CLUSTERING-BASED MATCH PROPAGATION FOR IMAGE-BASED RENDERING FROM MULTIPLE IMAGES

Jian Yao *Wai-Kuen Cham*

Department of Electronic Engineering, The Chinese University of Hong Kong, N.T., Shatin, Hong Kong

ABSTRACT

In this paper we propose a novel method for rendering a new view from multiple input images. A set of reliable pixel matches are initially selected and then propagated to neighboring pixels based on the piecewise smooth assumption in the spatial domain and the clustering-based photoconsistency constraint which is applied to handle pixel occlusion, partial pixel effects and lighting effects between images. Experimental results show that the proposed algorithm is promising and satisfactory.

1. INTRODUCTION

Recently, there has been much interest in image-based rendering (IBR) methods [1, 2] which generate new views of scenes from novel viewpoints using a collection of images with known viewpoints. Recently proposed methods and techniques for image-based modeling and rendering are described and discussed in [3, 4]. Image-based rendering has many potential applications, such as visual simulation or virtual reality [5], where computer graphics methods are not feasible because of very complex scenes.

One most natural approach to IBR is to explicitly compute a 3D representation of the scene for rendering by texture mapping from a collection of images utilizing computer-vision-based 3-D reconstruction techniques [6,7]. Arbitrary views of the scene can be synthesized by reprojecting the reconstructed 3-D model [8]. Typical examples of this approach are stereo reconstructions [9] and volumetric techniques such as space carving [10,11]. More recent work has shown that some implicit-geometry techniques [1], which assemble the pixels of the synthesized view from the rays sampled by the pixels of input images, are very efficient for rendering of complex scenes.

In [13], Lhuillier and Quan proposed an efficient match propagation algorithm which produces a quasi-dense pixel matching between two images using region growing principle. In this paper, we describe a novel image-based rendering method from multiple images based on the clustering-based match propagation principle. The basic idea of the proposed match propagation algorithm is the same as in [13]. It starts from a set of sparse matches as seed points, then propagate the results to neighboring pixels. In this paper, we propose a different propagation principle and a new measure criteria. The proposed algorithm implements the match propagation based on the piecewise smooth assumption in the spatial domain and clustering-based photoconsisntency constraint between multiple images.

2. PROBLEM FORMULATION

Given a set of 2D input images $\mathcal{I}_1$ to $\mathcal{I}_n$ taken by cameras in different positions, a new view $\mathcal{V}$ from a new viewpoint may be synthesized. Fig. 2 demonstrates the view synthesis situation. It is assumed that we are dealing with diffuse and opaque objects, and any deviation from this assumption may be considered as part of imaging noise. Both pixel match propagation and clustering-based photoconsistency constraint between input images are utilized to complete the task of the virtual view synthesis.

3. CLUSTERING-BASED MATCHING

3.1. Initialization

In the proposed rendering algorithm, image sequences of the scene to be rendered are first captured by a hand-held camera. Then we calibrate and track all camera views from the image sequence itself based on corresponding points. Euclidean locations of these corresponding points can be further recovered. Here we use a commercial camera tracking software [14] to complete these tasks. For each novel view to render, it has to be decided which input images to use. We use three camera ranking criteria developed in [15] for ordering input images. A list of best suited input images denoted by $\mathcal{I}_1$ to $\mathcal{I}_n$ is selected for view synthesis.

3.2. Clustering-Based Photoconsistency Constraint

For each pixel $\mathbf{x}$ on the view $\mathcal{V}$ to be rendered, we assumed that its depth search range $[z_{\min}, z_{\max}]$ is known. If no image noise such as pixel occlusion, partial pixel effects, or lighting effects exist, we can choose the optimal color $V(\mathbf{x})$ for the pixel $\mathbf{x}$ by searching the depth range $[z_{\min}, z_{\max}]$ and the whole RGB colorspace to minimize the deviation from photoconsistency at pixel $\mathbf{x}$ as follows:

$$E_{\text{photo}}(V(\mathbf{x}), z) = \sum_{i=1}^{n} \|V(\mathbf{x}) - I_i(\mathbf{X}(z))\|^2, \qquad (1)$$

where $\mathbf{X}(z)$ denotes a 3D point with the depth z along the back-projection unit ray at pixel $\mathbf{x}$ from the camera $\mathbf{C}_v$ and

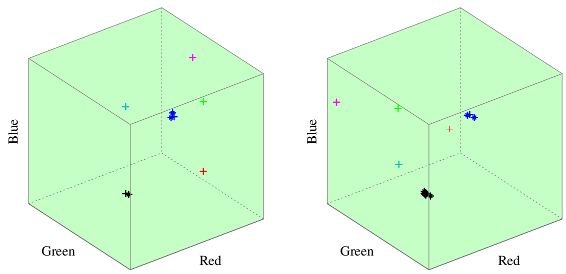

Fig. 1. Clustering results of the re-projected pixel sets $C(z)$ for two typical depth samples in the depth search range $[z_{\min}, z_{\max}]$. Only six largest clusters are shown here. The symbol '$+$' denotes the centroids of the clusters, the symbol '$*$' denotes the elements of the clusters, and the same color denotes the same cluster.

$I_i(\mathbf{X}(z))$ denotes the pixel in the image $\mathcal{I}_i$ on which the 3D point $\mathbf{X}(z)$ projects. $\mathbf{C}_v$ is the origin of the coordinate system used here.

The above minimization problem can be solved in two steps. In the first step, we find the optimal colour $V_z(\mathbf{x})$ for a particular depth z by averaging all the re-projected pixels $\{I_i(\mathbf{X}(z))\}$, i.e. $V_z(\mathbf{x}) = \sum_{i=1}^{n} I_i(\mathbf{X}(z))/n$. In the second step, we find the optimal depth value of these noise-free pixels by minimizing the following function:

$$E_{\text{photo}}(z) = \min_z \sum_{i=1}^{n} \|V_z(\mathbf{x}) - I_i(\mathbf{X}(z))\|^2. \quad (2)$$

Subsequently we can get the optimal 3D location for the pixel $\mathbf{x}$ to be rendered.

However, image noise often exists in part of image sequences because of the parallax between images, partial pixel effects or specular reflections. Hence, we must first try to find a set of reliable pixels on some images. For a given depth value z along the ray from $\mathbf{C}_v$, let the set of re-projected pixels on input image sequences $\mathcal{I}_i, i = 1, \cdots, n$ be $C(z) = \{I_i(\mathbf{X}(z)), i = 1, \ldots, n\}$. Here, we utilize a data clustering algorithm to keep only the reliable pixels by finding the largest cluster in the RGB colorspace.

We arbitrarily choose one pixel to be a centroid and cluster all the pixels around this centroid. We then find the pixel farthest away from the centroid and makes this pixel a new centroid. Next we cluster the pixels around the new centroid. This process is repeated until the distance from every pixel to its centroid is less than some predefined threshold Θ_{dist}. In our system, we set $\Theta_{\text{dist}} = 20$. Two typical clustering results are shown in Fig. 1. For noise-free pixels, we find the optimal value of the depth z using (2). For these noisy pixels, we find the optimal value of z in a different way in which two factors are considered. They are the deviation from photo-consistency and the number of elements of the largest cluster $N(z)$. These two factors are weighted and combined into one scalar value $\texttt{PMQ}_z(\mathbf{x})$ which represents the pixel matching quality:

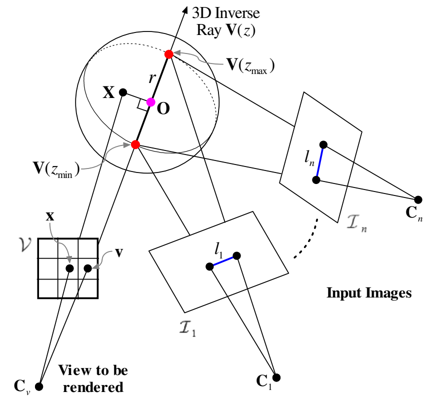

Fig. 2. Measure of new matches of neighboring pixels around seed points. Based on the piecewise smooth assumption in the spatial domain, both valid depth search range and depth sampling number are determined.

$$\texttt{PMQ}_z(\mathbf{x}) = \alpha \left(1 - \frac{\sum_{i \in S_z} \|V_z(\mathbf{x}) - I_i(\mathbf{X}(z))\|^2}{N(z) \times \Theta_{\text{dist}}^2}\right) + (1-\alpha)\frac{N(z)}{n}, \quad (3)$$

where n is the total number of used images and $\alpha \in [0, 1]$ is a weighting factor. We choose the value of z as the one that provides the best pixel matching quality over the depth search range $[z_{\min}, z_{\max}]$:

$$\texttt{BPMQ}(\mathbf{x}) = \max_{z \in [z_{\min}, z_{\max}], N(z) \geq 2} \texttt{PMQ}_z(\mathbf{x}). \quad (4)$$

Also, we require that the valid depth values are such that the total number of elements of the largest cluster $N(z) \geq 2$. From the above definition, $\texttt{BPMQ}(\mathbf{x}) \in [0, 1]$. With the optimal value of the depth z found using (4), we then compute the optimal color $V_z(x)$ by averaging all the elements of the cluster.

3.3. Match Propagation

For the corresponding points obtained in the initialization process, we can find the optimal colour based on the proposed clustering-based photoconsistency constraint as explained in Section 3.2. For a particular new view to be rendered, maybe some corresponding points are occluded. We may remove occluded or invalid corresponding points by setting $\texttt{BPMQ} < 0.8$. The remaining corresponding points are regarded as initial matches. All these initial sparse matches are put in a set called set of seed points. Elements of this set are sorted by decreasing $\texttt{BPMQ}$ scores. The seed point $\mathbf{x}$ with the best $\texttt{BPMQ}$ score is removed from the current set of seed points and the corresponding optimal color $V_z(\mathbf{x})$ is used as the color of $\mathbf{x}$ in the rendered image. As shown in Fig. 2, the seed point $\mathbf{x}$ has 8 neighboring pixels. We

(a) (b) (c) (d) (e)

Fig. 3. Recovering a missing view using 6 views: (a) Ground-truth view in which red cross '+' symbols denote 172 initial matches obtained from the reconstructed 3D points; (b)-(c) Two intermediate rendered images; (d) Final rendered image; (e) Difference image between (a) and (d).

(a) (b) (c) (d)

Fig. 4. (a) Corresponding depth map of recovered view in Fig. 3(d) (near=dark,far=light); (b)-(d) Three re-projected images on input images.

propose the following steps to find the possible values of z represented by $[z_{\min}, z_{\max}]$ for each of these 8 neighboring pixels, which are denoted by $\mathbf{v}$.

1. Get the 3D inverse ray $\mathbf{V}(z)$;
2. Find the perpendicular point $\mathbf{O}$ located on the 3D inverse ray $\mathbf{V}(z)$ projected from the 3D point $\mathbf{X}$ corresponding to the pixel $\mathbf{x}$;
3. Draw a sphere with the center $\mathbf{O}$ and the radius $r = 10 \times \overline{\mathbf{XO}}$ where $\overline{\mathbf{XO}}$ denotes the distance between 3D points $\mathbf{X}$ and $\mathbf{O}$;
4. Determine the two intersection points denoted by $\mathbf{V}(z_{\min})$ and $\mathbf{V}(z_{\max})$ of the 3D inverse ray $\mathbf{V}(z)$ and the sphere surface.

Another problem is the determination of the number of depth sampling. We can project the line segment from $\mathbf{V}(z_{\min})$ to $\mathbf{V}(z_{\max})$ on input images $\mathcal{I}_1$ to $\mathcal{I}_n$ and we get n line segments l_1 to l_n on these input images. Let $\mathbf{L}_{\max}$ be the largest length of these line segments in pixel. In our system, the number of depth sampling is set as $2 \times \mathbf{L}_{\max}$.

After that, we can find the optimal depth value and the optimal colour for the neighboring pixel $\mathbf{v}$ based on the proposed clustering-based photoconsistency constraint by explicitly sampling the obtained depth search range $[z_{\min}, z_{\max}]$. Similarly the best pixel matching quality BPMQ($\mathbf{v}$) is also calculated. If BPMQ($\mathbf{v}$) is not less than the threshold Θ_{BPMQ}, it will be put in the set of seed points for the next round of propagation. After all neighboring pixels around the removed seed point are handled, we sort the set of seed points again by decreasing the BPMQ scores. This process is repeated until the whole set of seed points is empty. Even so, there will be many pixels not rendered. To finish the whole rendering for the view $\mathcal{V}$, we collect all the rendered pixels whose adjacent pixels are not all rendered as the new set of seed points. We propagate these to neighboring pixels again by reducing the value of the threshold Θ_{BPMQ}. The whole process is repeated until all pixels are rendered or $\Theta_{\text{BPMQ}} < 0$.

3.4. Propagation Algorithm

The propagation algorithm can be summarized as follows. The input of the algorithm is `Seed` which is the set of the current seed points. The outputs include the rendered `Colour` and `Depth` which are the colour and the optimal 3D locations of pixels in the new view respectively.

Input: `Seed`
Output: `Colour` and `Depth`
initialize the threshold $\Theta_{\text{BPMQ}} = 0.8$
while not all the pixels are rendered **and** $\Theta_{\text{BPMQ}} > 0$ **do**
  while `Seed` $\neq \emptyset$ **do**
    pull the BPMQ-best pixel $\mathbf{x}$ from `Seed` for propagation
    for each pixel $\mathbf{v}$ in 3×3 neighborhood $\mathcal{N}(\mathbf{x})$ **do**
      if BPMQ($\mathbf{v}$) $\geq \Theta_{\text{BPMQ}}$ **then**
        store the rendered colour and the optimal depth value in `Colour` and `Depth`
        store BPMQ($\mathbf{v}$) and the optimal 3D point $\mathbf{V}$ in `Seed`
      end if
    end for
  end while
  reset `Seed` by collecting all the rendered pixels whose adjacent pixels are not all rendered
  sort `Seed` by decreasing BPMQ scores
  reset $\Theta_{\text{BPMQ}} = \Theta_{\text{BPMQ}} - 0.1$
end while

4. EXPERIMENTS

In our experiments, we used the same image sequences as in [12]. The complete MPEG sequences, camera projection matrices and reconstructed 3D points may be found at `http://www.robots.ox.ac.uk/~awf/ibr`. A missing view will be recovered using six existing views. Two typical intermediate rendered results, the final rendered image, and the difference image between the recovered image and the original one are shown in Fig. 3. The obtained optimal depth map of the recovered image and several reprojected images are shown in Fig. 4. The whole computation time for this example is 34.2 seconds with a P4 1.8Ghz. The image size is 600×470. We also tested the rendering of novel views of the scene and the results are shown in Fig. 5. From these experimental results, we may conclude that both the recovered depth map and the rendered image by the proposed algorithm are satisfactory.

Fig. 5. Four novel views of the monkey scene from viewpoints not in the original sequence.

5. CONCLUSIONS

A novel method for rendering a new view from multiple images by match propagation is proposed and discussed in this paper. First a set of reliable pixel matches are initially selected and then propagated to neighboring pixels based on the piecewise smooth assumption in the spatial domain and the clustering-based photoconsistency constraint which handles occlusion, partial pixel effects or lighting effects. In the proposed algorithm, we obtain not only the rendered colour but also the approximative 3D location for each pixel on the view to be rendered. Experimental results show that the proposed algorithm is very promising and satisfactory.

Acknowledgements

The work described in this paper was substantially supported by a grant from the Research Grants Council of the Hong Kong Special Administrative Region (Project no. CUHK4167/01E).

6. REFERENCES

[1] M. Lhuillier and L. Quan, "Image-Based Rendering by Joint View Triangulation," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 13, No. 11, pp.1051-1063, Nov. 2003.

[2] C. Zhang and T. Chen, "Spectral Analysis for Sampling Image-Based Rendering Data," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 13, No. 11, pp.1038-1050, Nov. 2003.

[3] M.M. Oliveira, "Image-Based Modeling and Rendering Techniques: A Survey," *RITA - Revista de Informática Teóricae Aplicada*, Vol. IX, No. 2, pp.37-66, Oct. 2002.

[4] D. Burschka, G.D. Hager, Z. Dodds, M. Jägersand, D. Cobzas, and K. Yerex, "Recent Methods for Image-Based Modeling and Rendering," *IEEE Virtual Reality*, Los Angeles, USA, March 2003.

[5] R.J. Radke, P.J. Ramadge, S.R. Kulkarni, and T. Echigo, "Efficiently Synthesizing Virtual Video," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 13, No. 4, pp.325-337, Apr. 2003.

[6] M. Magnor, P. Ramanathan, and B. Girod, "Multi-View Coding for Image-Based Rendering Using 3-D Scene Geometry," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 13, No. 11, pp.1092-1106, Nov. 2003.

[7] A.Y. Mulayim, U. Yilmaz, and V. Atalay, "Silhouette-Based 3-D Model Reconstruction From Multiple Images," *IEEE Trans. Syst., Man, Cybern. B*, Vol.33, No.4, pp.582-591, Aug. 2003.

[8] S. Yaguchi and H. Saito , "Arbitrary Viewpoint Video Synthesis From Multiple Uncalibrated Cameras," *IEEE Trans. Syst., Man, Cybern. B*, Vol.34, pp.430-439, Feb. 2004.

[9] P.J. Narayanan, P.W. Rander, and T. Kanade, "Constructing virtual worlds using dense stereo," In *Proc. ICCV*, pp.3-10, Freiburg, Germany, Jan. 1998.

[10] A. Broadhurst and R. Cipolla, "A Statistical Consistency Check for the Space Carving Algorithm," In *Proc. 11th BMVC*, pp.282–291, Bristol, Sept. 2000.

[11] S.M. Seitz and C.R. Dyer, "Photorealistic Scene Reconstruction by Voxel Coloring," In *Proc. CVPR*, pp.1067-1073, Puerto Rico, June 1997.

[12] A. Fitzgibbon, Y. Wexler, and A. Zisserman, "Image-based rendering using image-based priors," In *Proc. ICCV*, Vol.2, pp.1176-1183, Nice, France, Oct. 2003.

[13] M. Lhuillier and L. Quan, "Match Propagation for Image-Based Modeling and Rendering," *IEEE Trans. Pattern Anal. Machine Intell.*, Vol.24, No.8, pp.1140-1146, Aug. 2002.

[14] 2d3 Ltd. http://www.2d3.com, 2002.

[15] J.-F. Evers-Senne and R. Koch, "Image Based Interactive Rendering with View Dependent Geometry," In *Proc. Eurographics 2003*, Vol.22, No.3, Granada, Spain, pp.573-582, Sept. 2003.