Layered Photo Pop-Up

Lech Świrski*
University of Cambridge

Christian Richardt†
University of Cambridge

Neil A. Dodgson‡
University of Cambridge

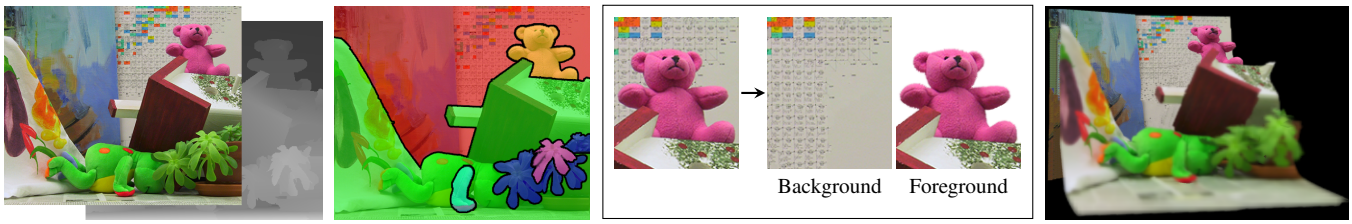

Figure 1: *Starting from an image with depth, we detect and separate objects. Each object is segmented using a depth-aware soft segmentation, and the region behind is filled using depth-aware inpainting. We use our layered representation to create novel views with depth-of-field, which is suitable for a documentary-style pop-up effect, similar to the popular "3D Ken Burns" effect.*

1 Introduction

A common technique in documentaries is to animate photographs by panning across them slowly. More recently, it has become popular to divide such photographs into layers, and to animate these layers as moving over each other to create a motion parallax effect, commonly known as the "3D Ken Burns effect". Producing this effect involves a laborious manual process that requires hours of manual rotoscoping, clone-brushing, positioning in 3D, and adjusting the panning speeds of individual layers.

Our work investigates how to automate this given depth information. We describe a novel workflow which, given an image with depth, mimics the manual creation of this motion parallax effect, by creating a layered image representation. Objects in the image are segmented into separate layers, and the regions behind them are filled by inpainting from the surrounding background. This imitation of the manual process gives a user the means to adjust the layers at various stages; with minimal interaction, objects can be manually marked for segmentation, or automatically filled regions can be augmented using human knowledge of the scene.

For our approach, we need a depth map. There are several ways of obtaining one. The most widespread is to calculate depth from stereoscopic imagery using stereo correspondence. Stereoscopic photographs are available historically, and are cheap to produce today. Another source of depth maps is active range-sensing cameras, for example using time-of-flight or structured light. The most popular recent example is the Microsoft Kinect. These greatly simplify the capture of high-quality depth maps alongside colour images.

2 Our Approach

We create layered images by iteratively selecting the foremost object in the image, segmenting it onto a new layer, and filling the hole behind it. We assume that layers are separated by depth discontinuities, so that they can move independently over each other under motion parallax.

To separate elements, we detect these depth discontinuities by thresholding depth differences between neighbouring pixels, and mark them as object boundaries. The foremost element is found by comparing the depths of pixels in neighbouring regions.

Once the foremost element has been selected, we use a modified version of GrabCut [Rother et al. 2004] to segment it from the remaining background. This has two steps: a hard segmentation followed by a soft, alpha matting refinement. Pixels are classified as foreground or background by fitting 4D GMMs to the data in *RGBZ* space, and we use graph cuts to perform a clean segmentation.

Following the hard segmentation, we apply alpha matting along the contour to create a soft segmentation. We calculate the alpha matte along the hard contours, using only the colour image as in GrabCut, and we compute the colour of each foreground pixel using Bayesian matting [Chuang et al. 2001]. Once the alpha matte is computed, we recalculate the depth of all semi-transparent pixels along the object border using plane-fitting.

Segmenting out a layer leaves behind a 'hole' of unknown data, which we mark as "invalid". We fill this hole with exemplar-based inpainting [Criminisi et al. 2004], extended to use and fill depth information. This technique fills holes by copying patches from other parts of the image; a patch is filled by finding the most similar patch from the remaining valid image pixels. Criminisi et al. use colour distance to determine patch similarity; we augment this to favour patches which are similar in 3D shape, and which are behind the object being erased.

Our representation is designed to allow arbitrary rendering of the given scene. We render the model by transforming each layer's mesh based on data from its depth map. Each vertex is positioned in 3D as a function of its (x, y) position in the image and the corresponding value d in the depth map. This function is implemented as part of the vertex shader in the graphics pipeline.

References

CHUANG, Y.-Y., CURLESS, B., SALESIN, D. H., AND SZELISKI, R. 2001. A Bayesian approach to digital matting. In *CVPR*, 264–271.

CRIMINISI, A., PÉREZ, P., AND TOYAMA, K. 2004. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on Image Processing 13*, 9 (Sep.), 1–13.

ROTHER, C., KOLMOGOROV, V., AND BLAKE, A. 2004. GrabCut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Computer Graphics 23*, 3, 309–314.

*e-mail:lech.swirski@cl.cam.ac.uk
†e-mail:christian.richardt@cl.cam.ac.uk
‡e-mail:neil.dodgson@cl.cam.ac.uk