Building a Videorama with Shallow Depth of Field

Soonmin Bae and Hao Jiang
Boston College

Abstract

This paper presents a new automatic approach to building a videorama with shallow depth of field. We stitch the static background of video frames and render the dynamic foreground onto the enlarged background after foreground/background segmentation. To this end, we extract the depth information from a two-view video stream. We show that the depth cues combined with color cues improve segmentation. Finally, we use the depth cues to synthesize the shallow depth of field effects in the final videorama. Our approach stabilizes the camera motion as if the video was captured from a static camera and improves the visual quality with the increased field of view and shallow depth of field effects.

1. Introduction

Although video cameras are widely available, creating a professional-looking video has not been common yet. It is difficult to capture good footage that is well composed, beautifully lit, and endearing. Casual videos usually follow the main subjects with unplanned shaky camera motions. As a result, the main subject is not in the middle of the frame, but a cluttered background is.

High-frequency camera shakes are often removed by video stabilization. We want to extend video stabilization to low-frequency camera motions and generate a *videorama*. Videorama presents all the video frames as if the video was captured from a fixed camera while including all the fields of view. As a result, videorama has an elongated field of view. To this end, we stitch the static background and render the dynamic foreground onto the enlarged background.

While automatic image stitching has been studied extensively [15, 4], automatic video stitching is not available. Complex motions of the camera and foreground objects are the main challenge for video stitching. Pritch et al. [12] present a panoramic stroboscopic synopsis , but they assume a simple motion of the camera and object. Gleicher and Liu [6] improve the camera motion by transforming each frame, but they inpaint the background instead of stitching the background. Liu et al. [10] stitch video frames, but their goal is to build a panorama instead of a videorama.

In addition to stitching the video frames, we provide shallow depth of field effects to blur cluttered background. To generate a visually pleasing results, the depth of field effects should be consistent with the actual depth. To this end, we estimate depth using a two-view video stream. The additional dimensional information in the Light field cameras [13, 16, 1, 11] is used in video stabilization and depth of field manipulation. In this paper, we use the depth to separate foreground and background and to render the final result with shallow depth of field. Zhang et al. [18] implement the depth of field effects in videos, but they do not stitch the background since they require a large camera motion.

In this paper, we propose an automatic approach to generate a videorama with shallow depth of field. We segment foreground/background, stitch the static background, and render the dynamic foreground on the stitched background. We use the depth extracted from a two-view video stream for the segmentation and for the final shallow depth of field rendering. It is natural to use the depth to separate the foreground, since the foreground is by definition the part of the scene closer to the camera than the background. We label the pixels that are close to the camera and with different motion from the background as foreground.

Moving object extraction from a video has been a great interest of the Computer Vision community. Many of them [14, 17] assume a static camera. Zhang et al. [19] use a structure from motion (SfM) method to extract a moving object, but they assume camera translation. Since we use two-view video streams, we do not require camera translation. Liu and Gleicher [9] assume the background colors are different from the foreground ones. By combining the color cues with the depth, our method works for the cases where the background and foreground colors are similar. Kolmogorov et al. [7] combine stereo, color, and contrast for segmentation. They assume rectified inputs and depth discontinuities, while we deal with unrectified inputs and do not require depth discontinuities.

2. Our Approach

We consider the moving foreground as regions that are close to the camera and move in some frames. We assume the camera motion is mostly panning and the backgrounds can be stitched with homography warping.

We extract depth from a two-view video stream and build rough backgrounds using a median filter. We perform segmentation in two steps. In our initial segmentation, we use the depth and the color difference between the current frame and the background to separate the foreground and background. Our final segmentation uses the color and depth cues learned from the initial segmentation results to extract the moving foreground. We stitch the background to generate an elongated background and blend the dynamic foreground with it. The depth information is used to synthesize shallow depth of field effects in the final results.

2.1. Depth estimation

We estimate the depth $D_t(r)$ of each pixel r by applying SfM to each pair of frames L_t and R_t at time t. SfM outputs a sparse depth map. We propagate the sparse depth map to the whole image using optimization [8, 2]. The second columns of Fig. 4 and 5 show our depth maps. Darker pixels are closer.

2.2. Initial moving object extraction

We identify moving objects by examining the depth and the color difference between the current frame and background. Previous work assumes a static camera or a known background [12, 14]. However, we do not have a known background, and a simple median filter does not work on videos from moving cameras. To remove the camera motion, we apply homography warping before taking a temporal median. For each frame t, we take a median of the warped frames which overlap with the frame t by more than 90%.

After constructing the background B_t, we perform moving object extraction using graph cuts [3]. Let R_t be the set of pixels in the current frame of the right camera and N_t be the set of 8-connected adjacent pixel pairs. A labeling f assigns $1(fg)$ or $0(bg)$ to each pixel $r \in R_t$. To obtain a binary image labeling f, we minimize a Gibbs energy $E(f)$:

$$E(f) = \sum_{r \in R_t} E_{data}(f_r) + \sum_{(r,s) \in N_t} E_{smooth}(f_r, f_s) \tag{1}$$

Our smoothness energy term is based on the normalized correlation between the colors of two pixels [8]:

$$E_{smooth}(f_r, f_s) \propto 1 + \frac{1}{\sigma_r^2}(I(r) - \mu_r)(I(s) - \mu_r) \tag{2}$$

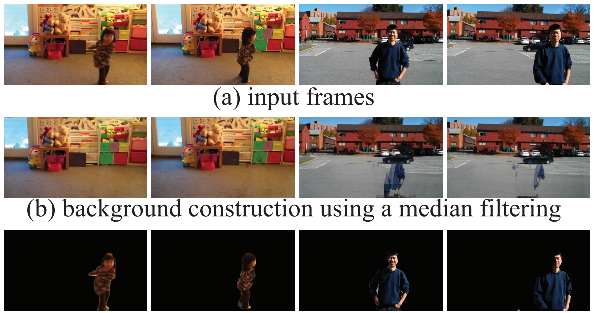
where μ_r and σ_r are the mean and variance of the colors in the neighboring pixels around r.

The data term evaluates the likelihood of each pixel belonging to the foreground or background. We assume that the color differences of the moving foreground objects from the background $V_t(r) = \| I_t(r) - B_t(r) \|$ are large and the moving foreground objects are close to the camera, i.e., $D_t(r)$ is close to 0. The mean and variance of D can be modeled for the foreground since the depth of foreground pixels are similar to each other, and the mean and variance for V can be modeled for the background since V is close to 0 in the background pixels. We compute the mean μ_d and variance σ_d of D for the pixels whose D values are less than the $10th$ percentile, and the mean μ_v and variance σ_v of V are for the pixels whose V values are less than the $90th$ percentile. Basically, we are assuming that approximately 90% of pixels are the background pixels. Our data term $E_{data}(f_r)$ is formulated as follows:

$$E_{data}(bg) = \quad -\log(e^{-\frac{\|V-\mu_v\|^2}{2\sigma_v^2}} \cdot (1 - e^{-\frac{\|D-\mu_d\|^2}{2\sigma_d^2}})) \tag{3a}$$

$$E_{data}(fg) = \quad -\log((1 - e^{-\frac{\|V-\mu_v\|^2}{2\sigma_v^2}}) \cdot e^{-\frac{\|D-\mu_d\|^2}{2\sigma_d^2}}) \tag{3b}$$

The above model extracts moving objects correctly when the median filtering results are accurate. However, the median filtering assumes that the background appears more than the foreground, and often the foreground object does not move and occupies the pixel more than the background. In Fig. 1, the background in the first two columns are built well and the foreground objects are correctly extracted, while the background in the last two columns are not correct. The foreground in the third column is correctly detected thanks to our depth cues, while a part of the foreground object in the last column is missing. To overcome this false detection, we learn color cues from the extracted foreground and background and perform our final segmentation.


(a) input frames

(b) background construction using a median filtering

(c) initial foreground extraction

Figure 1. If the foreground objects stay still, a median filter cannot generate a correct background. Our depth cues help recovering the foreground even with an incorrect background.

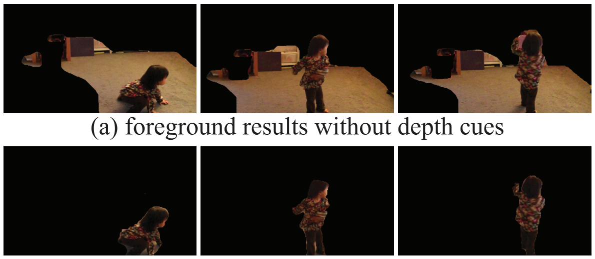
2.3. Final segmentation using color and depth cues

We train a Gaussian Mixture Model (GMM) on both foreground and background colors, and their depth maps as well. We train the foreground from the top 10% frames whose initial foreground sizes are small, and train the background from the top 10% frames whose initial background sizes are small. This prevents from learning from the false detection. The likelihood of a pixel belonging to the foreground or background is modeled as follows:

$$p(I_r|l) = \sum_{k=1}^{K_l} p_{l,k} G((I_r, D_r)|\mu_{l,k}, \Sigma_{l,k}) \qquad (4)$$

where $l \in \{fg, bg\}$ and $G(\cdot)$ is a Gaussian component. $(p_{l,k}, \mu_{l,k})$ represents the prior and the mean color and depth. $\Sigma_{l,k}$ is the covariance matrix of the k_{th} component of GMM, and forms as $\begin{pmatrix} \Sigma_{l,k,c} & 0 \\ 0 & \sigma_{l,k,d} \end{pmatrix}$, where c stands for color and d stands for depth. We use 12 for K_{bg} and 3 for K_{fg}. We solve a new labeling problem with the following data term using graph cuts.

$$E_{data}(l) = -\log p((I_r, D_r)|l) \qquad (5)$$



(a) foreground results without depth cues

(b) our foreground results with depth cues

Figure 2. When the foreground and background colors are similar, color cues are not sufficient to separate the foreground from the background. Our segmentation employs the depth cues and can generate correct segmentation results.

Since we use the depth cues in addition to the color cues, we can separate the foreground and background even when they have very similar color distributions. Figure 2 shows the comparison. The third columns of figure 4 and 5 show more of our segmentation results.

2.4. Videorama with shallow depth of field synthesis

To generate a videorama, we stitch only the background of each frame by taking a median after homography warping. We blend the foreground with the stitched background using a multi-band blending method [5] and apply a spatially varying blur according to our depth map.

3. Results

Figure 3 shows the background and depth maps stitched. They correctly capture the static background with an elongated field of view.

After our foreground and background composite, the videorama looks as if the video was captured from a fixed camera with a large field of view. In addition, our videorama blurs the cluttered background while keeping the foreground sharp. This simulates the shallow depth of field effects. Figure 4 and 5 present our videorama results in addition to the original frames, depth maps, and our segmentation results.

4. Conclusions

We have presented an automatic approach to generating a videorama with shallow depth of field. We extract the depth using a two-view video streams and segment foreground/background using the depth and color cues. We show that using the depth cues is intuitive and improves segmentation. Our approach stabilizes the camera motion by fixing a virtual camera location and improves the visual quality by increasing the field of view and by synthesizing shallow depth of field effects.

References

[1] E. Adelson and J. Wang. Single lens stereo with a plenoptic camera. *PAMI*, 1992.

[2] S. Bae and F. Durand. Defocus magnification. *Computer Graphics Forum*, 2007.

[3] Y. Boykov and V. Kolmogorov. An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *PAMI*, 2004.

[4] M. Brown and D. G. Lowe. Automatic panoramic image stitching using invariant features. *IJCV*, 2007.

[5] P. J. Burt and E. H. Adelson. A multiresolution spline with application to image mosaics. *ACM Trans. Graph.*, 2(4):217–236, 1983.

[6] M. L. Gleicher and F. Liu. Re-cinematography: Improving the camerawork of casual video. *ACM Trans. Multimedia Comput. Commun. Appl.*, 5(1):1–28, 2008.

[7] V. Kolmogorov, A. Criminisi, A. Blake, G. Cross, and C. Rother. Probabilistic fusion of stereo with color and contrast for bilayer segmentation. *PAMI*, 2006.

[8] A. Levin, D. Lischinski, and Y. Weiss. A closed form solution to natural image matting. *PAMI*, 2008.

[9] F. Liu and M. Gleicher. Learning color and locality cues for moving object detection and segmentation. In *CVPR*, 2009.

[10] F. Liu, Y. hen Hu, and M. Gleicher. Discovering panoramas in web videos. In *ACM Multimedia*, 2008.

[11] R. Ng, M. Levoy, M. Bredif, G. Duval, M. Horowitz, and P. Hanrahan. Light field photography with a handheld plenoptic camera. In *Technical Report CSTR 2005-02*. Stanford University Computer Science, 2005.

[12] Y. Pritch, A. Rav-Acha, and S. Peleg. Nonchronological video synopsis and indexing. *PAMI*, 2008.

[13] B. M. Smith, L. Zhang, H. Jin, and A. Agarwala. Light field video stabilization. In *ICCV*, 2009.

[14] J. Sun, W. Zhang, X. Tang, and H.-Y. Shum. Background cut. In *ECCV*, 2006.

[15] R. Szeliski. Image alignment and stitching: a tutorial. *Found. Trends. Comput. Graph. Vis.*, 2(1):1–104, 2006.

[16] B. Wilburn, N. Joshi, V. Vaish, E.-V. Talvala, E. Antunez, A. Barth, A. Adams, M. Horowitz, and M. Levoy. High performance imaging using large camera arrays. *ACM Transactions on Graphics*, 2005.

[17] T. Yu, C. Zhang, M. Cohen, Y. Rui, and Y. Wu. Monocular video foreground/background segmentation by tracking spatial-color gaussian mixture models. In *WMVC '07: Proceedings of the IEEE Workshop on Motion and Video Computing*, page 5, 2007.

[18] G. Zhang, Z. Dong, J. Jia, L. Wan, T.-T. Wong, and H. Bao. Refilming with depth-inferred videos. *IEEE Trans. on Visu. and Comp. Graph.*, 2009.

[19] G. Zhang, J. Jia, W. Xiong, T.-T. Wong, P.-A. Heng, and H. Bao. Moving object extraction with a hand-held camera. In *ICCV*, 2007.

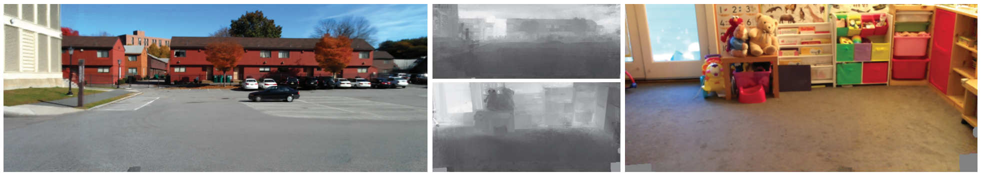
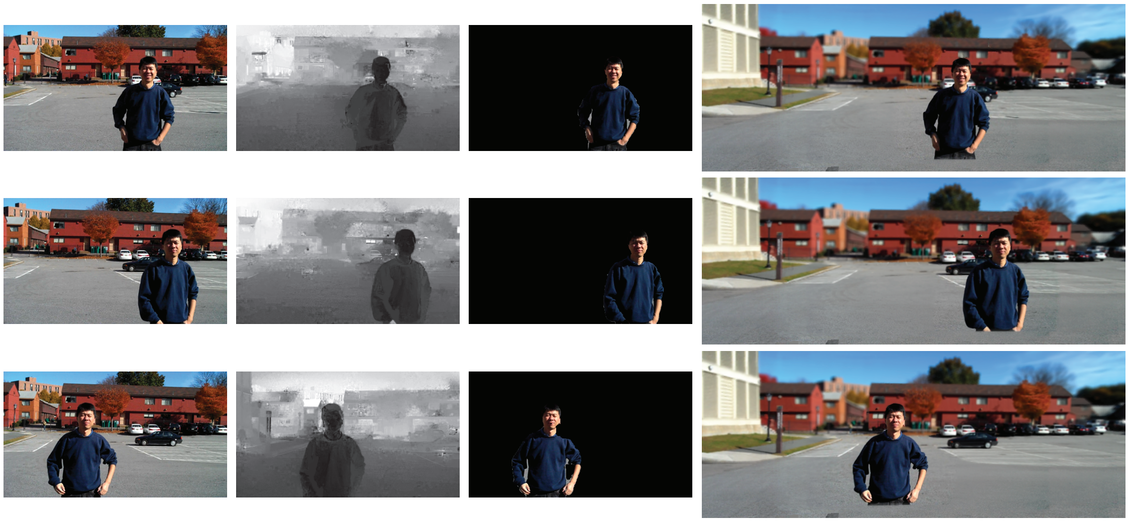
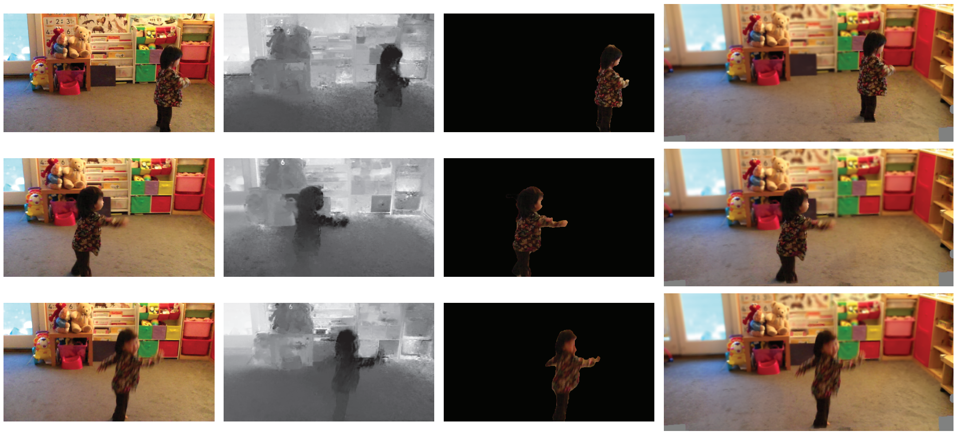


Figure 3. Backgrounds and depth maps stitched. In our depth maps, darker pixels are closer.



(a) input frames (b) depth maps (c) our segmentation results (d) our final videorama frames with shallow depth of field

Figure 4. The input frames (a) present unplanned camera motion and sharp and cluttered background, while our videorama result (d) looks as if the video was captured from a fixed camera with a large field of view and shallow depth of field.



(a) input frames (b) depth maps (c) our segmentation results (d) our final videorama frames with shallow depth of field

Figure 5. Our videorama result (d) has a fixed large field of view and shallow depth of field, while the input frames (a) have unplanned camera motion and sharp and cluttered background.