Fakeye: Sky Augmentation with Real-time Sky Segmentation and Texture Blending

Anh-Thu Tran, Yen Le
University of Science, VNU-HCM
227 Nguyen Van Cu Street, Ward 4, District 5, Ho Chi Minh City, Vietnam
{ttathu, lpnyen}@apcs.vn

Abstract

Augmented Reality (AR) has been intensively used to enhance human experience by providing artificial content in addition to their real surroundings. While many AR applications are focusing on serving indoor tasks, higher and vaster space such as the sky should also be paid attention to in order to create a fuller virtual environment. Since objects from afar behave differently from near objects in term of rendering, in this paper, we try to devise an approach to augment the sky with virtual objects where challenges such as real-time sky segmentation for creating illusion of occlusion, real-and-virtual scenes blending as well as cameras alignment are also addressed. Our mobile implementation of the approach which is called "Fakeye" produces promising result and brings about exciting experience.

Figure 1: The application Fakeye augments the sky with a virtual 3D cube.

1. Introduction

Augmented reality (AR) has been rapidly growing over the past few years with many different forms such as marker based [1], markerless [2] and location based [6]. Most of today's applications often aim to render virtual objects in a way that they appear to closely surround people so as to accompany domestic tasks of learning, training and playing with more visual information. However, when it's come to outdoor environment, there are not as many AR applications as those for indoor. We have been witnessing the sensation around the popular AR application Pokemon Go [5] as well as the rise of smart-glasses and AR navigation system. Hence, it is necessary to complement people's AR experience by also performing augmentation for outside environment. Apart from landscapes, buildings, boards and roads, the sky can also be a suitable subject to be augmented because it encapsulates the whole view above the ground while also has many distinct characteristics that can benefit AR design and development. Furthermore, sky augmentation can be used for multiple purposes such as illustrating astronomical events especially for places where they are less likely to occur, visually learning about weather phenomenon, doing world navigation, object caption or virtual observation of aircrafts, fireworks and many more possible artistic content.

In sky segmentation, we wish to make virtual objects appear to be put on the sky. Since objects at great distance have different perspective effects compared to those which are near to human eyes, sky augmentation can be carried out in a different manner from indoor AR. Some key challenges are: 1) virtual objects must appear behind ground objects to maintain occlusion, which requires real-time fine sky segmentation; 2) virtual objects must appear in harmony with real sky scenery including clouds, flying objects, the sun as well as weather and lighting condition; 3) virtual camera and physical camera streams must be alignment in a concordant manner so that virtual world could look realistic enough.

In this paper, we propose an approach to achieve sky segmentation which addresses the aforementioned challenges. We prototype our ideas in Unity3D game engine and then build the project as an Android application. The application "Fakeye" is shown in Figure 1.

2. Related Work

2.1. Solutions to occlusion in AR

Originally, AR always renders objects on top of real environment which does not provide immersive experience since they should obey occlusion. Besides high-end devices with depth sensors, depth computing from 2D imagery is an ongoing research. Wloka et al. propose a new version of stereo matching algorithm to compute depth information for AR occlusion which favors speed over accuracy [7]. Livingston et al. conduct multiple experiments and user surveys on different rendering styles, opacity, intensity, etc. in order to devise a guideline on human depth perception [3]. Their works aim to benefit large-scale outdoor AR navigation system where road instruction should be all visible to some extent, hence strict occlusion is not mandatory.

2.2. Methods to achieve outdoor/large-scale AR

To correlate the two physical and virtual worlds in large-scale AR, a complex environment perception using computer vision and/or a fusion between inertial measurement unit (IMU) and GPS unit are often used [4]. Likewise, in our work, we must acquire and process information about the sky from camera stream and rely on rotation signal to match up the two real and virtual views.

3. Real-time sky segmentation

In order to create occlusion between physical ground objects and virtual sky objects, we choose to perform image segmentation to obtain a mask for sky region to let virtual content appear upon it. As far as we are concerned, the sky area in an image has very unique features such as pixel position and colors. Since speed is critical for sky augmentation experience, our criteria for segmentation algorithm are that it should not slow down the rendering pipeline while at the same time provide adequate accuracy. To leverage such distinct features of the sky and meet the criteria, we choose to implement a naive machine learning method which is pixel-wise logistic regression with four features: pixel's y-coordinate and red-green-blue color channels. We are also interested in natural clouds for the final augmented scene so we label them as sky pixel when building the segmentation model.

4. Real-and-virtual scene blending

After having known the sky area in a frame, we try to preserve the sky's texture by blending the physical scene with virtual scene rather than completely replace it. The texture blending workflow is presented in Figure 2. We first apply weighted grayscale to the sky area to preserve contrast. Then an image blending algorithm is performed. For experiments purpose, we implement several Photoshop-like

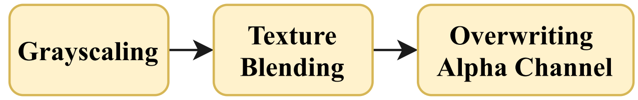

Figure 2: Scenes blending process.

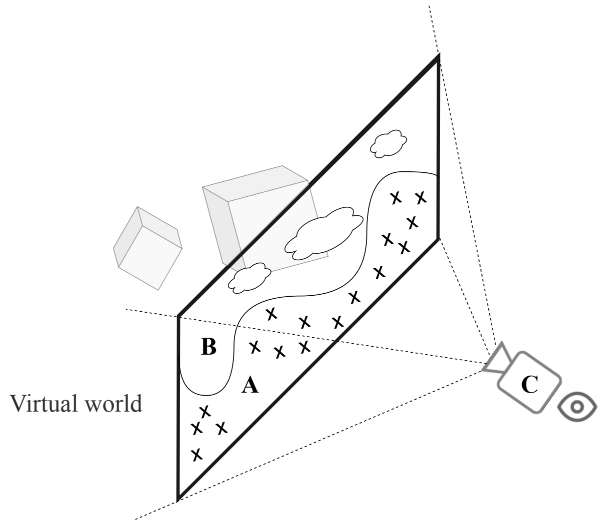

Figure 3: Fakeye's project setup.

image blending algorithms on each of the red, green and blue channels. We find blending on red channel to produce visually beautiful result. In the final step, we overwrite the alpha channel of the real sky texture with the above computed blending value. As a result, this method let virtual objects appear to be surrounded by clouds and have appropriate opacity as shown in Figure 1 and Figure 4.

5. Real-and-virtual camera alignment

One of the major objectives of AR is that virtual camera and physical camera must have corresponding streams in term of spatial perception so that virtual objects look real enough regarding to those in physical world. As we try to make objects looked as if they are far away aside from occlusion, we must create an illusion such that they are stationary with respect to viewer camera. This turns out to be much simpler since we do not need to track user's camera translation but only rotation. This kind of signal can be independently read from a gyroscope integrated on smartphones. However, this would be error-prone to reading noise and therefore should be improved by fusing with some other sensors such as accelerometer and gravity sensor. In addition to rotation, virtual camera's field-of-view (FOV) also matters for which it must have the same configuration as physical camera. Otherwise, digital content may appear to rotate faster (zoom-in effect with smaller FOV) or slower (zoom-out effect with larger FOV) than real objects.

Figure 4: Virtual cube laying among the clouds.

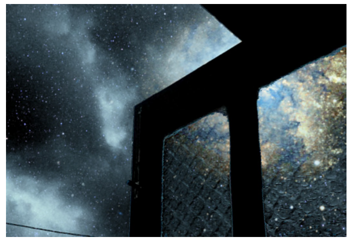

Figure 5: Virtual objects rendered through glass door.

Figure 6: Bad augmentation caused by false segmentation.

6. Experiment

The application Fakeye's scene setting is shown in Figure 3. Camera **C** is the virtual camera where user views the final augmentation scene. A rectangular plane is stationary with respect to the camera view as a render canvas. Physical camera's input is streamed as the top texture of the plane.

6.1. Segmentation process

Sky segmentation is performed in the top texture which results in a mask where region of non-sky object **A** is visible with full opacity. For the pixel-wise logistic classifier, we train the model separately from the application with SkyFinder dataset. The dataset contains approximately 100,000 sky-annotated images in which we only use bright daytime sky. We train multiple models with different sets of the five features including pixel's x-coordinate (X), pixel's y-coordinate (Y), red channel (R), green channel (G) and blue channel (B). Our best model which is the four-feature YRGB gains 76.89% in mean Intersection-Over-Union (mIOU). However, its Misclassification Rate (MCR) is also high with 8.61%. The train weights are then used in the application to determine if a pixel should be of sky region.

6.2. Blending process

After segmentation process, sky region **B** is the one which enters texture blending process. Here the top texture will be blend with the base texture of the virtual camera stream. After that, the blending process changes region **B**'s opacity to values which should be less than or equal to 1.0 (for 1.0 is full opacity). With such array of alpha values, the objects of the virtual world can be visible behind the viewer plane and simultaneously appear to be blending in with the real sky.

6.3. Camera alignment

The viewer plane's scale ratio must match with the phone's screen ratio so that objects are rendered correctly in term of scaling. The Android device we use is a Pixel 2 XL with 2880$\times$1440 resolution so the plane scale ratio $x : y : z$ should be $2 : 1 : 1$. Virtual camera **C** is attached with a script controlling its rotation by phone's gyroscope signal. The virtual camera's FOV is approximately 50.7° calculated based on the test device camera specification.

7. Result

Our implementation shows a basic workflow to achieve sky augmentation in real-time camera stream. Figure 1 shows correct sky segmentation and object rendering. Another capture in Figure 4 shows the blending effect where object appears to be surrounded by the clouds. Texture blending also benefits rendering through transparent material as shown in Figure 5. Since our sky segmentation model is a simple logistic classifier, the segmentation result is not much accurate. Therefore, it causes unwanted holes in real scene as seen in Figure 6.

8. Conclusion

It is essential to carry out AR in large-scale environment in order to enhance overall user experience. Sky augmentation has many meaningful application and is achievable with less demanding requirements in comparison to indoor AR. Although our proposed approach can be implemented with ease on gyroscope-enabled smartphones, it still needs much improvement on the segmentation process, lightning and gyroscope reading. For sky augmentation to be more immersive, it should be accompanied with spatial sound in addition to artificial visuals and be viewed with AR specialized devices.

9. Acknowledgments

This research is supported by research funding from Advanced Program in Computer Science, University of Science, Vietnam National University - Ho Chi Minh City.

References

[1] Swati V. Birje. Marker based augmented reality using android os. 2013.

[2] V. Ferrari, T. Tuytelaars, and L. Van Gool. Markerless augmented reality with a real-time affine region tracker. In *Pro-*

ceedings IEEE and ACM International Symposium on Augmented Reality, pages 87–96, 2001.

[3] M. A. Livingston, J. E. Swan, J. L. Gabbard, T. H. Hollerer, D. Hix, S. J. Julier, Y. Baillot, and D. Brown. Resolving multiple occluded layers in augmented reality. In *The Second IEEE and ACM International Symposium on Mixed and Augmented Reality, 2003. Proceedings.*, pages 56–65, 2003.

[4] T. Oskiper, S. Samarasekera, and R. Kumar. Multi-sensor navigation algorithm using monocular camera, imu and gps for large scale augmented reality. In *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pages 71–80, 2012.

[5] Janne Paavilainen, Hannu Korhonen, Kati Alha, Jaakko Stenros, Elina Koskinen, and Frans Mayra. The pokémon go experience: A location-based augmented reality mobile game goes mainstream. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, page 2493–2498, New York, NY, USA, 2017. Association for Computing Machinery.

[6] R. Paucher and M. Turk. Location-based augmented reality on mobile phones. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Workshops*, pages 9–16, 2010.

[7] Matthias M. Wloka and Brian G. Anderson. Resolving occlusion in augmented reality. In *Proceedings of the 1995 Symposium on Interactive 3D Graphics*, I3D '95, page 5–12, New York, NY, USA, 1995. Association for Computing Machinery.