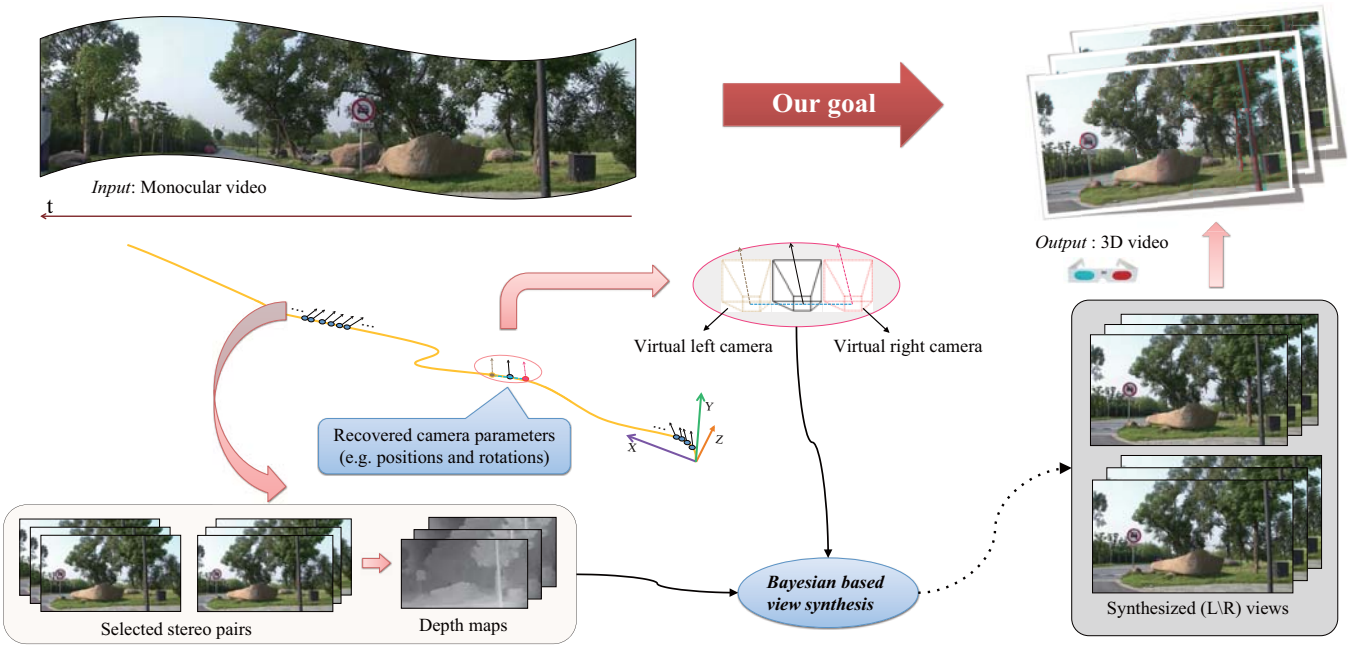


Figure 2. **Framework overview diagram.** Camera parameters for each frame are recovered from the input monocular videos followed by estimation of virtual view camera positions for left-/right- eyes. The stereo pairs are then formed for each frame and calculate the corresponding depth map. Bayesian-based view synthesis method is finally used to generate the image views for estimated virtual cameras. 3D videos can be easily obtained by resembling these synthesized image accordingly.

Table I
NOTATIONS.

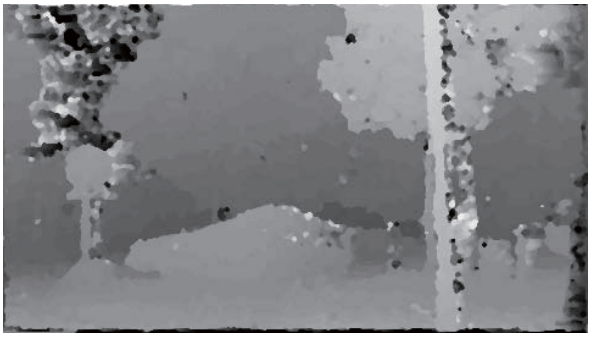
MF_i :	Input original monocular frame sequences, $i = 1...N$.
MP_i :	Estimated camera parameters for MF_i.
MD_i :	Recovered dense depth map for MF_i.
TP_v :	Estimated camera parameters for target stereo left-/right- virtual views.
TF_v :	Synthesized image for virtual view TP_v.
(x, y) :	Superscript, which indicates the pixel location in an image or a depth map, e.g., $\{MF_i^{(x,y)}\}$ refers to the pixel at coordinate (x, y) in frame MF_i, and $\{MD_i^{(x,y)}\}$ is the depth value for pixel $\{MF_i^{(x,y)}\}$.
$fC(TF, MF)$:	shows the pixel correspondences from synthesized stereo images to the original monocular frame, e.g., $fC(TF_v, MF_i)$ shows the correspondence map from TF_v to MF_i, and $fC(TF_v^{(x,y)}, MF_i)$ indicates the corresponding pixel in MF_i for $TF_v^{(x,y)}$.

IV. BAYESIAN-BASED VIEW SYNTHESIS

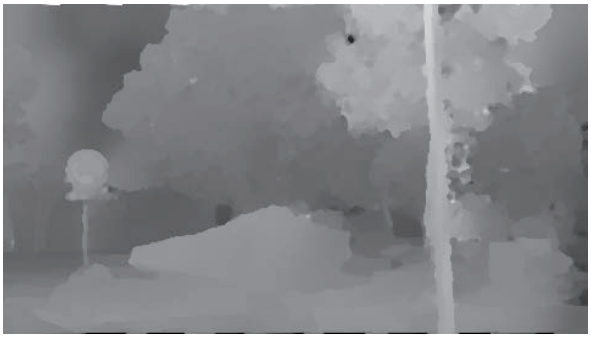
Bayesian-based View Synthesis aims to generate high-quality virtual views with estimated noisy 3D geometry information. There are two observation priors behind the proposed BVS approach. *First*, frames in a video sequence are constrained by color consistency. Since an object appears in at least some consecutive frames in a continuously captured video, a 3D point in the scene can be captured in several consecutive frames with similar color appearances. These pixel correspondences in adjacent frames are called color consistency constraint. *Second*, the synthesized images should be smooth like a natural image. These priors can be fused to eliminate ambiguous geometry information, and improve the quality of the synthesized image. Therefore, BVS is a novel image synthesis method for assuring both color consistency and smooth priors.

To succinctly describe the synthesis algorithm, we define the notations in Table I. Given a monocular video frame sequence MF_i, and inaccurate 3D geometry information MP_i, MD_i, $i = 1...N$, our goal is to synthesize a new image TF_v at a certain virtual view TP_v. The camera parameter for frame i can be denoted as $MP_i = \{K_i, R_i, T_i\}$, where K_i is the intrinsic matrix (e.g., focal length f_i), R_i is the rotation matrix, and T_i is the translation vector.

The depth map MD_i can be considered as another 3D information for frame i, and its value indicates the Z coordinate (depth) of the corresponding 3D points relative to the camera coordinate system. In addition, a point in



(a) before propagation



(b) after propagation

Figure 3. Disparity map results before and after propagation. Whiter parts means greater disparities (closer objects), and values are scaled to gray level for display.

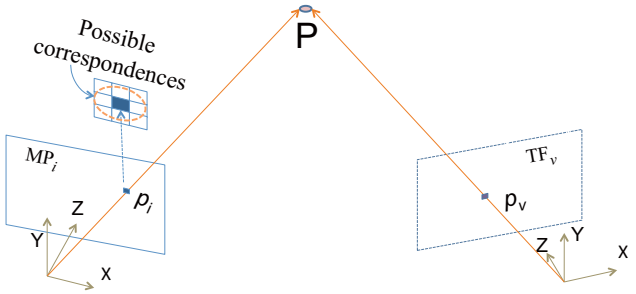


Figure 4. Pixel correspondences between the original monocular frame MF_i and the target virtual view TF_v. Point P can be located in 3D space with view camera parameters and the depth value at pixel p_i. P is then projected to TF_v to get the corresponding point p_v.

3D space can be projected to a view by $p \sim K[R|t]P$, which is called standard projection relationship, shown in Fig. 4. Thus, 3D points corresponding to all pixels in one frame (with known camera parameter and depth map) can be located in the 3D space, and easily mapped to a virtual view as:

$$\lambda_v p_v^h = MD_i(p_i) K_v R_v^T R_i K_i^{-1} p_i^h + K_v R_v^T (T_i - T_v) \quad (1)$$

In this way, the correspondence map between the original frames i and the virtual frame can be built and is also shown by function $fC(TF_v, MF_i)$ in Table I. Since the procedure of generating only one virtual view will be presented in this Section, we will use fC_i instead of $fC(TF_v, MF_i)$ for succinctness.

The objective of BVS is to synthesize the most probable virtual view TF_v based on given available information (e.g., $MF_i, MP_i, \cdots$). Therefore we formulate this problem as a maximization problem in a Bayesian framework and aim to generate the virtual view TF_v, which can maximize:

$$p(TF_v | TP_v, \{MF_i\}, \{MP_i\}, \{MD_i\}), i = 1, ..., N \quad (2)$$

Since the pixel correspondences only depend on the camera and depth information according to Equation (1), we can first construct all correspondence maps $\{fC_i\}$ with 3D geometry information (MP_i, MD_i, TP_v). With the correspondence maps, each pixel in the virtual view can be traced back to input monocular frames $\{MF_i\}$, so the color information can refer to these corresponding pixels, and further synthesize the whole virtual view TF_v. Since MP_i, TF_v, MD_i are already calculated from input video in Section 3, we can treat them as constants and ignore their dependencies on the input frame sequence. These statistical relationships can be summarized in Fig. 5.

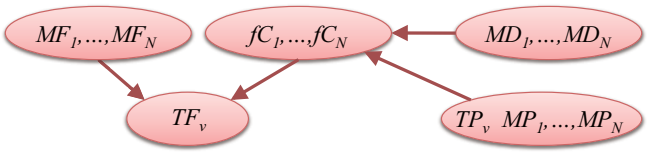


Figure 5. Illustration of statistical dependencies among variables. Although MD_i and MP_i are computed from MF_i, they are independent in the BVS framework, since MD_i and MP_i are assumed to be given and noisy in BVS.

So, We can decompose the above joint probability given by Equation 2 to:

$$p(TF_v | \{MF_i\}, \{fC_i\}) p(\{fC_i\} | TP_v, \{MP_i\}, \{MD_i\}) = \frac{\prod_{i=1}^{N} p(MF_i | TF_v, fC_i) \cdot p(TF_v)}{\prod_{i=1}^{N} p(MF_i)} \prod_{i=1}^{N} p(fC_i | TP_v, MP_i, MD_i) \quad (3)$$

considering the independence of original frames as in [16].

This formulation consists of four parts:

1) $p(MF_i | TF_v, fC_i)$ is the color-consistency prior. That means, the corresponding pixels in monocular frame MF_i and virtual view TF_v are more likely to have similar color texture. Therefore, this prior can be defined as:

$$\begin{aligned} & p(MF_{i, fC_i^{(x,y)}} | TF_v^{(x,y)}, fC_i^{(x,y)}) \\ = & \, exp\left(-e^{-D_{i,v}} \cdot \rho\left(MF_{i, fC_i^{(x,y)}} - TF_v^{(x,y)}\right)\right) \end{aligned} \quad (4)$$

where $D_{i,v}$ is the camera distance between MF_i and TF_v. The function ρ is a robust kernel and we use $\rho(x) = |x|$ in this work.

2) As mentioned before, the computed images should

be smooth like natural images. This is achieved by defining a prior on the synthesized virtual view, $p(TF_v)$ as:

$$p(TF_v) = \prod_{(x,y)} e^{-\left|TF_v^{(x,y)} - AvgN\left(TF_v^{(x,y)}\right)\right|} \quad (5)$$

where $AvgN(\cdot)$ means the average value over 8-connected neighborhoods of pixel location (x, y).

3) $p(fC_i|TP_v, MP_i, MD_i)$ is the likelihood of the *computed correspondence* (CC) to be *true correspondence* for given TP_v, MP_i, and MD_i. For example, in Fig. 4, p_v is CC for p_i. If TP_v, MP_i, and MD_i are precise, CC is the *true correspondence*.

Since the computed TP_v, MP_i, and MD_i could be noisy, we consider the 8-connected neighborhood of CC to be the candidate preferences of CC. When we index the 8-connected neighborhood of CC including CC by $j = 0, \cdots, 8$, $j = 4$ indicates CC. Thus, $p(fC_i) = \sum_{j=0}^{8} p(fC_i^{(j)})$. The possibilities for these possible correspondences are defined as:

$$p(fC_i^{(j)}|TP_v, MP_i, MD_i) = e^{-\alpha_j} \quad (6)$$

where $\alpha_j = \theta + 30$ for $j \neq 4$, $\alpha_j = \theta$ for $j = 4$, and $\theta = 10$ in our experiments since CC has higher probability of being true correspondence than its neighborhood for given TP_v, MP_i, and MD_i.

4) $p(MF_i)$ is the prior on the input monocular frames. We have no prior knowledge about the input video so we assume a uniform prior in this work and ignore this term.

Finally, the objective function can be approximated by choosing only the best j, which maximizes the joint probability with color consistency prior and it is rewritten as:

$$\begin{aligned} &\prod_{i=1}^{N} p(MF_i|TF_v, fC_i)p(fC_i|TP_v, MP_i, MD_i) \cdot p(TF_v) \\ &\approx \prod_{i=1}^{N} max_j[p(MF_i|TF_v, fC_{i,j}) \\ &\times p(fC_{i,j}|TP_v, MP_i, MD_i)] \cdot p(TF_v), \end{aligned} \quad (7)$$

although different cases with all possible j should sum up for the objective function theoretically.

In the implementation, we minimize the negative log of the objective probability function, and get the energy formulation objective:

$$\begin{aligned} &\sum_{i=1}^{N} \sum_{(x,y)} e^{-D_{i,v}} min_j(\rho(MF_{i,fC_{i,j}^{(x,y)}} - TF_v^{(x,y)}) + \alpha_j) \\ &+\lambda \sum_{(x,y)} (|TF_v^{(x,y)} - AvgN(TF_v^{(x,y)})|) \end{aligned} \quad (8)$$

where λ determines the degree of smoothness constraint that should be imposed on the synthesized image. We set $\lambda = 0.1$.

Optimization: Although minimization of the energy function could be performed by global optimization strategies such as a simulated annealing, attaining a global optimum is time consuming, thus preventing one from synthesizing many frames for a video. Since the possibilities for each correspondence are only a few, similar to [17], we implement a variant of iterated conditional modes (ICM) algorithm to obtain an approximate solution. We alternately optimize the color-consistency prior (first term) and virtual view prior (second term). For the initial estimation V^0, we simply choose the most likely correspondences $(j = 4)$ for each pixel, and the synthesized results can be obtained by weighted average of correspondences from all frames. The optimum solution T^k of the second term on current estimation can be obtained by mean filter. A median filter can also be used instead to avoid outliers and blur sharp boundaries. The input V^{k+1} for next iteration can be set as the linear combination of V^k and T^k:

$$V^{k+1} = \left(V^k + \lambda T^k\right) \times (1 + \lambda)^{-1} \quad (9)$$

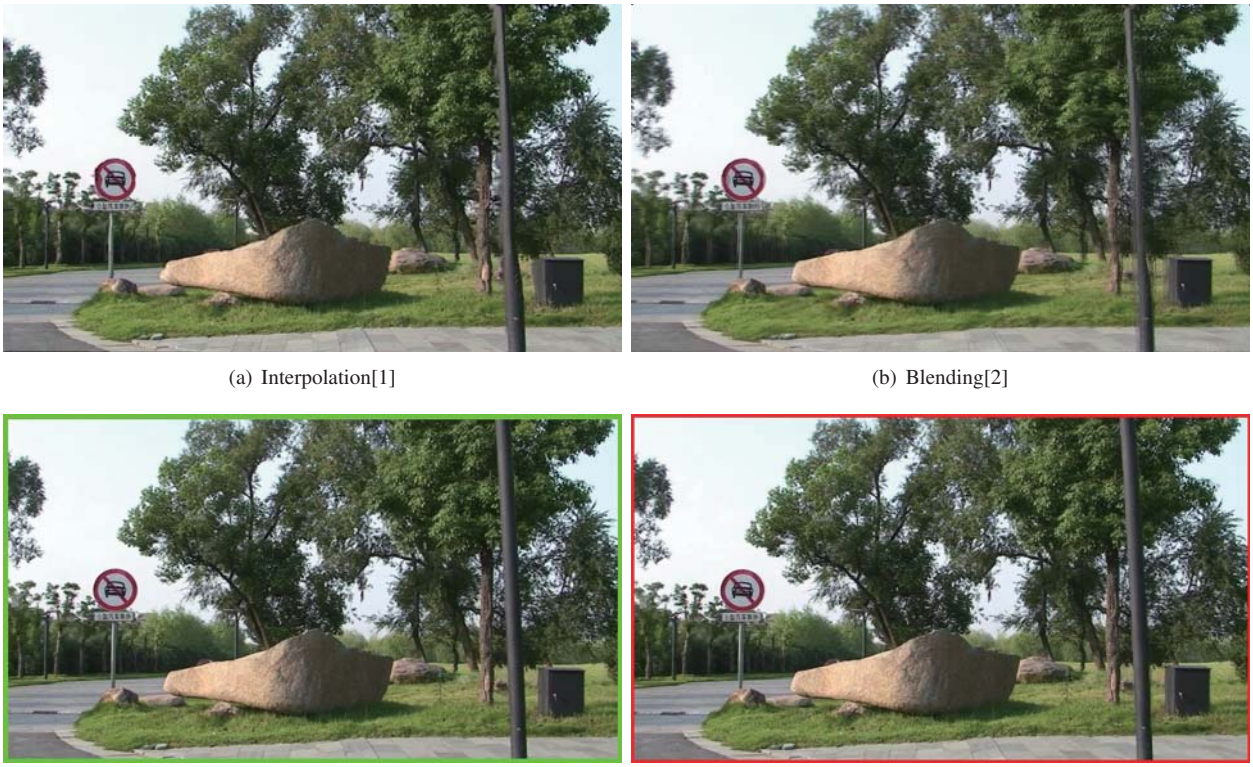
Generally after a few iterations $(5 \sim 10)$, the algorithm converges and we synthesize new virtual views by estimated parameters.

(a) Input (b) 3D frames by our system

Figure 7. Stereoscopic synthesis results. Input samples are zoomed out to save space. From top to bottom are: *pavilion, stele, pride*. **Please view this figure on the screen, and use archetypical 3-D glasses to view 3D results.** More 3D results can be found in the supplementary material.

V. Experiments

To evaluate the performance of the proposed method, we have conducted experiments on several challenging sequences. Two are from public data sets [18], *road, lawn*. These two videos have very complex scenes with many objects located at different depth layers. We also produced two videos ourselves with a casual camera, *pavilion, stele*. Another one is a clip from the movie "Pride and Prejudice",



(a) Interpolation[1] (b) Blending[2]

(c) Ground truth (d) Our result

Figure 6. Synthesized results by all 3 compared methods. Compared to the ground truth, the result of interpolation has severe blur at objects boundaries, while the result of the blending method suffers from ghosting. All figures are best viewed at at least 600% zoom on the screen. Some close-up images also can be seen in Fig. 1. More examples can be found in Fig. 10.

called *pride* for short. Some example frames of these videos can be found at Fig. 7(a).

We compare our view synthesis algorithm with two recent works [1], [2]. Zhang et al. [1] employ cubic-interpolation to fill the holes generated by parallax, and Zitnick and Kang [2] blend the virtual views generated by two closest camera frames to synthesize the final virtual view.

Ground Truth: Since the ground truth of virtual views is impossible to obtain, existing views from the original sequence are selected as virtual views. Each method aims to produce the selected virtual views from other frames in a monocular video.

Quantitative Evaluation: We perform quantitative comparisons for these methods [1], [2] on all testing video sequences, *Road, Lawn, Pavilion, Stele, Pride*. For each video, we select one frame out of every 10 frames to be synthesized by all methods and evaluate the performance. The results are measured by Peak Signal-to-Noise Ratio (PSNR) and Structural SIMilarity (SSIM) [19]. As can be seen in Fig. 8, our method consistently achieves higher scores than the other two methods with smaller or comparable variance on all testing videos in terms of PSNR and SSIM. Interpolation generally outperforms blending, which indicates that there is a significant amount of noise in the computed 3D geometry

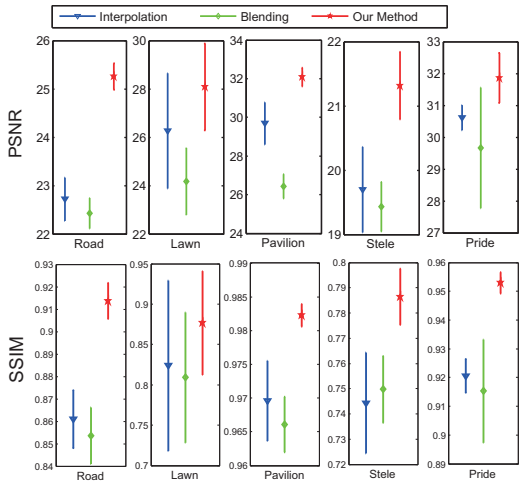


Figure 8. The PSNR and SSIM score and variance comparison for interpolation [1], blending [2], and our method on five testing videos. Our method outperforms the other two methods consistently on all testing videos. Blending obtains the worse results, since it suffers greatly from misalignments.

information. This result also proves the effectiveness and necessity of our method in fusing multiple warped results.

Qualitative Evaluation: Generally, the interpolation method [1] can produce good results, but quality degrades significantly at object boundaries as can be seen in Fig. 6 and 10. This is so because there are larger holes to be synthesized due to depth parallax whenever an object is closer to the camera. Since the interpolation method uses the surrounding areas to fill such holes, it unavoidably produces severe blur effects.

The blending method [2] blends two warped results from two closest frames to complement the missing information in the occluded areas. However, it requires precise 3D geometry information to enable perfect alignment of two warped views. This method cannot produce satisfactory results when the estimated camera parameters and depth map are noisy. As can be seen in Fig. 6(b), the generated results suffer from blur as compared to the results of interpolation and our method.

Although our method also fuses the information from multiple warped views to complement the missing information after warping, it does not suffer from the misalignment problem that originates from noisy estimation of the camera parameters because our proposed BVS takes this noisy estimation in particularly into account.

Some artifacts still occur in some difficult parts, such as the pillars in Fig. 1 due to the significant changes in illumination on the pillars. The result demonstrates our proposed method can properly fuse misaligned results warped by inaccurate 3D geometry information, and thus synthesize the best quality view. More results and synthesized 3D videos can be found in our supplemental material.

VI. Conclusion and Future Work

In this paper, we proposed an automatic stereoscopic video synthesis framework to convert casually captured 2D videos of complex scenes into 3D videos without ruining actual 3D geometry scene structures. More importantly, a novel Bayesian-based view synthesis (BVS) approach is proposed to address the challenges in synthesizing high-quality virtual views with inaccurate 3D geometry information. Extensive experiments on various videos suggest that our system outperforms the state-of-the-art methods in generating virtual views and 3D videos.

Since none of the existing automatic methods including ours can deal with independently moving objects with non-static scene, stationary camera and zoom in/zoom out camera, we will further improve our system to handle such cases.

References

[1] L. Zhang, C. vazquez, and S. Knorr, "3d-tv content creation: Automatic 2d-to-3d video conversion." *IEEE Trans. on Broadingcasting*, 2011.

[2] C. L. Zitnick and S. B. Kang, "Stereo for image-based rendering using image over-segmentation." *IJCV*, 2006.

[3] C. Fehn, "Depth-image-based rendering (dibr), compression and transmission for a new approach on 3d-tv." *SPIE*, 2004.

[4] P.V.harman, "Home-based 3d entertainment–an overview." *ICIP*, 2000.

[5] M. Guttmann, L. Wolf, and D. Cohen-Or, "Semi-automatic stereo extraction from video footage." *ICCV*, 2009.

[6] X. Cao, Z. Li, and Q. Dai, "Semi-automatic 2d-to-3d conversion using disparity propagation." *IEEE Trans. on Broadingcasting*, 2011.

[7] G. Zhang, W. Hua, X. Qin, T. Wong, and H. Bao, "Stereoscopic video synthesis from a monocular video." *IEEE Trans. on Visualization and Graphics*, 2007.

[8] M. Pollefeys, L. V. Gool, M. Vergauwen, F. Verbiest, K. Cornelis, J. Tops, and R. Koch, "Visual modeling with a hand-held camera." *IJCV*, 2004.

[9] S. E. Chen and L. Williams, "View interpolation for image synthesis." *SIGGRAPH*, 1993.

[10] S. J. Gortler, R. Grzeszcauk, R. Szeliski, and M. F. Cohen, "The lumigraph." *SIGGRAPH*, 1996.

[11] M. Levoy and P. Hanrahan, "Light field rendering." *SIGGRAPH*, 1996.

[12] N. Snavely, S. M. Seitz, and R. Szeliski, "Photo tourism: Exploring photo collections in 3d." *SIGGRAPH*, 2006.

[13] M. Park, S. Kashyap, R. Collins, and Y. Liu, "Data-driven mean-shift belief propagation for non-gaussian mrfs." *CVPR*, 2010.

[14] D. G. Lowe, "Object recognition from local scale-invariant features," *ICCV*, pp. 1150–, 1999.

[15] G. Sourimant, "Depth maps estimation and use for 3dtv," INRIA Rennes Bretagne Atlantique, Rennes, France, Technical Report 0379, feb 2010.

[16] C. liu and D. Sun, "A bayesian approach to adaptive video super resolution," *CVPR*, 2011.

[17] A. Fitzgibbon, Y. Wexler, and A. Zisserman, "Image-based rendering using image-based priors." *IJCV*, 2005.

[18] G. Zhang, J. Jia, T. Wong, and H. Bao, "Consistent depth maps recovery from a video sequence." *TPAMI*, 2009.

[19] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *TIP*, vol. 13, no. 4, pp. 600–612, 2004.

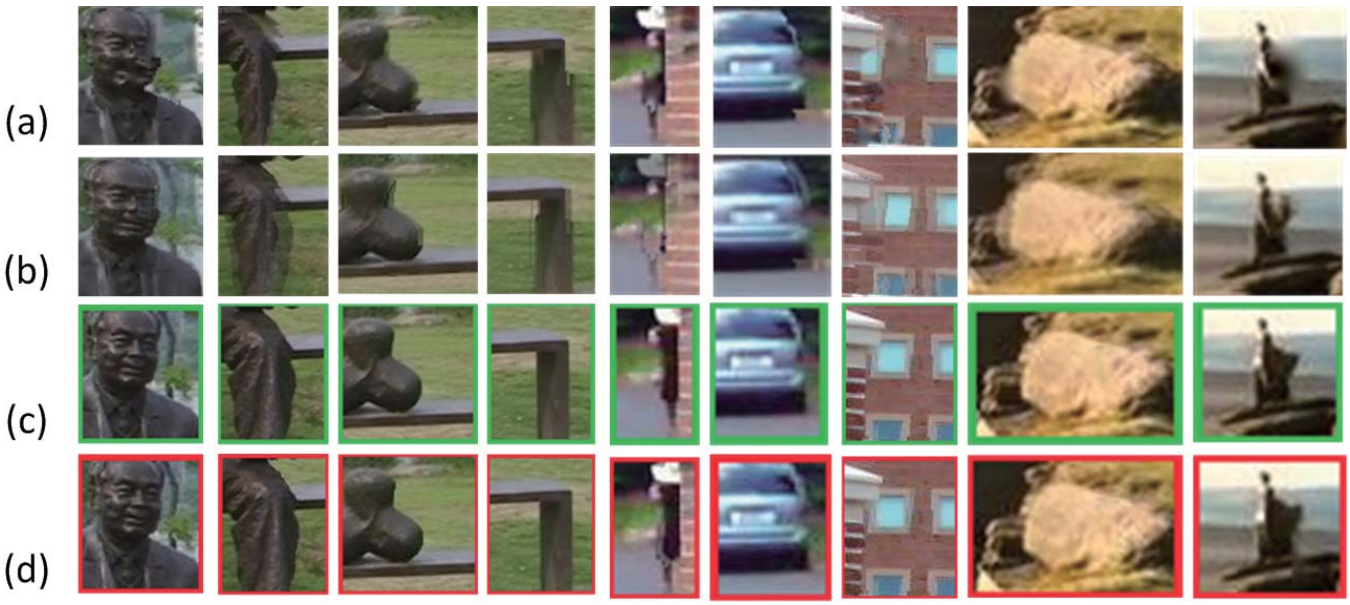


Figure 9. (a) Interpolation[1] (b) Blending[2] (c) Ground truth (green box) (d) Our method (red box). Most IBR methods suffer from inaccurate estimated 3D geometry information, especially at object boundaries. For zoom out version, refer to Figure 10.

(a) Interpolation[1] (b) Blending[2] (c) Ground truth (d) Our result

Figure 10. Synthesis results by all 3 compared methods. Compared to the ground truth, result of interpolation has severe blur at objects boundaries, while result of blending method suffers from ghosting. All figures are best viewed at least 600% zoom on screen. Some close-up images are also can be seen in Figure 9.