QUESTION INTERFACE FOR 3D PICTURE CREATION ON AN AUTOSTEREOSCOPIC DIGITAL PICTURE FRAME

Chris Varekamp[a], *Patrick Vandewalle*[a], *Marc de Putter*[b]

(a) Philips Research, Eindhoven, The Netherlands
(b) Philips 3D Solutions, Eindhoven, The Netherlands

ABSTRACT

We propose an interface for creating a depth map for a 2D picture. The image and depth map can be used for 3D display on an autostereoscopic photo frame. Our new interface does not require the user to draw on the picture or point at an object in the picture. Instead, semantic questions are asked about a given indicated position in the picture. This semantic information is then translated automatically into a depth map.

Index Terms— still picture conversion, autostereoscopic display, digital photo frame.

1. INTRODUCTION

A 3D digital photo frame with autostereoscopic display is currently investigated as possible product by Philips 3D Solutions. Ideally, a user can load pictures from a normal digital camera onto the photo frame and view them in 3D. However, here we face the problem that existing fully automated 2D to 3D conversion methods produce insufficient 3D quality and existing semi-automated conversion methods are too complex for efficient operation by the typical photo frame user.

We believe that it is too complex for many users to draw a good depth map. Furthermore, an approach where a depth map is drawn also requires an interface that uses a pen to outline object contours or to point to objects. It is therefore not easy to make a simple user interface that requires little or no explanation before use.

We propose an interface and algorithm that lets the user convert a picture without requiring 3D knowledge. Instead of having to draw a depth map, the user needs to answer a few questions about specific locations in the picture (Figure 1). A depth map is automatically calculated and the picture may be viewed in 3D.

A location in the picture is indicated with a circle. Initially this circle is placed at the center of the picture. The user can select one of a small set of predefined object classes such as: ‘Sky’, ‘Ground’, ‘Building’, ‘Person’, ‘Animal’. This information is then used to produce a dense depth map. This means that by answering a couple of questions a user can convert 2D pictures into 3D.

This paper is organized as follows. In Section 2 we review relevant previous work. In Section 3 we describe the autostereoscopic display for which the conversion is intended. Section 4 provides details on the conversion algorithm including the proposed method of repositioning the position indicator for a new question. In Section 5, we show results of the proposed method and discuss these. Conclusions are drawn in Section 6.

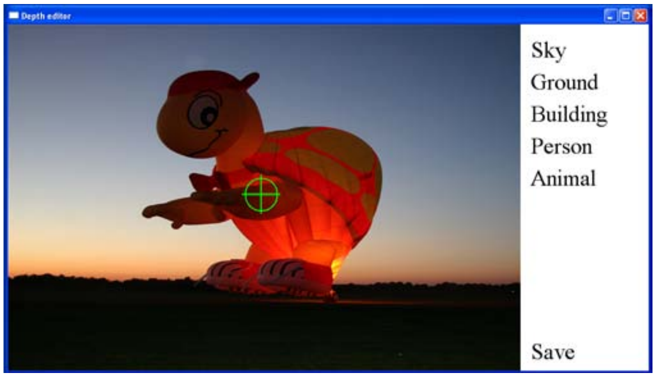

Figure 1: Question interface for 3D picture creation. Pressing one of the buttons sets the class at the location being indicated by the circle. The green circle is then automatically placed in the position where a label input is most needed.

2. PREVIOUS WORK

Ideally, depth estimation should be fully automated as proposed by Saxena et al. [1] and Hoiem et al. [2]. Although this approach is very promising, segmentation errors and depth errors will remain for pictures that are not characterized well in the training set on which the model is learned. On the other hand, semi-automated methods for 3D creation [3][4][5] have not addressed the constraint that the user-interface must be very simple. DDD (Harman et al. [3]) has patented a method to assign a certain depth to a set of pixels and automatically extend this to other pixels. While conceptually simple, the user must understand which depth to assign to a certain object. This may cause problems, especially for planes that have varying depth (such as the ground surface). The method proposed by van den Hengel et al. allows for a high-quality conversion but requires object boundary specification and 3D input together with some 3D modeling knowledge [4]. The system described by Russell et al. uses object detection from learned object categories [5]. A depth ordering is then possible once the outlines of multiple objects are known. Using this approach for picture conversion seems promising but depends much on the ability to handle the large variation of objects (e.g. a building can have many shapes and sizes). Finally, roughly indicating an object’s interior with a pen offers an efficient way of segmenting objects. Bai and Sapiro describe this new form of interaction to efficiently produce a foreground/background seg-mentation and transparency channel [6]. Our method differs from these approaches since we do not require the user to draw or indicate anything in the picture area.

978-1-4244-4318-5/09/$25.00 ©2009 IEEE

3. AUTOSTEREOSCOPIC DISPLAY

Philips 3D Solutions has developed a 3D photo frame (Figure 2). This photo frame consists of standard 2D photo frame hardware and software, an autostereoscopic display and a 3D signal processing chip called IC3D. The autostereoscopic display has a 5-view lenticular lens with optimal viewing distance in the range 0.7-1.0 m. It is glued on a standard 8 inch LCD, aspect ratio 4:3 and 800×600 pixel resolution. The input data format for the 3D photo frame is a horizontally stacked image of the visual image with corresponding depth map. This stacked image is JPEG-compressed and loaded into the photo frame using a USB interface or memory card.

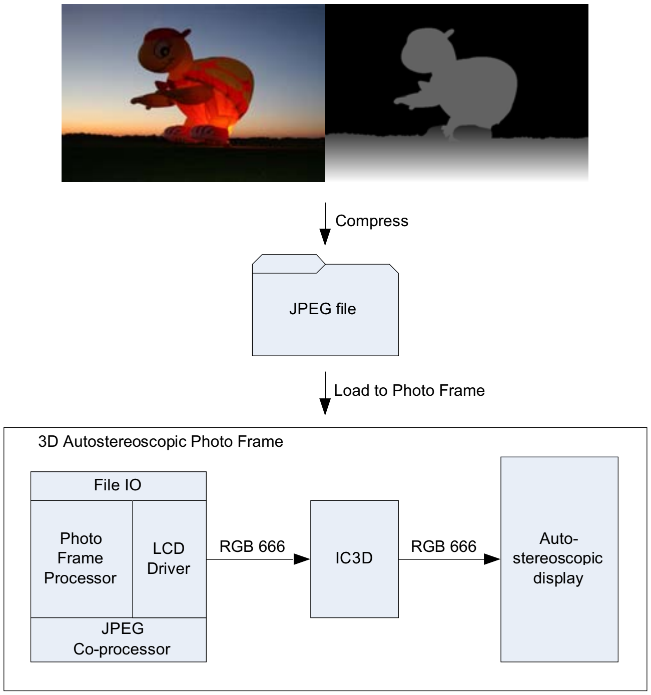

Figure 2: Block diagram of the 3D photo frame.

4. CONVERSION ALGORITHM

4.1. Color image segmentation

The purpose of color image segmentation is to make the question interface fast enough for practical use. We use an iterative segmentation algorithm that modifies the geometry of a square grid such that the regions become homogeneous with respect to color but have a smooth boundary. This approach was taken by Oliver and Quegan for synthetic aperture radar images [7]. We have implemented the global energy minimization efficiently using the derivations given in [8, pp. 548-549]. The resulting segmentation is shown in Figure 3. It can be seen that some initial regions keep their original square shape while other regions adapt their shape to become homogeneous with respect to color. As can be seen, important depth discontinuities coincide with region edges.

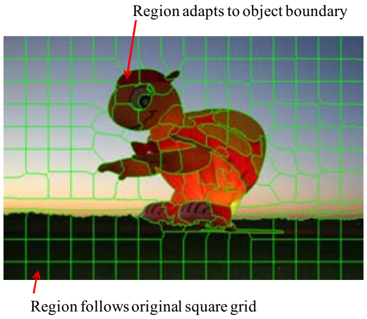
Region adapts to object boundary
Region follows original square grid

Figure 3: Result of color image segmentation.

4.2. Calculating a dense class label map

Figure 4 shows our pre-defined class labels that the user can select when answering the questions. Note that these labels were chosen intuitively based on our current data set. Other/additional labels can be easily incorporated.

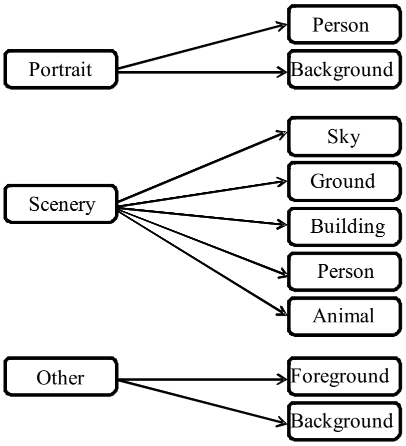

Figure 4: Three picture types with corresponding classes.

After each user input, the confidence in class $k \in K$ is re-initialized to zero for all regions for which the class is not known. The confidence is then calculated from the labeled regions, for which the class label is known. This is done independently for each class using an iterative algorithm based on confidences of neighbor regions and color differences between neighbor regions. For region i, with neighbor regions j, the confidence at iteration t is calculated as:

$$p_{i,t}^{(k)} \leftarrow \frac{\sum_j w_{ij} p_{j,t-1}^{(k)}}{\sum_j w_{ij}} \qquad (1)$$

where

$$w_{ij} = \exp\left(-\alpha\left(\left|r_i - r_j\right| + \left|g_i - g_j\right| + \left|b_i - b_j\right|\right)\right), \qquad (2)$$

and (r, g, b) is the pixel color where each channel takes a value from the set {0, …, 255} and $\alpha = 0.1$. Note that the use of certainty is known in literature[1]. Note also that w_{ij} can be pre-computed since it does not depend on the iteration number. We found that 50 iterations give good results for our test set. After each user input, the label map is automatically updated by selecting for each segment i the class k_i that has maximum confidence c_i after 50 iterations:

$$\begin{aligned} c_i &\equiv \max_{k\in K}\left(p_{i,50}^{(k)}\right) \\ k_i &\leftarrow \arg\max_{k\in K}\left(p_{i,50}^{(k)}\right) . \end{aligned} \quad (3)$$

For each new question we position the displayed indication in the region for which the maximum classification confidence c_i is lowest over all regions:

$$i^{(\text{new})} \leftarrow \arg\min_{i\in S}(c_i), \quad (4)$$

where $i^{(\text{new})}$ is the selected region and S is the set of all regions.

4.3. From class label map to depth map

What remains is converting the class label map into a depth map. For this step we use likely relations that exist between class labels and depth. For instance, a pixel that has label ‘Ground’ is likely to be part of a horizontal plane, while a pixel labeled as ‘Sky’ should go to the background.

For images labeled as ‘Scenery’, we first determine the largest occurring (top) vertical position, $y_{\text{g,max}}$, that has label ‘Ground’ (with the origin of the coordinate system at the lower-left of the picture). For all connected sets of regions k that have class label ‘Person’, ‘Animal’ or ‘Building’ we determine the smallest occurring (bottom) y-coordinate $y_{\min}$. The depth map is then computed by scanning the labeled image L from bottom to top and assigning depth values d_{xy} to a pixel with coordinates (x,y) as follows:

$$d_{xy} = \begin{cases} 0 & \text{if} \quad L_{xy} = \text{'Sky'} \\ 255 - sy & \text{if} \quad L_{xy} = \text{'Ground'} \\ 255 - sy_{\min} & \text{if} \quad L_{xy} = \{\text{'Pers'}, \text{'Anim'}, \text{'Build'}\} \end{cases} \quad (5)$$

where $s = 200/y_{\text{g,max}}$. This creates a slanted surface for the ground area, and places persons, animals and other objects vertically on this surface. The sky is put at the furthest point. Note that larger depth values are more to the front, with depth values ranging from 0 to 255.

For images labeled as ‘Portrait’ or ‘Other’, we only make a distinction between ‘Foreground’ (or ‘Person’) and ‘Background’. In this case, depth values are calculated as follows:

$$d_{xy} = \begin{cases} 0 & \text{if} \quad L_{xy} = \text{'Background'} \\ 128 & \text{if} \quad L_{xy} = \{\text{'Person'}, \text{'Foreground'}\} \end{cases} . \quad (6)$$

In this way, a depth map is created that can be displayed on the 3D picture frame (in combination with the original image).

5. RESULTS AND DISCUSSION

Figure 5 shows an input image of type ‘Scenery’ and the resulting confidence map and depth map after answering five questions about the class labels. The resulting depth map is satisfactory in terms of object alignment and depth realism. However, the picture is ideal since it consists of large regions of homogeneous color for which image segmentation and confidence propagation (equation 1) work well. It is interesting to note that a good depth map is produced even before confidences are high in all regions.

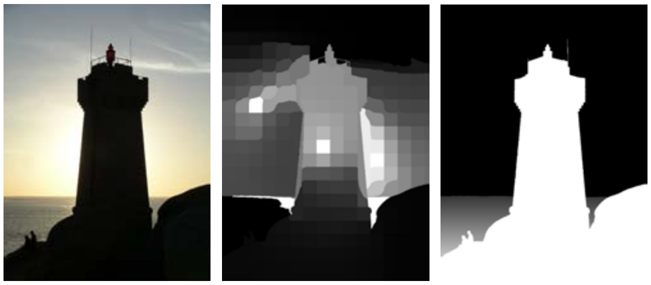

Figure 5: Input image (left); confidence map c (center); depth map (right). Five questions were answered to obtain this result.

Figure 6 shows resulting depth maps after 5, 10 and 15 questions for a larger set of pictures, from top to bottom: ‘balloon’, ’house’, ‘mill’, ‘ducks’, ‘tower’, ‘church’, ‘plumeria’, and ‘patrick’.[2] The images ‘balloon’ and ‘house’ are easy for our system and require respectively 5 and 10 questions to achieve an acceptable quality. The ‘mill’ image fails due to detailed depth structure. For instance the mill wings are gone. Another difficult case is the ‘ducks’ image. For this picture, lack of color contrast with the water and the large color variation of the ducks cause the classification to be unreliable. Total failure occurs for ‘church’. This may be attributed to the high amount of color variation inside the church making the propagation problematic.

Figure 7 shows the picture average confidence as a function of the question number. The most successful conversion results (‘balloon’ and ‘tower’) end up highest and the poorest conversion result (‘church’) ends up lower. However, in general, the average confidence is not a good predictor for depth map quality since all curves rise in roughly the same way.

To summarize, four out of the eight pictures in Figure 6 are converted successfully within 10 questions. The remaining 4 pictures suffer from grouping cues: the weighting based on color difference only (equation 2) is too limited.

[1] See for instance the work by Kohli and Torr for measurement in the context of ‘Graph Cut’ solutions for a Markov Random Field model [9].

[2] Images can be provided by the authors upon request.

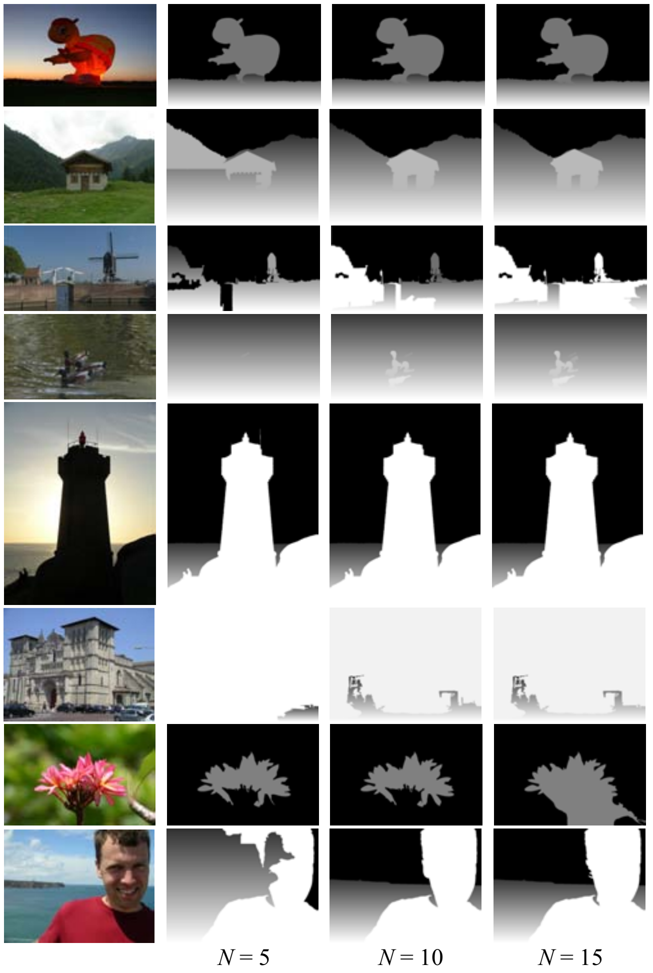

Figure 6. Depth maps after 5, 10 and 15 questions.

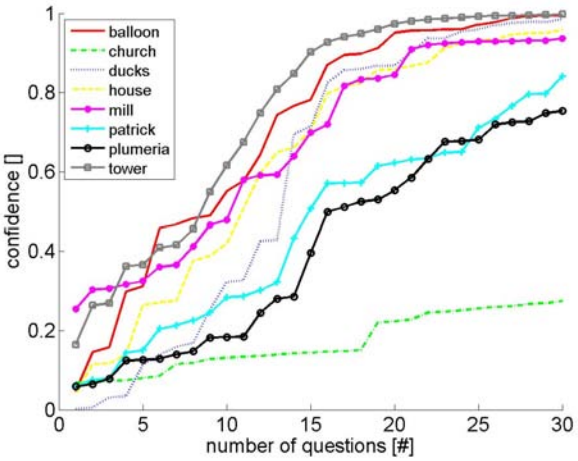

Figure 7. Average confidence as a function of question number.

6. CONCLUSIONS

We have introduced a user-friendly and efficient method for 2D to 3D picture conversion that can be used with an autostereocopic (3D) photo frame. The interface is efficient and easy to use. However, to improve the success rate of the conversion, more image features (such as texture) should be included in the estimation of class label confidences. This will allow better grouping of regions belonging to the same object, resulting in better depth maps.

7. ACKNOWLEDGEMENTS

We thank Hans Nellen (Philips 3D Solutions) for providing us with the 'balloon' picture. The 'mill' and 'ducks' picture were provided by Gerard Bloemen (Philips Research). We thank Frits de Bruijn and Karl van Bree (both Philips Research) for reviewing the manuscript.

8. REFERENCES

[1] A. Saxena, M. Sun, A.Y. Ng. Make3D: Learning 3D Scene Structure from a Single Still Image, To appear in *IEEE Transactions of Pattern Analysis and Machine Intelligence (PAMI)*, 2009.

[2] D. Hoiem, A.N. Stein, A.A. Efros, M. Hebert. Recovering Occlusion Boundaries from a Single Image. *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2007.

[3] P. V. Harman, S. R. Fox, M. R. Dowley, and J. C. Flack, Image Conversion and Encoding Techniques, *US Patent No. US7.035.451B2*, 2006.

[4] A. van den Hengel, A. Dick, T. Thormählen, B. Ward, P. Torr. VideoTrace: Rapid interactive scene modeling from video. *ACM Transactions on Graphics*, Vol. 26, No. 3, Article 86, 2007.

[5] B.C. Russell, A. Torralba, K.P. Murphy and W.T. Freeman. LabelMe: A Database and Web-Based Tool for Image Annotation. *International Journal of Computer Vision*, Vol. 77, issue 1-3, pp. 157-173, May 2008.

[6] X. Bai, G. Sapiro. A Geodesic Framework for Fast Interactive Image and Video Segmentation and Matting. *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2007.

[7] C. Oliver, S. Quegan. Understanding Synthetic Aperture Radar Images, Artech-House, 1998.

[8] Richard O. Duda, Peter E. Hart, and David G. Stork. Pattern Classification. John Wiley and Sons, Inc., New York, 2001.

[9] P. Kohli, P.H.S. Torr. Measuring Uncertainty in Graph Cut Solutions. *Journal of Computer Vision and Image Understanding, special issue Discrete Optimization in Computer Vision*, Vol. 112, No. 1, pp. 30-38, 2008.