STEREOSCOPIC CONTENT PRODUCTION OF COMPLEX DYNAMIC SCENES USING A WIDE-BASELINE MONOSCOPIC CAMERA SET-UP

Jean-Yves Guillemaut, Muhammad Sarim and Adrian Hilton

Centre for Vision, Speech and Signal Processing, University of Surrey, Guildford, Surrey, UK

ABSTRACT

Conventional stereoscopic video content production requires use of dedicated stereo camera rigs which is both costly and lacking video editing flexibility. In this paper, we propose a novel approach which only requires a small number of standard cameras sparsely located around a scene to automatically convert the monocular inputs into stereoscopic streams. The approach combines a probabilistic spatio-temporal segmentation framework with a state-of-the-art multi-view graph-cut reconstruction algorithm, thus providing full control of the stereoscopic settings at render time. Results with studio sequences of complex human motion demonstrate the suitability of the method for high quality stereoscopic content generation with minimum user interaction.

Index Terms— Stereoscopic rendering, 3D video, multiple view reconstruction, segmentation.

1. INTRODUCTION

Recent years have seen a dramatic increase in demand for stereoscopic content such as 3D movies. In coming years, the demand is likely to further increase as 3DTV technology starts to reach maturity. Stereoscopic video production is challenging in many respects. Firstly, it induces a significant overhead in equipment costs, doubling camera requirements and introducing a need for additional pieces of hardware to mount and control camera pairs. Secondly, stereoscopic settings such as the interaxial distance and the convergence distance must be set at capture time and are difficult to edit afterwards. Finally, stereo post production requires more sophisticated techniques able to jointly edit left and right streams.

In this paper, we propose an alternative approach where stereo cameras are replaced by a small number of easier to control conventional (monoscopic) cameras sparsely distributed around the scene and used to automatically synthesise stereoscopic output, thereby eliminating the need for setting stereo parameters at capture time and providing full control of these parameters at render time. This presents a major challenge due to the wide baseline configuration and the large volume of data that must be automatically processed.

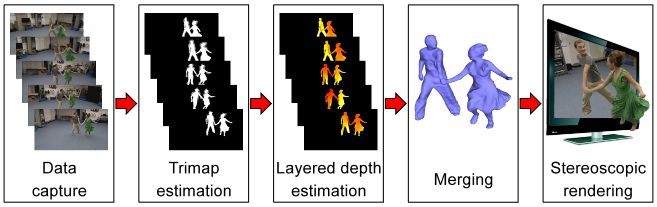


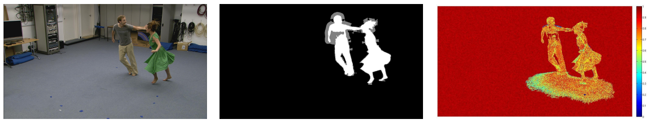
Fig. 1. Pipeline overview.

There has been two main strands of research concerned with 3D video production from monocular cameras. 2D to 3D conversion techniques concentrate on generating stereo output from a single monocular input. Such a problem is under constrained without additional scene assumptions or user input. For static scenes, structure-from-motion techniques have been successful at estimating a sparse scene structure which, although not sufficient on its own, can produce convincing results when combined with image based rendering techniques [1]. For dynamic scenes, single camera techniques usually lack accuracy and temporal stability unless significant user interaction is provided.

In contrast, multi-camera technique offer a more reliable alternative without a need for manual interaction. In their seminal paper [2], Narayanan et al. used a set of 51 cameras distributed on a hemispherical dome to compute 3D models of a dynamic scenes which can then be rendered from any viewpoint with an accuracy similar to that of the input cameras. Since then, many approaches have been proposed for 3D reconstruction [3] and have been applied to free-viewpoint video synthesis [4, 5]. These techniques remain generally limited to a narrow baseline camera set-up or require a very large number of cameras.

Our approach, in contrast, only requires a small number of cameras (usually eight) in a wide-baseline configuration. We propose a full pipeline (see Fig. 1) based on a probabilistic framework for propagating segmentation across time and cameras, combined with a state-of-the-art graph-cut layered depth inference framework. The method is able to produce high quality stereoscopic sequences of complex human motion with user interaction only required on a single key frame.

This work was supported by TSB/EPSRC projects “i3Dlive: interactive 3D methods for live-action media” (TP/11/CII/6/I/AJ307D, TS/G002800/1).



(a) (b) (c)

Fig. 2. Example of input image (a) and its estimated trimap (b) and confidence map (c).

2. PROPOSED APPROACH

The problem considered is the conversion of a set of monoscopic video streams, captured from a set of synchronised and calibrated cameras, into corresponding stereoscopic streams. In this paper, the focus is on foreground processing; full scene rendering is obtained by compositing the foreground with a real or a virtual background during post processing. To solve this problem, we must perform two key operations, namely for each input image we must (1) identify the foreground pixels (segmentation) and (2) estimate their depth (reconstruction). The proposed approach (illustrated in Fig. 1) starts by estimating a coarse segmentation (trimap propagation) which is then jointly refined with depth estimation in a view-dependent manner (layered depth estimation), before global refinement which maximises multi-view consistency (merging) for final stereoscopic rendering.

2.1. Trimap propagation

Automatic foreground segmentation in a wide-baseline camera set-up is a challenging task due to variations in colour distributions across cameras and time. Instead of attempting exact segmentation, we compute a more robust coarse segmentation known as a trimap [6] which will be refined in subsequent stages based on multi-view information. A trimap (see Fig. 2(b)) is a partition of the image into three regions, namely, definite foreground, definite background and unknown. Trimap generation is often a manual process which precedes matting and would be prohibitively expensive in the case of a multi-camera set-up. To automate this process, we propose a Bayesian inference framework which extends previous work in [7] by propagating trimaps across space and time from a small set of manually segmented key-frames (1 or 2) in a single view.

The framework relies on colour models initialised using the key-frames and automatically updated as trimaps are sequentially propagated across cameras and time and more information is gathered about the colour distributions. Two types of models are considered: a local background colour model $\mathcal{L}_i^{\mathrm{B}}$ for each camera i consisting of a single Gaussian distribution at each pixel, and global colour models $\mathcal{G}^{\mathrm{F}}$ and $\mathcal{G}^{\mathrm{B}}$ for foreground and background respectively defined as multi-variate Gaussian distributions $N(\mu_k, \Sigma_k)$ weighted with a confidence value λ_k accounting for the component uncertainty. The estimation process consists of two stages: trimap estimation and confidence map estimation (see Fig. 2).

2.1.1. *Trimap label estimation*

At each pixel, we seek the trimap label maximising the posterior probability given the learnt colour models. The posterior probability of a pixel $\boldsymbol{p}$ with colour $\boldsymbol{I_p}$ belonging to the k^{th} component of a model with mean μ_k and covariance Σ_k is

$$P(\mu_k, \Sigma_k \mid \boldsymbol{I_p}) = P(\boldsymbol{I_p} \mid \mu_k, \Sigma_k) P(\mu_k, \Sigma_k) / P(\boldsymbol{I_p}), \quad (1)$$

where $P(\mu_k, \Sigma_k)$ is the prior for the cluster k given by its confidence value λ_k and $P(\boldsymbol{I_p})$ is the prior for pixel $\boldsymbol{p}$ which is independent of the models and can be ignored. After having estimated posterior probabilities corresponding to foreground and background hypotheses, inferential statistics based on the χ^2 test with 95% certainty are used to validate foreground and background hypotheses and automatically classify less likely hypotheses as unknown.

2.1.2. *Confidence map estimation*

Pixel misclassifications are unavoidable due to overlapping foreground and background distributions and scene regions that were not visible in the key frames. In order to alleviate their effect, we associate a confidence level $C_{\boldsymbol{p}}$ to each trimap pixel assignment. Confidence values are used to prevent drift in global models by weighting new samples accordingly when incorporated into the global foreground and background models. The confidence level of pixel assignment is derived from its posterior probability $P(\mu_k, \Sigma_k \mid \boldsymbol{I_p})$ obtained from Eq. (1) and the confidence of the corresponding component k in the model λ_k according to the formula

$$C_{\boldsymbol{p}} = \lambda_k P(\mu_k, \Sigma_k \mid \boldsymbol{I_p}). \quad (2)$$

2.2. Layered depth estimation

Having estimated a coarse scene segmentation, we proceed to jointly refine the segmentation and estimate depth in a view-dependent manner for each camera view. The main motivation for using this approach instead of a standard sequential pipeline where segmentation is followed by depth estimation is to avoid propagation of errors between the two stages (segmentation errors would result in reconstruction errors in a sequential pipeline) and also disambiguate the problem by simultaneously using all available cues (valid foreground pixels must for example be photo-consistent).

This defines a labelling problem where we seek the mappings $l : \mathcal{P} \rightarrow \mathcal{L}$ and $d : \mathcal{P} \rightarrow \mathcal{D}$, which respectively assign a layer label $l_{\boldsymbol{p}}$ and a depth label $d_{\boldsymbol{p}}$ to every pixel $\boldsymbol{p}$ in a given image. $\mathcal{P}$ denotes the set of pixels in the reference image; $\mathcal{L}$ and $\mathcal{D}$ are discrete sets of labels representing the different layer and depth hypotheses. $\mathcal{L}$ consists of a single background layer and multiple foreground layers defined by the different visual hull connected components computed

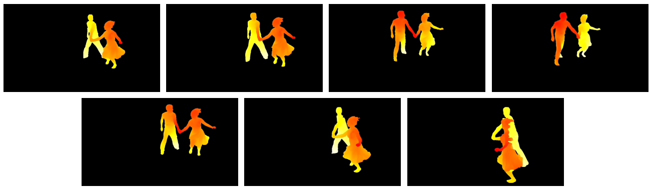


Fig. 3. Examples of depth maps estimated from different camera viewpoints at a given time instant.

from the coarse trimaps. The visual hull (see Fig. 4(a)) produces a coarse scene reconstruction that is used to initialise the layered depth estimation process. The set of depth labels $\mathcal{D}$ is formed of depth values d_i obtained by discretising the 3D space together with an unknown label $\mathcal{U}$ accounting for occlusions. Occlusions are common and can be severe with a wide-baseline set-up, especially in the background where large areas are often visible only in a single camera.

Computation of the optimum labelling (l, d) is formulated as an energy minimisation problem of the cost function

$$E(l, d) = w_1 E_1(l) + w_2 E_2(l) + w_3 E_3(d) + w_4 E_4(l, d). \quad (3)$$

The energy terms correspond to various cues derived from layer colour models, contrast, photo-consistency and smoothness priors, whose relative contribution is controlled by the parameters w_1, w_2, w_3 and w_4. Optimisation of the energy defined by Eq. (3) is an NP-hard problem, however an approximate solution with strong optimality properties can be computed using the $alpha$-expansion algorithm based on graph-cuts [8, 9]. Next we give a brief description of each energy term (see [10] for more details).

2.2.1. *Colour term*

The colour term encourages assignment of pixels to the layer following the most similar colour model, and is defined as

$$E_1(l) = \sum_{\boldsymbol{p} \in \mathcal{P}} - \log P(\boldsymbol{I_p} | l_{\boldsymbol{p}}), \quad (4)$$

where $P(\boldsymbol{I_p} | l_{\boldsymbol{p}} = l_i)$ is the probability at pixel $\boldsymbol{p}$ in the reference image of belonging to layer l_i. This probability is computed from learnt local and global colour models similar to those defined in Section 2.1 and combined in a similar fashion to [11]. A dual colour model combining global and local components allows for dynamic changes in the background.

2.2.2. *Contrast term*

The contrast term encourages layer discontinuities to occur at high contrast locations. This naturally encourages low contrast regions to coalesce into layers and favours discontinuities to follow strong edges. This term is defined as

$$E_2(l) = \sum_{(\boldsymbol{p},\boldsymbol{q}) \in \mathcal{N}} e_2(\boldsymbol{p}, \boldsymbol{q}, l_{\boldsymbol{p}}, l_{\boldsymbol{q}}), \text{ with} \quad (5)$$

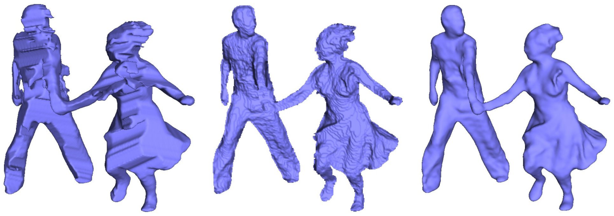


Fig. 4. Example of visual hull (a), view-dependent reconstruction obtained from a single depth-map (b) and final geometry after depth maps merging (c).

$$e_2(\boldsymbol{p}, \boldsymbol{q}, l_{\boldsymbol{p}}, l_{\boldsymbol{q}}) = \begin{cases} 0 & \text{if } l_{\boldsymbol{p}} = l_{\boldsymbol{q}}, \\ \exp(-\beta ||\boldsymbol{I_p} - \boldsymbol{I_q}||) & \text{otherwise.} \end{cases} \quad (6)$$

$\mathcal{N}$ denotes the set of interacting pairs of pixels in $\mathcal{P}$ (a 4-connected neighbourhood is assumed) and β is a weighting parameter.

2.2.3. *Matching term*

The matching term encourages depth assignments to maximise appearance similarity across views and is defined as

$$E_3(d) = \sum_{\boldsymbol{p} \in \mathcal{P}} \text{NCC}(\boldsymbol{p}, d_{\boldsymbol{p}}), \quad (7)$$

where NCC denotes the normalised correlation averaged over all image pairs for the 3D point hypothesis corresponding to pixel $\boldsymbol{p}$ and located at a depth $d_{\boldsymbol{p}}$ in the reference image. Occluded pixels are assigned a constant penalty $\mathcal{U}$.

2.2.4. *Smoothness term*

The smoothness term encourages the depth labels to vary smoothly within each layer. It is defined as

$$E_4(l, d) = \sum_{(\boldsymbol{p},\boldsymbol{q}) \in \mathcal{N}} D_{l_{\boldsymbol{p}}, l_{\boldsymbol{q}}}(d_{\boldsymbol{p}}, d_{\boldsymbol{q}}), \quad (8)$$

where $D_{l_{\boldsymbol{p}}, l_{\boldsymbol{q}}}(d_{\boldsymbol{p}}, d_{\boldsymbol{q}})$ is a truncated linear distance (see [10]). Such a distance is discontinuity preserving and prevents over-penalising large discontinuities within a layer; this is known to be superior to simpler non-discontinuity functions (see [8]).

2.3. Multi-view merging

After view-dependent reconstruction, the depth maps may be noisy and inconsistent, which would produce artefacts if directly used for rendering. To remove these artefacts and improve multi-view consistency, the depth maps are merged into a unique representation using Poisson surface reconstruction [12] (see Fig. 4(b) and Fig. 4(c) respectively for an example of mesh obtained from a single depth map and the improved geometry after merging). This results in an accurate reconstruction suitable for high-quality stereoscopic rendering.

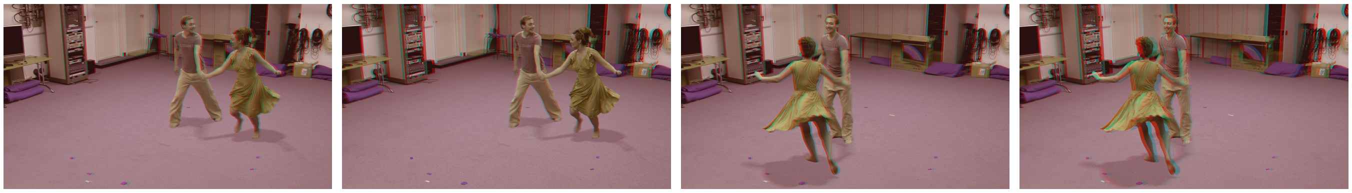


Fig. 5. Sample images from two dance sequences illustrating adjustment of the convergence distance (left two images) and inter-axial distance (right two images) in optimised red/cyan anaglyph format. Full sequences can be downloaded from our webpage.

3. RESULTS

The method was tested on two 250 frame sequences containing complex dance motions with multiple actors generating self-occlusions and motion blur. Data was captured using 8 Viper cameras located around the scene (7 at the front covering a 120° baseline and one at the back). The technique is applied to automatically segment and reconstruct the foreground using a single manually defined key-frame in a single view. The background, which does not require the same level of modelling accuracy as the foreground, is represented as a cube; in a real production environment, this could be replaced with a computer generated background model. We define left and right virtual camera views located on either side of each input monoscopic camera and with the same intrinsic parameters and orientation. Two key parameters defining the stereoscopic camera configuration are the inter-axial distance and the convergence distance. The inter-axial distance (separating left and right camera's optical centres) controls the amplitude of the depth effect. The convergence distance controls the location of the scene with respect to the 3D display in viewer space. Points with positive parallax appear located behind the display, while points with negative parallax appear in front. To avoid discomfort to the viewer, these parameters should be defined so as not to severely break the accommodation/convergence relationship which the eyes are used to [13]. Images for the left and right views are synthesised using a view-dependent texture mapping technique [14]. The synthesised stereoscopic sequences are very realistic and fully reconfigurable at render time. Sample images for different stereoscopic settings are shown in Fig. 5. Full sequences can be downloaded from http://www.guillemaut.org/publications/10/GuillemautICIP10.

4. CONCLUSIONS AND FUTURE WORK

We proposed a full pipeline for stereoscopic content production based on state-of-the-art Bayesian inference techniques and graph-cut optimisation to automatically convert multiple wide-baseline monoscopic camera feeds into stereoscopic outputs and thus provide full control of the stereoscopic parameters at render time. Experiments with scenes containing complex human motions have demonstrated the applicability of the technique. Future work will concentrate on improving temporal consistency and extending the technique to free-viewpoint video applications; this would further enhance a viewer's experience by providing control of the stereoscopic rendering viewpoint.

5. REFERENCES

[1] S. Knorr, M. Kunter, and T. Sikora, "Super-resolution stereo- and multi-view synthesis from monocular video sequences," in *3DIM*, 2007, pp. 55–64.

[2] P.J. Narayanan, P.W. Rander, and T. Kanade, "Constructing virtual worlds using dense stereo," in *ICCV*, 1998, pp. 3–10.

[3] S.M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski, "A comparison and evaluation of multi-view stereo reconstruction algorithms," in *CVPR*, 2006, pp. 519–528.

[4] C.L. Zitnick, S.B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," in *SIGGRAPH*, 2004, pp. 600–608.

[5] A. Smolic, "An overview of 3d video and free viewpoint video," in *CAIP*, 2009, pp. 1–8.

[6] Y.Y. Chuang, A. Agarwala, B. Curless, D. Salesin, and R. Szeliski, "Video matting of complex scenes," *SIGGRAPH*, pp. 243–248, 2002.

[7] M. Sarim, A. Hilton, and J.-Y. Guillemaut, "Wide-baseline matte propagation for indoor scenes," in *CVMP*, 2009, pp. 61–70.

[8] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *PAMI*, vol. 23, no. 11, pp. 1222–1239, 2001.

[9] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *PAMI*, vol. 26, no. 9, pp. 1124–1137, 2004.

[10] J.-Y. Guillemaut, J. Kilner, and A. Hilton, "Robust graph-cut scene segmentation and reconstruction for free-viewpoint video of complex dynamic scenes," in *ICCV*, 2009.

[11] J. Sun, W. Zhang, X. Tang, and H.-Y. Shum, "Background cut," in *ECCV*, 2006, vol. 3954, pp. 628–641.

[12] M. Kazhdan, M. Bolitho, and H. Hoppe, "Poisson surface reconstruction," in *Symp on Geometry Processing*, 2006, pp. 61–70.

[13] L. Lipton, *Foundations of the Stereoscopic Cinema*, Van Nostrand Reinhold Company, 1982.

[14] P.E. Debevec, C.J. Taylor, and J. Malik, "Modeling and rendering architecture from photographs: a hybrid geometry- and image-based approach," in *SIGGRAPH*, 1996, pp. 11–20.