The University of Tokyo
Graduate School of Engineering

A DEPTH CUE METHOD BASED ON BLUR EFFECT IN AUGMENTED REALITY

拡張現実感におけるブラーエフェクトに基づく奥行き表現方式

A Thesis in
Department of Electrical Engineering and Information Systems
By

Lin Xueting

Master of Engineering

August 2013

The University of Tokyo
Graduate School of Engineering

A DEPTH CUE METHOD BASED ON BLUR EFFECT IN AUGMENTED REALITY

A Thesis in
Department of Electrical Engineering and Information Systems
by
Lin Xueting

Master of Engineering

August 2013

Abstract

The motivation of this research is to enhance the user experience by offering better depth interpretations in augmented reality systems. The thesis focuses on how to use blur effect to enhance the depth interpretations since the blur effect has been proved to be an independent depth cue to human visual system and the proper blur in the virtual scene could increase the consistency in augmented reality. In this thesis, the perceptual problems in video see-through augmented reality applications such as how to generate a consistent perception between virtual and real parts, how the blur effect act as a depth cue are addressed. The whole thesis is structured to solve three main problems: 1. how to determine the degree of blur which should be rendered on the virtual objects; 2. how to render the certain degree of blur onto the virtual objects; 3. how the users would perceive the blurred virtual contents as a depth cue.

To solve the first problem, a new depth cue method based on blur effect is proposed. Different from the previous approaches, the proposed method offers an algorithm which estimates the blur effect in the whole scene based on the spatial information in the real world and the intrinsic parameters of the camera. To solve the second problem, a prototype AR system was designed and implemented. The prototype system realized the proposed depth cue method and the system design and implementation details are introduced in this thesis. The algorithm of the blur shader is discussed.

Three user tests were conducted to solve the problem how the users would perceive the blur effect rendered by different blurring methods. The user tests results are discussed in this thesis. It could be confirmed from the user test results that the proposed depth cue method could help the user to gain a better depth interpretation in AR system. In the user tests, a comparison between the proposed method (which estimates the blur effect in the whole scene based on the measured blur radius of some known position) and previous method (which only renders blur effect on the position whose blur radius is measured) is addressed. However, the test results could not confirm which method is better.

In the future work, three aspects are considered: 1. for the proposed depth cue method, besides the blur effect caused by defocusing of the camera lens, the motion blur should also be investigated; 2. for the prototype system implementation, a moving camera should be introduced and the real-time detecting and rendering should also be discussed. The camera view of the prototype system will no longer be limited to a desk but expanded to various situations; 3. for the user tests, more researches should be addressed to investigate on how the blur effect would match depth interpretation in quantity.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I am very grateful to my instructor, Professor Ogawa Takefumi, for his instruction in the last two years. With his help, I got a wide knowledge in the field of augmented reality and dug into the interesting topic of depth cue method among the perception problems. Prof. Ogawa taught me how to find a topic out of my own interests in science technology and how to push the idea into not only a research but also a prototype which could be used by normal users.

I am thankful to the professors and teachers in Computer Network Laboratory (CNL) who have instructed me: Prof. Wakahara, Prof. Nakayama and Dr. Miyamoto. They offered me lots of advices on how to do a scientific research, how to make an impressive presentation, how to write a convincing paper and so on. Although their research fields are different from mine, the methodology of how to solve the problems would always be helpful to me.

Here I would also like to thank all my lab mates: Zilu Liang, Meng Wang, Xinru Yao, Okasaki, Sagawa, Yonezawa, and my seniors Nijima, Otsuka and Ishikawa. Without their help, I could hardly move any forward in my research as well as my student life in Todai.

Last but not least, I would like to thank my parents who are thousands miles away in Shanghai. I know they have come though a hard time but they never complained to me. Sometimes people could not stay together for some reason, but the distance will never change the love in their hearts.

This research was supported in part by a Grant-in-Aid for Scientific Research (C) numbered 25330227 by the Japan Society for the Promotion of Science (JSPS).

Chapter 1

INTRODUCTION

1.1 Background and Motivation

Augmented reality (AR) is a realtime, direct or indirect, view of a real-world environment whose elements are augmented by computer-generated sensory input such as sound, video, graphics, GPS data, smell or even haptic."Augmented reality" is not a new concept. The first application of AR appeared in the late 1960s and 1970s. This AR application is a motorcycle simulator called Sensorama with visuals, sound, vibration and smell[36]. In the past, augmented reality (AR) was always regarded as a variation of virtual reality (VR). Virtual reality is the technologies that immerse a user inside a synthetic environment and the users cannot see the real world around him. In contrast, AR allows the user to see the real world, with virtual objects superimposed upon or composed with the real world. AR supplements reality, rather than completely replacing it[3].

In recent years, with the development of hardware components, graphic processors, HD and 3D displays, sensors and input devices, AR technologies are enjoying a growing popularity in a variety of applications. Especially after the commercialization of head-mounted display (HMD) (invented in 1968 by Ivan Sutherland[43]; the wearable device is not widely used until recent years), smart phone and projection glasses, the researches or developments on new platforms for AR technology keep being a hot topic. Now augmented reality is regarded as a wider concept. It is also considered as a part of "mixed reality". In Figure 1.1, the relationship between the AR technology, VR technology and MR technology is showed. The biggest figure of augmented reality technology is that it shows a strong connection with the reality world. Following the definitions given in [3] and [2], the following aspects should be noticed when discussing augmented reality:

1. AR combines real and virtual objects in a real environment. The combination is not limited to the sense of vision, but also contents sense of sound, smell, touch and so on. This thesis only focuses on the sense of vision. There are normally two kinds of display screen in AR system, the video see-through screen and the optical see-through screen. The optical

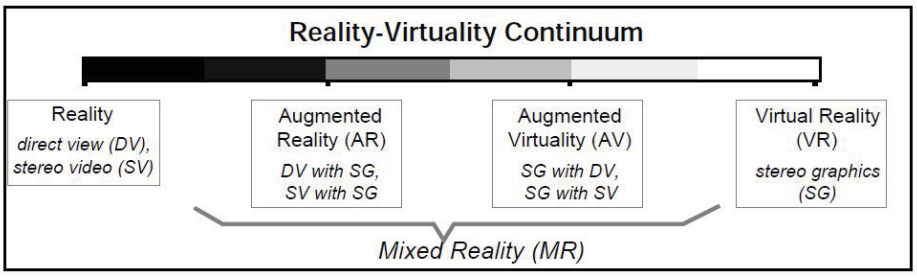


Figure 1.1: Simplified representation of the Reality- Virtuality Continuum[8]. The stereo graphics (SG) are also known as computer graphics (CG). These are the virtual contents appearing on the screen. The real world is presented by direct view (DV) though optical see-though displays or by stereo view (SV) though video see-through displays.

see-through screen indicates the see-through screen or projection glasses and the video see-through screen is the normal display often used in daily life such as the computer screen or the smart phone screen. All the discussions made in this thesis are based on the visual see-through screen.

2. AR registers (aligns) real and virtual objects with each other. The registration makes AR different from computer graphics. Computer graphics (CG) technology focuses more on the rendering method in computer systems. However, AR technology also discuss tracking technologies which are also the main topics in computer vision (CV).

3. AR technology runs interactively. In recent applications, moving-cameras, HD cameras and depth cameras are often used in AR application. On the other hand, with the progress in 3D computer graphics, 3D virtual objects became the main contents in AR applications. In the same time, whether the registration of the real world and the rendering of virtual contents could be conducted in real time become one of the biggest challenges in AR applications.

Besides products in entertainment and gaming industry, the AR applications are now widely used in all kinds of industries. In education, LEGO, the Denmark toy maker began to make toys with augmented realty app. In training industry, driving classes using AR equipment help many people practice their driving skills. Training AR systems are especially popular in military training. Using a HMD helmet, the soldiers could have a better sense of reality even when there is no live ammunition. In medical care industry, augmented reality technologies are sometimes used in the rehabilitation for patients. In construction industry, augmented reality models are becoming a popular way to present the architecture plan. However, although the researches and developments of AR technology are fruitful in the last few years, the fundamental problems in the design of effective AR still remain.

The fundamental problems could be categorized into three aspects: 1. how to combine the real and virtual world; 2. how to render the virtual contents; 3. how to interact in real-time. One of the popular way to solve the first problem is to introduce markers into AR applications. Nowadays, markers are the main stream for most of the practical AR applications. Only a small

Figure 1.2: A typical marker in AR application. This is one of the sample markers in ARToolkit. ARToolkit is one of the first project in using the marker to make the easy registration for AR applications[17].

amount of AR applications keep using marker-less registration method. A typical maker used in AR application is a black-white square marker with some pattern inside it, as shown in Figure 1.2. The shape of square makes it easy to track the corners and construct the three-dimension coordinate. The inside pattern makes it easy to recognize the difference between the markers. The second problem is a common problem in VR technology or computer graphics (CG) technology. The third problem is highly related with the tracking and rendering algorithms. Sometimes it is also limited by the performance of the hardware.

On the other hand, how to help the users to percept the correct information in AR is a big and complex problem. Some of the problems in perceptually correct augmentation could be traced to issues with depth and illumination that are often interconnected, or by issues related to the display of the direct view such as viewing angle offset. These problems may cause scene and depth distortions, and visibility issues, which can potentially lead to poor user experience in AR application. Some of these issues claimed above result from technological limitations, for example, the graphic card can hardly render photorealistic 3D virtual objects in real time[7]; or the HMD limited the users' view of scene. However, many are caused by limited understanding or by inadequate methods on how to display information in AR system properly.

Among the perceptual issues, the incorrect depth interpretations degrade users experience significantly[22]. The depth interpretations in augmented reality refer to egocentric depth perception and exocentric depth interpretation, as shown in Figure 1.3. The egocentric depth per-

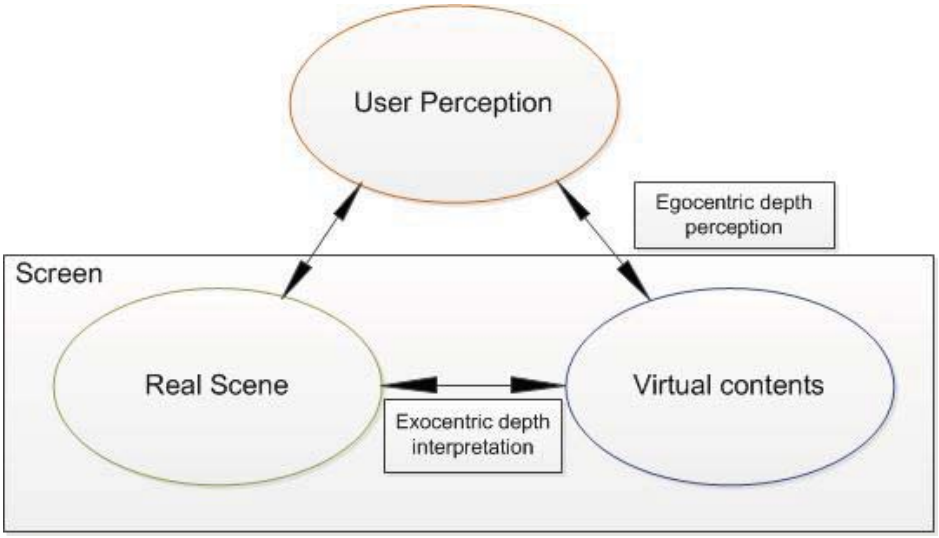


Figure 1.3: Spatial relationship in Augmented Reality. The depth interpretations in augmented reality refer to egocentric depth perception and exocentric depth interpretation. The egocentric depth perception includes the user's first-person perspective and the depth perception of the contents in view.

ception includes the user's first-person perspective and the depth perception of the contents in view. The exocentric depth interpretation refers to the depth consistence between the real scene and the virtual objects (overlaid computer graphic contents). In many AR applications, such as medical AR systems (see the book *Medicine Meets Virtual Reality* [47]), architectural AR systems or AR navigation systems, there is a need for the users to understand the depth relation between the virtual object and the real view or at least understand the display order of the objects. On the other hand, with the popularity of hand-hold or wearable display devices such as the HD camera attached smart phone or projection glasses, the users would pursue a more comfortable experience with less perceptual errors. Therefore, the depth misinterpretation has become one of the biggest problems among the perceptual issues and reduces the user experience in AR technology.

Table 1.1: Categories of depth cue methods.

Pictorial Depth Cues	Occlusion
	Linear perspective
	Texture perspective
	Relative brightness
Kinetic Depth Cues	Motion parallax
Physiological Depth Cues	Binocular disparity
	Binocular convergence
	Accommodation focus

In order to improve the depth interpretation for general users, many methods have been

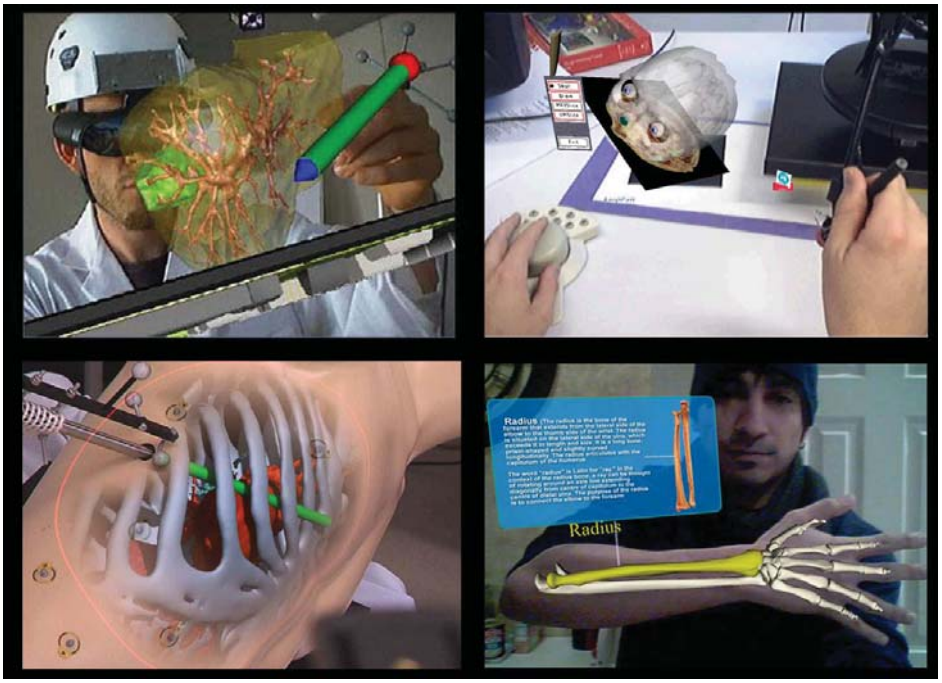


Figure 1.4: Augmented reality technology used in medical professionals[12]. AR technologies, such as AR x-ray vision are now quite popular in practical use as well as education. The depth interpretation in these applications shows important influence on users' perception.

Figure 1.5: IPad-based augmented 3D-printing models. The 3D-printing model is used as a marker. The whole AR system was invented as a new way to show architectural plans[14].

Figure 1.6: Navigation AR smart phone application[15].

invented basing on the perceptive characters of human. Table 1.1 shows the basic depth cue methods usually concerned in computer graphics. Occlusion and Linear perspective are often used in augmented reality systems in order to indicate the order of the virtual object and the real object([6][10][41][39]). However, it is not always possible to insert the virtual object into the real view. Indicating depth ordering based on the contours or the size of the virtual object (linear perspective cue) sometimes lead the observer a misunderstanding of the depth because the linear perspective is affect greatly by the reference scene and can hardly act as an individual depth cue. In virtual reality, relative brightness, shading and shadowing have been well investigated [42], however, some researches, such as [35] showed that the high fidelity of the virtual object could hardly help the observer to have a better performance. Binocular stereo parallax and motion parallax become popular recently [24] [16] [49]. However, the extra need of hardware makes it still far from practical application. Texture perspective means changing the visual information on the texture by blurring it, decreasing the contrast or other methods. The researches in biology and psychology have proved that the texture perspective could act as the depth cue independently[1][45].

In video see-through augmented reality, the seamlessness of the real and virtual worlds is affected by the differences in image quality between the superimposed virtual objects and the real background. This problem is caused by the difference between real and virtual camera model. The ideal camera model in which the image quality is not degraded by the defocus or motion blur is used when rendering virtual objects in AR applications. If the virtual objects could be rendered considering the real camera model, the AR application might be able to provide a more

Figure 1.7: Previous research: Sketchy AR. Equipped with a handheld visor, visitors can see the real environment overlaid with virtual objects with both the real and virtual content rendered in a non-photorealistic style[11].

correct depth cue. Blur effect is one of the methods to rendering texture perspective in computer graphics. It is a common sense that cameras and eyes have limited depth of focus, so images of objects nearer or farther than the point of focus or fixation are blurred. The limitation of frame frequency also causes blur. This kind of blur is called motion blur. Actually, in augmented reality, the moving camera or the moving virtual contents are both the reason for motion blur. Several previous researches have been conducted based on applying blur effect or changing the quality of texture to the stereo view so as to provide the users a better depth interpretation. One of the methods is to render the real view as well as the virtual object to a cartoon like view [9][11]. In this approach, the consistency of the real view and the virtual object was increased while the egocentric depth judgment might be largely affected because the stereo view's quality was also decreased. Another approach is to capture the blur level of the real object in the stereo view first and then render the same level of blur to the virtual object[30]. However, one important drawback of this proposed AR system is that the blur effect could only be rendered on the marker where the blur in the real scene is measured. This limits the practicality of the system since in many applications the virtual contents are out of the marker's region or even moving.

The motivation of this research is to enhance the user experience by offering better depth interpretations in augmented reality systems. The thesis focuses on how to use blur effect to enhance the depth interpretations since the blur effect has been proved to be an independent depth cue to human visual system and the proper blur in the virtual scene could increase the consistency in augmented reality. In this thesis, the perceptual problems in video see-through augmented reality applications such as how to generate a consistent perception between virtual and real parts, how the blur effect act as a depth cue are addressed.

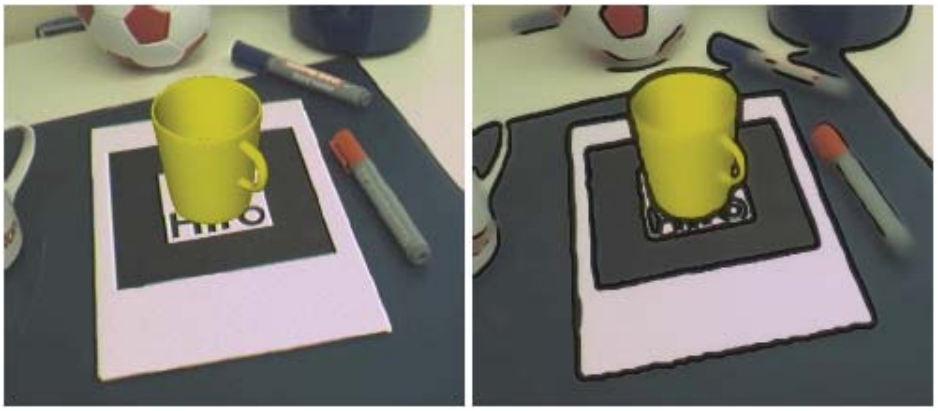


Figure 1.8: Previous research: Stylized AR. The research attempts to create similar levels of realism in both the camera image and the virtual objects with the help of artistic or illustrative rendering and image filtering[9].

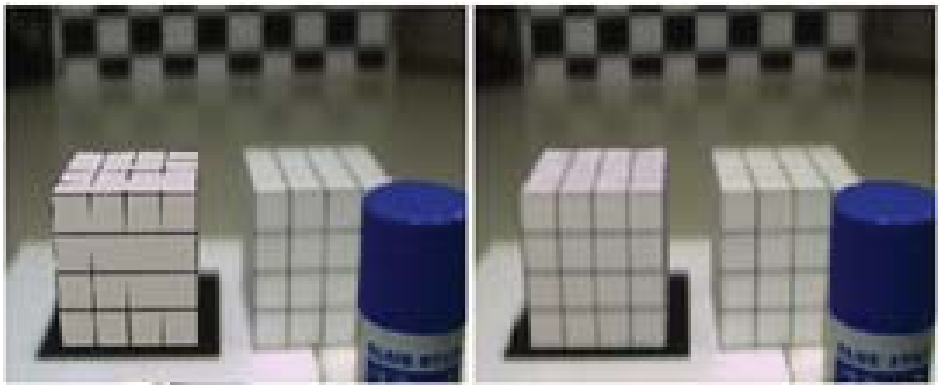


Figure 1.9: Previous research: Detecting-blurring AR. The blur level of the real object in the stereo view is first detected and then the same level of blur is rendered to the virtual object[30].

1.2 Goal, Purpose and Contribution of the Research

The purpose of this thesis is to conduct further discussion on how to use the blur effect as a depth cue to enhance the user experience in AR applications. The whole thesis is structured to solve three main problems:

1. how to determine the degree of blur which should be rendered on the virtual objects;
2. how to render the certain degree of blur onto the virtual objects;
3. how the users would perceive the blurred virtual contents as a depth cue.

In this thesis, the solutions of the three problems listed above are discussed. A new method to estimate the degree of blur that based on estimating the Point Spread Function (PSF) parameter all over the scene of view is proposed. Referring to the thin lens camera model and the intrinsic

parameters of the camera, such as the focus length and the size of CCD (Charge-coupled Device, the image sensor for digital imaging), the method makes it possible to estimate the degree of blur that varies spatially with the scene, and to render the degree of blur to virtual objects. To solve the second problem, a prototype AR system was designed and implemented. The prototype system realized the proposed depth cue method and the system design and implementation details are introduces in this thesis. The algorithm of the blur shader is discussed. The thesis focuses on the users perception issue. User tests were conducted to figure out how the users would perceive the blurred virtual contents as a depth cue. It could be confirmed from the user test results that the proposed depth cue method could help the user to gain a better depth interpretation in AR system. In the user tests, a comparison between the proposed method (which estimates the blur effect in the whole scene based on the measured blur radius of some known position) and previous method (which only renders blur effect on the position whose blur radius is measured) is addressed. However, the test results could not confirm which method is better.

1.3 Organization of This Thesis

This thesis is divided into six chapters. In Chapter 2, the related works is introduced. The related technology topics are covered. The technical concepts in AR applications are listed up and the researches related to our proposed method are introduced. The proposed depth cue method based on blur effect is introduced in Chapter 3. In this chapter the algorithm of the blur estimation method is presented. Chapter 4 introduces how the prototype system of the proposed method is designed and implemented. Based on the prototype system, the user study is carried on in Chapter 5. In the end, the conclusion and future work of the research are discussed in Chapter 6.

Chapter 2

RELATED WORKS

2.1 Overview

In the previous researches on augmented reality, researchers have studied on how the depth cues would affect the perception of users. The approaches are mainly in three kinds of depth cues: 1. pictorial depth cues, including occlusion, linear perspective, texture perspective and relative brightness; 2. kinetic depth cues, including motion parallax; 3. physiological depth cues, including binocular disparity, binocular convergence and accommodation focus. The blur effect is one of the factors in texture perspective. Although, the blur effect was not regarded as the main factor of the depth cues in augmented reality, in the studies of perceptual science, the blur effect has been proved to be among one of the strongest depth cues in human visual perception system.

There were several researches on using depth cue to increase the consistency of the visual objects and the real scene in the field of augmented reality. However, none of them gave a convincing study on how the blur effect could be induced as a depth cue. In this chapter, I would introduce several related researches in other research fields. First, I will go through what role the blur effect plays in human's depth perception. Secondly, I will introduce how the blur effect is used in computer graphics and virtual reality. Thirdly, I will introduce the main researches related to blur effect in the field of augmented reality. In the end, I will discuss how the anti-blur technology in computer vision could become an inspiration to solve the problem of how to decide the degree of depth cue in augmented reality.

2.2 Blur Effect as A Depth Cue in Human Perception

When asking about why the blur effect could be used as a depth cue, it is a common answer that the object in the farther places is blurred and the object in the nearer places is clearer. The answer comes from our common experience. In fact, this common sense comes from our

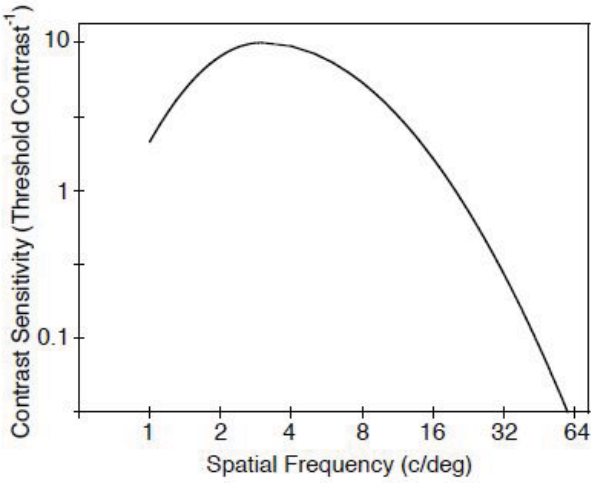


Figure 2.1: Band-pass approximation to a human contrast-sensitivity function for a low-contrast, high luminance, static scene[25].

characteristic in perception.

In human visual system, the image formed on our retina could be regarded as the serious information in terms of spatial frequencies. A clear texture in our perception is the one which contents a larger amount of high spatial frequency. The manifestation of it is that there are many edges and corners in the texture. On the contrary, a blurred texture in our perception is the one which contents a larger amount of low spatial frequency. If we make the images formed on our retina all in frequency domain, we will find that each image is found of both the so-called "high spatial frequency" and the "low spatial frequency" contents. However, unlike the computer vision, we human could not percept each channel of the spacial frequency in the same way. The researches in visual perception sensitivity have found that human have different sensibility to visual contents in different spatial frequency. This characteristic even makes it possible for human to perceive the same virtual contents differently in various situations. Figure 2.1 shows a typical contrast sensitivity function. The higher the contrast sensitivity is, the more sensitive human is to this channel of spatial frequency. If a user is given a test of an image setting from near to far, it is easy to find that in the near distance, the channels in the high frequency region hold the main role in perception, while in the far distance region, the channels in the low spatial frequency hold the leading role in the perception. Based on this perception difference, blur effect, emphasizing on the low spatial frequency, shows a potential of acting as depth cue.

In Dhanraj's research[45] and Georeg Mather's study[26] on the blur effect, the blur effect was proved to have an obvious effect on depth perception. Mather indicated that the blur variation acts as an effective cue to depth: if one image region contains sharply focused texture, and another contains blurred texture, then the two regions may be perceived at different depths, even in the absence of other depth cues. In Figure 2.1, the blurred black rectangle seems farther in depth. In some other researches, the blur effect also shows an effect for reducing eye strain in

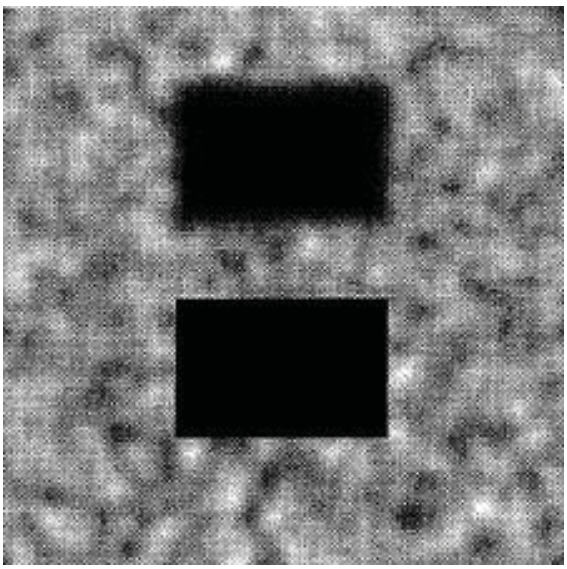


Figure 2.2: A 2D image shows the 3D perception of depth. The lower black rectangle appears nearer than the upper rectangle, because it has sharply defined edges[26].

stereoscopic displays[23] and directing attention in visual displays[44]. These perception issues all show the great potential of inducing blur effect as a depth cue in augmented reality.

However, the blur effect generally comes with two effects. The first is the contrast reduction and the other is the edge blur. O'Shea's research[32] proved that blur can act as a depth cue independently from contrast[27]. The researches on edge blur[28] show that the edge blur plays an important role in rendering blur as the depth cue. These two points will be later discussed in the implementation part.

2.3 Blur Effect in Computer Graphics

In the field of computer graphics, it has been a long history studying on the topics of rendering blur effect to virtual contents. It is a common sense that cameras and eyes have limited depth of focus, so images of objects nearer or farther than the point of focus or fixation are blurred. In the moving scene, the limitation of frame frequency also causes blur. Rendering the blurring effect to an image means eliminating the high space frequency part of the image and keep or emphasis on the low space frequency part.

In the previous researches in computer graphics, it has been proved that in order to make a virtual object consistent to the real scene, it is necessary to consider the relative brightness, shading, shadowing and other factors. The blur effect is also one of the factors. However, with only the blur effect, it is impossible to make the virtual object seem real. In fact, selective image blurring is very commonly used in photographic and video for establishing the impression of depth. Although blur effect is not regarded as the key factor in rendering a virtual object.

It actually can act as a monocular depth cue even when all other cues are removed[32]. Some researches, such as [45] and [46], on how the blur effect affect the perception of egocentric distance provided psychophysical evidence of the role of blur gradients in distance perception. Some experiences in [45] show that when extra blur gradient is rendered to normal images, the viewer will get illusion of the depth perception.

Although the blur effect keeps a hot topic in computer graphics, in augmented reality we should concern not only on how to create a fantastic image but also on how to merge the virtual world into the real world. The first question we should ask is that will the blur effect on the screen represent the certain distance in the real world? In Harley's research[25], a distance-as-filtering hypothesis was raised. Experiments were conducted and the results showed that blurring the image will mimic its spatial-frequency composition at certain distance of the real world. This result shows that by applying blur effect in a right way, the user could get the same depth perception they have to the real objects in the real world. This research shows the potential of using blur effect as depth cue in augmented reality systems.

2.4 Related Researches in Augmented Reality

In Fischer's research[9], a cartoon like view was rendered so as to take advantage of the blur effect to increase the consistency. In Haller's research[11], a loose and sketchy view was rendered for the same purpose. In these approaches, the consistency of the real view and the virtual object was increased while the egocentric depth judgment might be largely affected because the stereo view's quality was decreased. Therefore, in order to make a better depth perception, it is better not to affect the quality of the real scene.

From 2006 to 2008, Okumura Bunyo's group had conducted deep research on how to use the blur effect to render seamless augmented reality scene. In his research of Geometric and Photometric Registration Considering Image Quality for Augmented Reality, the photometric registration is regarded as important as geometric registration. He supposed that the difference in image quality between a real image and virtual objects decreases the seamlessness for users because the real image is degraded in quality by blur effect caused by the imperfect optical system of a camera. To increase the seamlessness for users in AR application, his group proposed a method to correct the difference in image quality between the real and virtual objects using image markers.

The proposed method focus on the augmented reality systems in which virtual objects are composed around image markers in the real world. They assumed that the blur effect of virtual objects is represented by the degree of blur at the marker because the point spread function (PSF) is related to the depth of the target object and the change of depth is not large around the marker in most case[31]. Their proposed method can be divided into 4 steps.

(A) Detect the marker. A square marker with known shape and color is used. By fitting straight line, each edge of the marker is detected. The corners of marker are estimated by calculating intersections of detected lines.

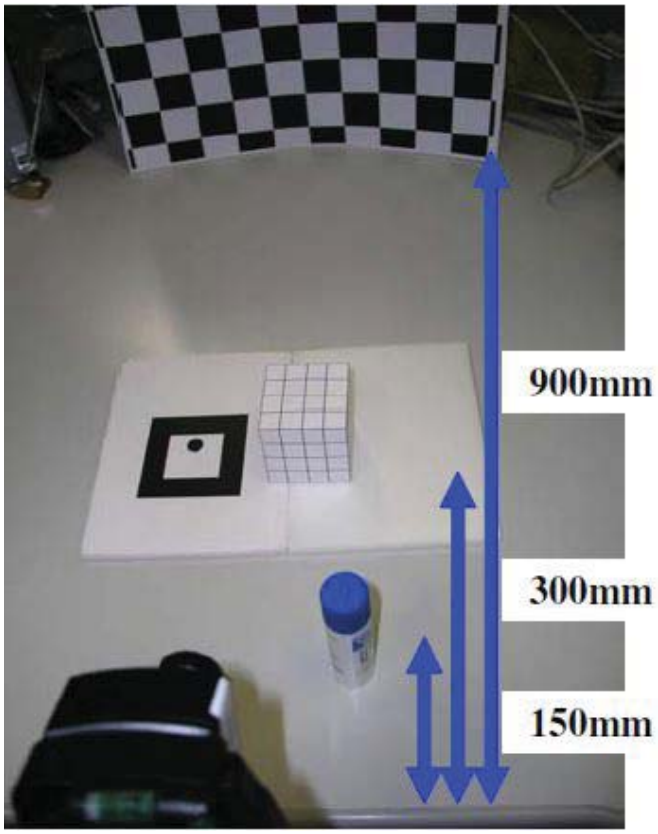


Figure 2.3: Experiment scene of the prototype system in Okumura's research.

(B) Estimate the blur parameter from captured image. The size of blur is estimated from the change of intensities along the direction orthogonal to the edge of image marker.

(C) Estimate the camera position and orientation in consideration of blur effects. The camera position and orientation are estimated from position of corners of the marker and the restoration of the geometrical features of maker edges is used so that the process are not influenced by the blur effects.

(D) Render virtual objects with blur effect. The rendering and simulation of blur effects are carried out using graphic hardware.

Figure 2.3 and Figure 2.4 show the prototype system and the rendering result of Okumura's research. The result shows that the proposed method could improve the seamlessness of the scene. By rendering a certain degree of blur on the virtual object, the over-clear virtual object turns to seem fit into the real scene.

In Okumura's research, the blur effect caused by defocus as well as motion blur are considered. The other aspects in photometric registration such as lighting and shading are not considered during the rendering process. The limitation of the research is that the blur effect could only be

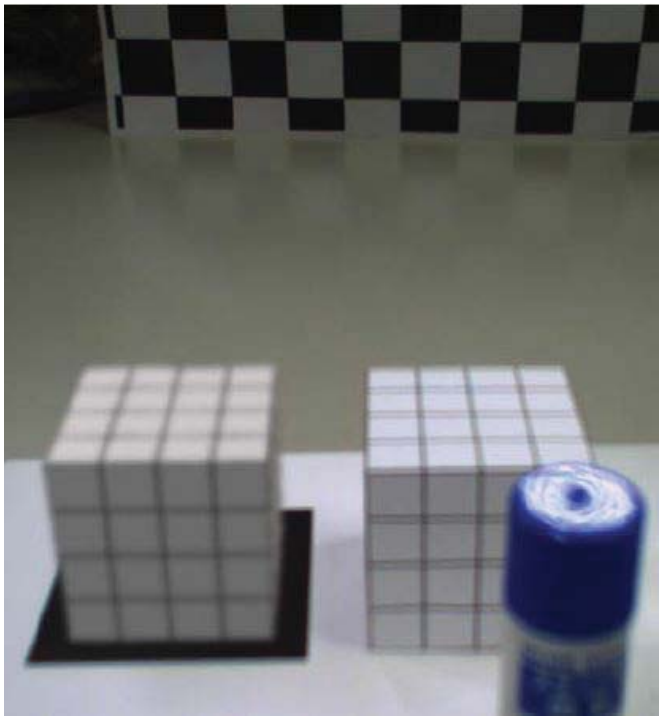


Figure 2.4: The rendering result of Okumura's prototype system.

rendered to the virtual objects which were composed around the image marker. In many of the AR application now, the virtual objects are rendered out of the marker region and sometimes the virtual objects are moving around in the whole scene. In the evaluation, Okumura's group offered plenty of data to evaluate the estimation method of camera position and orientation. He also conducted a comparison between real and virtual objects by comparing the difference in the size of defocus as well as in the size of motion blur.

2.5 Depth and Defocusing

In the previous paragraphs, the blur effect and the related works in computer graphics and augmented reality have been discussed. In the normal camera optic systems, the blur effect is mainly caused by two reasons: 1. defocus of the camera and 2. motion blur during the shoot. In this research, I will only discuss the blur caused by defocusing.

There are two branches in computer vision which share strong connection with the defocusing problem: 1. autofocus and 2. image recovery. Autofocus systems have become the main trend in the camera systems these days. The autofocus systems rely on one or more sensors to determine correct focus. During the autofocus process, the system analysis the defocus and estimate the depth for the focus zooming[48]. Many method to do the estimation has been raised. In Masashi's research[4], a new camera model accounting for both effects of defocus and lens center translation

by zooming is introduced. In fact, the thin lens model for optical systems is often used. In Pentland's research[33], a depth-from-convergence model was raised. In this model, the camera parameters are changed to measure depth at a single point, whereas the error signal (blur and disparity) is utilized to estimate depth. Most biological lens systems are exactly focused at only one distance along each radius from the lens into the scene. The locus of exactly focused points forms a doubly curved, approximately spherical surface in three-dimensional space. Only when objects in the scene intersect this surface is their image exactly in focus, objects distant from this surface of exact focus are blurred, an effect familiar to photographers as depth of field. The amount of defocus or blurring depends solely on the distance to the surface of exact focus and the characteristics of the lens system; as the distance between the imaged point and the surface of exact focus increases, the imaged objects become progressively more defocused[33]. In this way, if the amount of blur at a given point in the image could be measured, it seems possible to use the parameters of the lens system to compute the distance to the corresponding point in the scene. The model derived from this idea is called smooth gradient of focus as a function of depth which is showed below.

$$D = \frac{Fv_0}{v_0 - F - \sigma f} \tag{2.1}$$

where v_0 is the distance between the lens and the image plane, f the f-number of the lens system, F the focal length of the lens system., and σ the spatial constant of the point spread function (i.e., the radius of the imaged point's "blur circle") which describes how an image point is blurred by the imaging optics.

Since the defocus and depth share a linear relation, it is possible to use the depth information to estimate the defocus too. The details about this algorithm and the derivation from the thin lens model will be discussed in Chapter 3.

2.6 Discussion and Conclusions

The previous researches proved that blur effect is more than a factor to improve the consistency of the virtual object and real scene. The blur effect could be used as a depth cue in AR applications. These research results show the feasibility on making an AR system using blur effect as a depth cue. Okumura's researches on how to make a seamless scene in AR application show a good reference on the system design. If the limitation of the position to render the blur effect in Okumura's research could be eliminated, the depth cue method would become practical in real applications. In this thesis, a new method to estimate the blur effect from the scene is proposed. In this method, as long as the blur parameter of a single point in the real world is known, the blur effect of other point in the scene can be estimated. In this way, the system could render the blur effect not only on the virtual object which is composed on the marker but also on the virtual object away from the marker. In previous researches about the human perception, many user tests have been done to figure out how the viewer would percept blur effect as a depth

cue. In this research, user tests are also conducted to solve the problem on how the users would perceive the blurred virtual contents as a depth cue.

Chapter 3

PRE-RESEARCH AND PROPOSED METHOD

3.1 Overview

The chapter focus on discussing the blur effect caused by defocus of a camera lens. Based on the thin lens model of a camera lens, a method is proposed to estimate the blur parameter (i.e., the radius of the imaged point's "blur circle") of any point in the whole screen referring to its spatial information in the real world coordinate. In the proposed method, point spread function (PSF) is introduced to evaluate the blur of a point.

In Chapter 2, the potential of making blur effect as a depth cue is introduced. However, some problems remain. One of the biggest problems is that would the observers simply prefer the clear virtual contents instead of the blurred virtual contents because the clear ones seem to provide more information in sharpness, colour or lighting condition. To figure out the answer of this problem, a pre-research user test was conducted and introduced in the beginning of this chapter.

3.2 Pre-research: the Influence of Blur Effect on Egocentric Distance

In this pre-research, user tests are raised to discuss whether the blur effect would lead the user a better understanding of the egocentric distance to the virtual object or not. In the test, users were given the same content (a picture of human face) in two different ways at the same time. One image was a printed photo and the photo was pasted on a thing board putting in the real scene. The other image was the same photo but a digital content on the computer screen. In the experiment, the real photo was moved from near region to far region and the digital photo

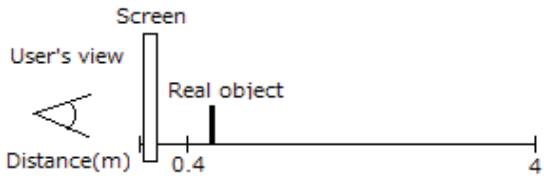


Figure 3.1: Illustration of the pre-research system. The observers were invited to sit facing the AR system. The track for moving the real objects was behind the screen and the camera was set at the back of the screen so that although we used a see-through screen in the experiment, the observers were still able to perceive the egocentric depth in an appropriate way.

Figure 3.2: Stereo view of the experiments for pre-research. In the real scene, the image of face was pasted on a thing board. The thing board was put at a certain distance and was showed in the lower half of the screen. On the upper half of the screen, the image rendered by low pass filter was showed. When the observers click the window, the images would change. There were 22 images could be selected in one trial. The observers were asked on which image appeared in the upper half of the screen seems have the same depth with the real image showed in the lower half of the screen. In this figure, the photo of the face in the real scene was put on the position 80cm from the observer and the the cut-off frequency of the virtual image is 20Hz.

was filtered by low pass filters with different cut-off frequency. The observers were asked to match the images between the ones in the real scene and the ones blurred to different extend. The observers were asked to choose the one of the digital images which they thought could make them feel that it was at the same depth with the real photo. In this pre-research, the user tests focused on near field and medium field AR perception from 0.4m to 4m.

3.2.1 Test Settings

Figure 3.1 shows the illustration of the experiment system. 7 observers were invited to sit in front of the AR system. The track for moving the real objects was behind the screen and the

camera was set at the back of the screen so that although the observers were using a see-through screen in the experiment, they were still able to perceive the egocentric depth in an appropriate way.

In order to eliminate the influence from other depth cues, such as occlusion and three-dimension perception, only a 2D image was used in the experiment. The real object was a thing board with a human face pasted on it. The face is about 12cm high and 8cm wide, and window size of the augmented reality application is 960*720 pixels with $30f/s$ frame rate. The camera used in the research was a Logitech 2MP autofocus web camera with the Carl Zeiss Tessar lens with the maximum aperture of 2.0 and a focal length of 3.7 mm.

In Figure 3.2, the upper half of the window was blocked and virtual object, which is the same 2D image of the real object, was overlaid on the gray background. The setting of the monochrome background was to avoid the influence of linear perception. When the real thin board was moved away in distance, the size of the real objects became smaller. The size of the virtual images was always kept the same with the real photo so that the observers needed only to focus on the perceptual difference the blurring effect brought. When the observers click the window, the images would change in a random order.

In this test, the virtual object was rendered by a second order Butterworth filter[40]. By changing the cut-off frequency, the original image has been rendered into 22 different level of blur effect. In the real scene, experiment track started from 0.4m to 4m and is divided into 26 sections. In each trail, observers were asked to test on the whole 27 points of the distance.

3.2.2 Task

Observers were asked to sit in front of the screen and face experiment system. By clicking the mouse, the 22 blurring images was showed in turn. The observer was asked to choose one image which they think they could accept that the real object and the virtual object seemed to be on the same vertical plane. The virtual objects randomly ordered by their blurring level and the 27 distance points also appeared randomly by manual moving.

3.2.3 Test Results

The result of the pre-research was showed in Figure 3.3. The results show that when the egocentric distance increased, the observers prefer to see more blurred texture on the virtual object. There were two pieces of feedback need to be noticed here. First, although the observers figured out quickly that the difference between the sample virtual images was the blur effect, they paid much time on comparing the difference between the virtual images from times to times in one trial. After comparing the blur of the real photo on the thing board and the virtual image, they chose the one with the certain blur which could match the depth in the real scene. From this process, I made an assumption that the blur effect, in the near region AR, could play a role as a depth interpretation. However, the users are not very sensitive to the degree of blur. The details of their sensitivity should be discussed later in the research. Second, some of observers thought

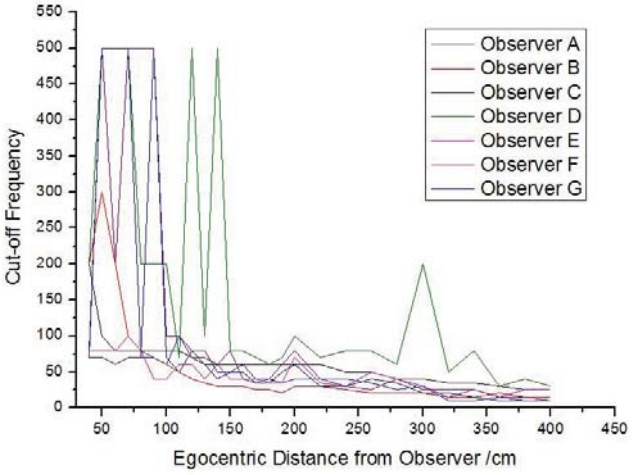


Figure 3.3: Result of the pre-research. The x axis shows the egocentric distance from observer and the y axis shows the cut-off frequency of the low pass filter used to blur the virtual image. 7 observers were interviewed in this test.

that it was difficult to make a choice because the real photo and the virtual image blurred in different ways. At some distance, the real object showed blur at the eyes while a clear contour of the cheek. However, some of the virtual images showed blurred in both the eyes and the cheek. The most plausible explanation for this might be that the second order Butterworth filter was not a proper filter to render the blur effect in augmented reality. In the later part of this thesis, I will continue the discussion on how to render the blur in a proper way.

3.3 Proposed Method Based on Blur Effect in Augmented Reality

Inspired by the autofocus researches mentioned in Chapter 2, a new model to estimate blur effect in augmented reality is proposed. The model is derived from the thin lens model. The blur parameter r should be referred to the definition of point spread function(PSF)[37]. In this section, the point spread function and thin lens model are first introduced. Then, the proposed blur estimation model is discussed.

3.3.1 Point Spread Function

The point spread function (PSF) describes the response of an optical imaging system to a point source or point object. The PSF is the impulse response of a focused optical system. The degree of spreading (blurring) of a point is a measurement for the quality of an imaging system and is quite useful in field such as electron microscopy or lens performance testing. Due to the linearity property of optical imaging system, the image of an object can be computed by

expressing the object-plane field as a weighted sum over point of spread functions. The image of a complex object can be regarded as a convolution of the true object and the PSF as shown in Function (3.1).

$$I'(x,y) = I(x,y) * P(x,y) \tag{3.1}$$

In Function (3.1), $I(x,y)$ is the input function of the ideal image and $If(x,y)$ is the output function of the blurred Image. $P(x,y)$ is the point spread function which causes the blur in the optical system. x and y is the position in the image coordinate. The PSF of blur is generally defined as an impulse function.

$$P(x,y,r) = \begin{cases} \frac{1}{r^2} & ,x^2+y^2 \leq r^2 \\ 0 & ,otherwise \end{cases} \tag{3.2}$$

where x and y are the 2D position of the corresponding point on the image, r is the blur radius.

In Function (3.2), PSF is presented as an ideal impulse function. The r in Function (3.2) could be regarded as the radius of the blur circle or the blur parameter. For any point on the image, the blur circle should be ideally symmetrical in all directions. However, in practice, due to the imperfection of the optical system, the blur circle is not perfectly symmetrical too. The advantage of introducing the point spread function (PSF) is that in the rendering process afterward, the blur of virtual contents could be rendered by simply convoluting with the PSF.

3.3.2 Thin Lens Model

The thin lens camera model[4] is a general used model when discussing problems on zooming and focusing. The model is described below,

$$\frac{1}{u} + \frac{1}{v} = \frac{1}{f} \tag{3.3}$$

where u is the distance between a point in the scene and the lens, v the distance between the lens and the image plane (the plane where the images form on in optical systems), and f the focal length of the lens.

In Chapter 2, Function (2.1) is also derived from the thin lens model. There is one thing should be noticed here. The lens used in the implementation is not a typical thin lens but a compound lens. A compound lens is an array of simple lenses (elements) with a common axis; the use of multiple elements allows more optical aberrations to be corrected than is possible with a single element. The model for a compound lens is very complex. In approximate cases, it is possible to apply the thin lens model[4].

3.3.3 Proposed Blur Estimation Model

In Figure 3.1, when two points in the scene are considered, their blur parameters are indicated as $R1$ and $R2$ on the image plane. If the point is in focus, the R equals to zero. Suppose that D

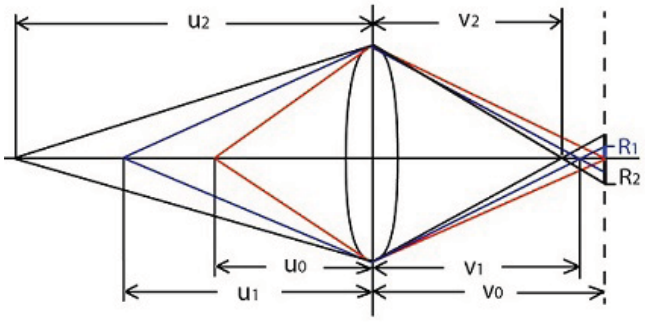


Figure 3.4: Illustration of the thin lens camera model.

is the diameters of the thin lens (aperture diameter) and $v0$, $u0$ are the standard distances when a point is exact focused. According to the principle of similar triangles, it is easy to derive,

$$\frac{2R_1}{D} = \frac{v_0 - v_1}{v_1} \tag{3.4}$$

So,

$$\frac{2R_1 + D}{2R_1 + D} = \frac{v_2}{v_1} = \frac{u_2(u_1 - f)}{u_1(u_2 - f)} \tag{3.5}$$

$$R_2 = [\frac{u_1(u_2 - f)}{u_2(u_1 - f)} \cdot (2R_1 + D) - D]/2 \tag{3.6}$$

$$r_2 = R_2/\sigma = [\frac{u_1(u_2 - f)}{u_2(u_1 - f)} \cdot (2r_1\sigma + D) - D]/2\sigma \tag{3.7}$$

Since f and D are the intrinsic parameter of the lens, u_1 could be measured and u_2 is known as the position of the estimating point. As long as R_1 is known, R_2 could be easily estimated. The unit of R requires special attention. In the previous section, the unit of r is pixel which is easy to understand from the computer vision point of view while the unit of R should be the same as u and v. Thus, a parameter σ is introduced to correct the unit problem. σ could be calculated by the CCD sensor size divided by the image resolution as mm/pixel. After correction, Function (3.7) could be derived as shown.

3.4 Discussion and Conclusions

In this chapter, a pre-research user test is discussed. The user test shows that when the egocentric distance increased, the observers preferred to see more blurred texture on the virtual object in AR systems. The result of the user test confirmed the feasibility of using blur effect as a depth cue in AR applications. In the later half of this chapter, an blur estimation method based on the thin lens model is derived. The new blur estimation model would be used in the prototype implementation in Chapter 4.

Chapter 4

PROTOTYPE SYSTEM DESIGN AND IMPLEMENTATION

4.1 Overview

In Chapter 3, the proposed blur estimation model is proposed. This chapter focuses on how to implement the blur estimation model into the a prototype system. Figure 4.1 shows the flowchart of the prototype system. The processing flow is divided into five steps: 1. detect the marker; 2. register the intrinsic and extrinsic parameters of the camera; 3. measure the defocusing (blur) parameters on the marker; 4. estimate the blur parameters in the whole view; 5. render the virtual objects with blur effect. In the prototype system, a checkerboard calibration process is also induced so that the blur parameter in the scene could be measured. Although in the estimation section, only parts of the measured data would be used, the measured blur parameter in the whole region on the checkerboard would be saved for future evaluation.

4.2 Checkerboard Detection and Registration

In order to detect the blur parameters in a larger region, a checkerboard calibration method is introduced. Checkerboard is widely used in computer vision to calibrate the intrinsic and extrinsic parameters of the camera and fix the image distortion [34]. In this research, apart from the camera calibration and registration, checkerboard is also used to provide adequate linear edge information, which helps to calculate the defocusing parameter. Once parts of the defocus parameters on the checkerboard are measured, the model could be used to estimate the blur parameter out of the checkerboard based on the linear feature of defocusing. It should be noted here that in real application, since only part of the blur parameters need to be measured, it is proper to measure the blur parameter of the marker instead of using a checkerboard.

In order to measure the degree of blur in the real scene, it is important to have a texture in

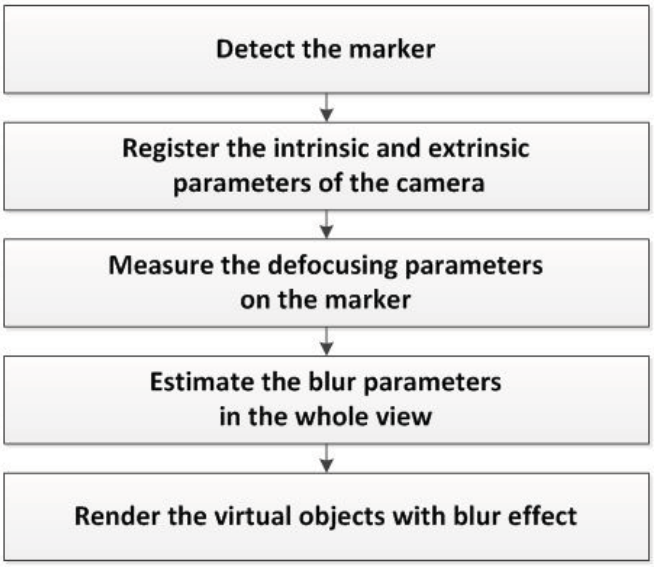


Figure 4.1: Process flow of the prototype system.

the real scene which is regarded as "clear". In general, it is more easy to measure the degree of blur on an edge. Although there are plenty of edges in the real scene, it is difficult to figure out which edge is originally blurred and which is not. The most reliable way is to induce the black and white edges on a checkerboard or a marker as a reference. In this implementation, a checkerboard is used. In the practical application, blur measurement could be conducted by detecting the edges on the AR marker.

The checkerboard used in our implementation is showed in Figure 4.2. It is a 7×10 checkerboard. In the detection part, the openCV library[5] is used to do the checkerboard detection and the checkerboard have to be an even number times an odd number checkerboard. The checkerboard is printed on a A4 paper card. The checker is 25×25 mm^2 in size. The classical black-white checkerboard could be used as a typical calibration tool for camera calibration in OpenCV. Camera calibration involves the estimation of both extrinsic and intrinsic camera parameters.

The first stage of the camera calibration procedure is to establish correspondences between 2D points in the image and 3D points on the checkerboard, so-called point-correspondences. If the checker size of the checkerboard is known, it is easy to calculate the camera pose and relative position by using the released API[13]. In this prototype implementation, the intrinsic parameters of the camera, the focal length f and the aperture diameter D are pre-stored in the system. How to estimate the camera pose in real-time will not be discussed in this thesis.

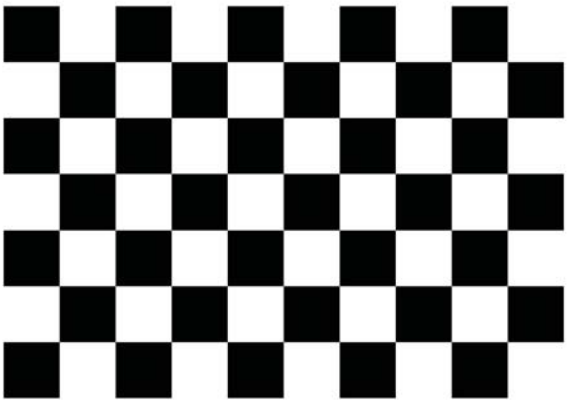


Figure 4.2: The checkerboard pattern.

4.3 Blur Parameter Measurement

After the registration process of the camera, the checkerboard has already been calibrated and the 3D position of the corners and edges could be known. By setting the black-white edge as field of interest (sub-image or block in the whole image [21]), the analysis on certain edge in the particular space could be conducted as shown in Figure 4.3. The PSF is calculated by measuring the intensity of pixels perpendicular to the linear edged. According to Figure 4.4, the intensity of pixels perpendicular to the edge is shown. In Figure 4.5, the gradient of the intensity curve is shown. The curve well describes the spreading (blurring) of the optic system. If the optic system is ideal and there is no blur in the measured area, the gradient of pixel intensity perpendicular to an edge should be an impulse instead of a distribution curve. The position of the peak of the gradient curve is supposed to represent position of the linear edge. However, due to the imperfection of the optic system, the gradient value is asymmetric around the peak position. In the measurement method, the exact position of the edge is not necessary since the radius of the blur could be estimated by measuring the diameter of the spreading (blurring) on the edge.

A threshold of 1 intensity/pixel is set in order to reduce the influence of the noise. The pixels whose intensity gradient is over the threshold are counted into the blur diameter. The discrete stars show the pixels included in the blur diameter. The r could then be calculated by simply dividing the blur diameter by two.

The high resolution of the image background used in the prototype AR provides adequate intensity data on the edge. For each of the field of edge, the intensity of around 200 pixels is measured. In Figure 4.6, a set of data analysis of one section of the edge is shown. A Simple Moving Average (SMA) filter is used to smooth the raw data as shown in Function (4.1). The average of every n pixels is calculated and shown in the red line. The gradient curve shows the

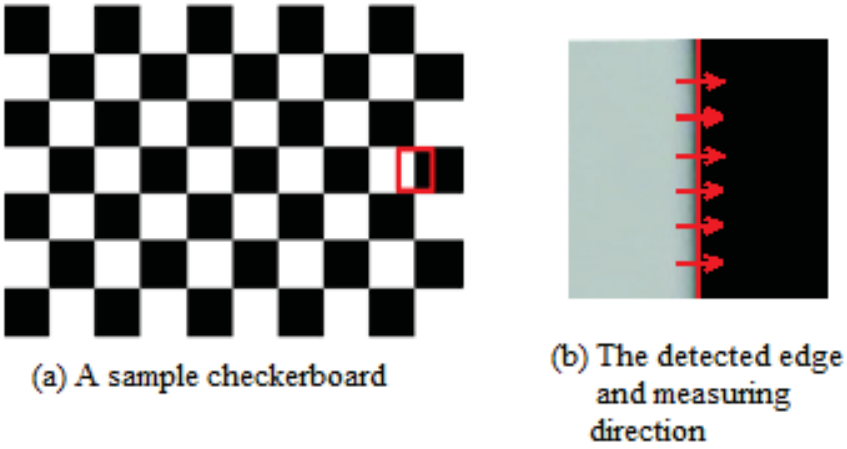


Figure 4.3: The checkerboard and a detected edge.

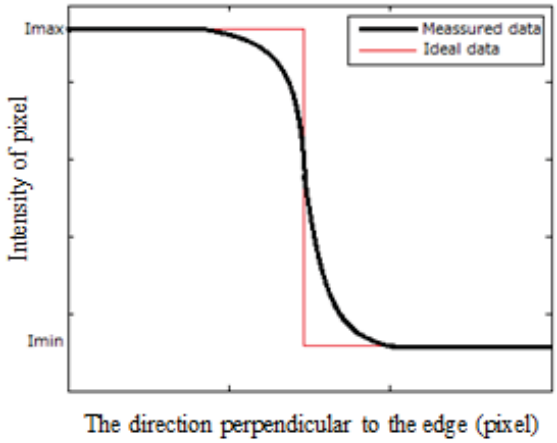


Figure 4.4: Intensity of pixels perpendicular to an edge.

asymmetry resulting from measurement error or a lack of accuracy of the checker.

$$SmoothedData = \frac{I_M + I_{M-1} + \cdots + I_{M-(n-1)}}{n}, \tag{4.1}$$

where I is the intensity of pixels, M the position of the pixel.

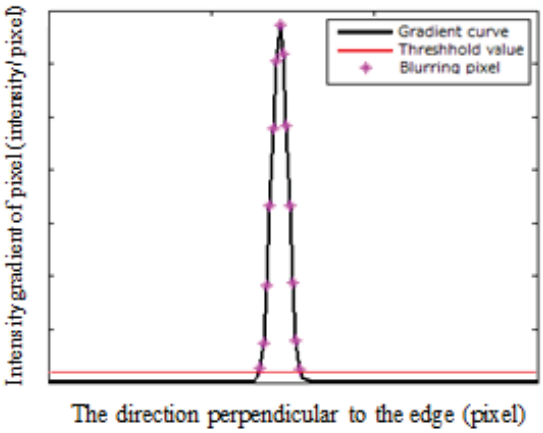


Figure 4.5: Gradient of pixel intensity perpendicular to an edge.

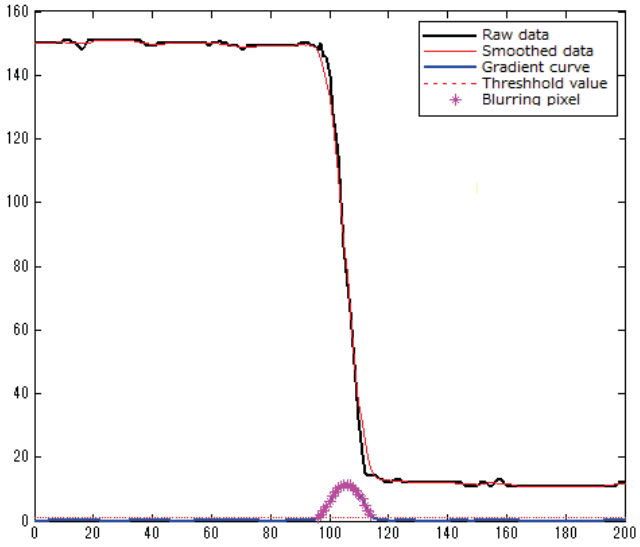


Figure 4.6: A set of sample data of the pixel intensity in the prototype AR system.

4.4 Blur Parameter Estimation

In Chapter 3, the blur parameter estimation method can be summarized as Function (3.7) . Note thatF u_1, u_2 are distance between a point in the scene and the lens; f the focal length; D the aperture diameter; r_1, r_2 the blur radius.

$$r_2 = R_2/\sigma = [\frac{u_1(u_2 - f)}{u_2(u_1 - f)} \cdot (2r_1\sigma + D) - D]/2\sigma \tag{4.2}$$

In the implementation of the estimation method, the nearest parallel edge on the checkerboard is chosen as the reference edge. The blur radius of the edge on the image plane is measured. The u_1 is the distance from the nearest edge (Edge 1) to the image plane. r_1 is the blur radius of the points on Edge 1. Using the blur parameter measurement method introduced above, the r_1 could be measured. The focus length f and the aperture diameter D was stored in advance in the registration process. In the implementation of the prototype system, the camera view is fixed which makes it much easier to register the camera parameters. In the practical application, the focal lens and aperture registration of a video stream might be more complicated. For the camera whose focus and aperture are auto adjusted, the registration method might should be adjusted in some way.

In order to know r_2, I also need to know u_2. u_2 is the distance from the estimating point to the image plane. After the checkerboard calibration, a perspective of the real scene is built. Since the size of the checkerboard and the marker in the real scene is known, the relation between the position in the real scene and the position in the virtual 3D coordinate could be built. In this way, r_2 could be estimated.

4.5 Virtual Object Rendering in the Scene

In the implementation, ARToolkit[18] is used to build up the graphic coordinate. All the computer graphics in this implementation are drawn using OpenGL[38][29]. The blur effect is rendered by filtering the high frequency information in the image. A shader is often used to change the information in the image. In the implementation, a blur shader was written instead of using a packaged shader. The advantage of writing a new shader is that the detailed parameters such as the blur radius, the color of pixels and the point spread function can be adjusted freely. On the other hand, the blur shader is a high cost calculation in rendering. By writing a shader, the blur rendering could be accelerated by providing a low cost algorithm. In this section, an instruction of GLSL (OpenGL Shading Language) is given first and then how to render the blur effect in the prototype system is discussed.

4.5.1 Introduction of GLSL

There are several languages that can be used to write shaders, such as Cg, HLSL and GLSL. OpenGL Shading Language (GLSL)[19], is a high-level shading language based on the syntax of the C programming language. It was created by the OpenGL ARB ithe OpenGL Architecture Review Board is an industry consortium that governed the OpenGL specification) to give developers more direct control of the graphics pipeline without having to use ARB assembly language or hardware-specific languages. GLSL code is similar to C code, but with a strong emphasis on computation.

There are two types of graphics pipelines, the programmable pipeline and the fixed function pipeline. The "fixed function" pipeline refers to the older generation pipeline used in GPUs

(graphics processing unit) that was not really controllable – the exact method in which the geometry was transformed, and how fragments (pixels) acquired depth and color values were built-in to the hardware and couldn't be changed by the developers. These fixed methods allowed the programmer to display many basic lighting models and effects, like light mapping, reflections, and shadows (always on a per-vertex basis) using multi-texturing and multiple passes. This was done by essentially multiplying the number of vertices sent to the graphic card. With the programmable function pipeline, the limits of the fixed function pipeline were removed. All fixed per-vertex and per-fragment computations could be replaced by custom computations, allowing developers to do vertex displacement mapping, morphing, particle systems, and such all within the vertex stage. Per-pixel lighting, toon shading, parallax mapping, bump mapping, custom texture filtering, color kernel applications, and the like could now be controlled at the pixel stage. Figure 4.7 shows the programmable pipeline function scheme. The OpenGL Shading Language is actually two closely related languages. The specific languages will be referred to by the name of the processor they target: vertex or fragment. The vertex processor is a programmable unit that operates on incoming vertex values and their associated data.

Programs written in the OpenGL Shading Language that are intended to run on the processor are called vertex shaders. Vertex shaders can be used to specify a completely general sequence of operations to be applied to each vertex and its associated data. Programs written in the OpenGL Shading Language that are intended to run on this processor are called fragment shaders. Fragment shaders can be used to specify a completely general sequence of operations to be applied to each fragment. Fragment shaders that perform some of the computations from the list above must perform all desired functionality from the list above[20].

4.5.2 Blur Shader Implementation

The shader is based on Gaussian Function. First, the virtual contents are drawn into a frame buffer. Then, the whole buffer is shaded with the blur shader. The shader parameter is based on the blur radius estimated by the blur estimation method. The shader algorithm is discussed below. The vertex shader implemented in the prototype is shown in Appendix-A.

4.5.2.1 Gaussian Function As Blur Model

Gaussian blur is a widely used effect in graphics software, typically to reduce image noise and reduce detail. Pentland's research[34] indicates that the sum of the various function obtained at different wavelengths has the general shape of a two-dimensional Gaussian, as shown in Function (4.3). The graph of the Gaussian function is a characteristic symmetric "bell curve". Gaussian blur is done by convolving each point in the input array with a Gaussian kernel and summing them all to produce the blurred image. The two dimensional gaussian function is showed in Function 4.8. This Gaussian distribution function could be used to simulate the point spread function mentioned in the previous chapters.

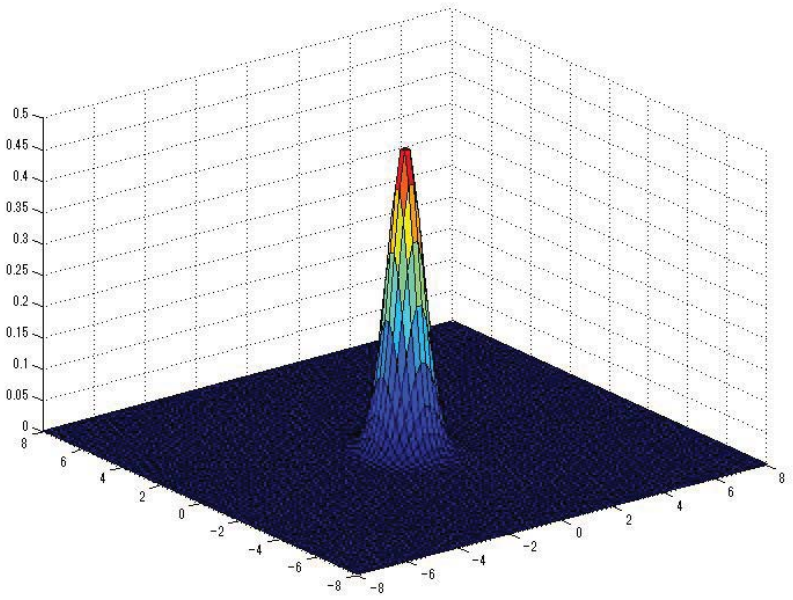


Figure 4.7: Graph of the two dimensional Gaussian distribution. The graph of the Gaussian function is a characteristic symmetric "bell curve". The Gaussian distribution could used to approximate to the point spread function.

$$G(x,y) = \frac{1}{2\pi\sigma} exp\Big\{ - \frac{(x^2+y^2)}{2\sigma_2} \Big\}, \tag{4.3}$$

where x is the distance from the origin in the horizontal axis, y is the distance from the origin in the vertical axis, and σ is the standard deviation of the Gaussian distribution.

4.5.2.2 Blur Shader Implementation in GLSL

In implementation, the Gaussian blur shader is divided to process into two passes based on its separable property. In the first pass, a one-dimensional kernel is used to blur the image in only the horizontal or vertical direction. In the second pass, another one-dimensional kernel is used to blur in the remaining direction. The resulting effect is the same as convolving with a two-dimensional kernel in a single pass, but requires fewer calculations. When converting the Gaussianfs continuous values into the discrete values needed for a kernel, the sum of the values will be different from 1. This will cause a darkening or brightening of the image. To remedy this, the values can be normalized by dividing each term in the kernel by the sum of all terms in the kernel. The Gaussian function was changed into: (The code of this implementation in GLSL

could be referred to Appendix-A.)

$$G_x(x,y;r) = \begin{cases} \frac{1}{M_x} exp(-\frac{x^2}{2\sigma^2}) & ,if y = 0, x \leq r \\ 0, & otherwise \end{cases} \tag{4.4}$$

$$G_x(x,y;r) = \begin{cases} \frac{1}{M_x} exp(-\frac{x^2}{2\sigma^2}) & ,if x = 0, x \leq r \\ 0, & otherwise \end{cases} \tag{4.5}$$

where,

$$M_x = M_y = \sum_{i=-r}^{r} exp(-\frac{i^2}{2\sigma^2}). \tag{4.6}$$

4.6 Prototype System Implementation

In order to verify the feasibility of the algorithm mentioned in the previous section and the practicality of the proposed method in augmented reality, a prototype AR system based on the proposed processing flow is constructed as showed in Figure 4.8. In this prototype system, a high resolution digital camera is used instead of a general web camera and the augmented scene is composed of static pictures taken by the camera other than a video stream. The high resolution picture helps to achieve clearer evaluation of the algorithms proposed and the easy manipulation of the aperture and focus of a digital camera helps to make the experiment more accurate. In Table 4.1, the specs of the prototype system are shown.

In Figure 4.9, we can see the three main elements in the system: 1. a digital camera; 2. an AR marker; 3. the 7×10 checkerboard. The image plane and lens axis are also marked in Figure 4.9. All the distances mentioned in last chapter, for example $v0$, $v1$, $v2$, $u0$, $u1$, $u2$ have to be perpendicular to the lens axis and parallel to the image plane. In the implementation, the tilting angle of the camera, the vertical distance and the horizontal distance are necessary for the calculation of $v0$, $v1$, $v2$, $u0$, $u1$, $u2$.

Figure 4.10 is an image of the camera view. This image shows a stronger blur effect in the farther region than the normal video in the general AR applications. This is because the focus point was focused on the middle of the first edge of the checkerboard(the nearest parallel edge to us in the scene). And a quite small depth of focus is used so that the blurring effect caused by defocusing seems stronger than the normal situation.

4.7 Evaluation of the System Implementation

The key part in the proposed process is the blur radius estimation method of the whole camera view. On supposing the blur radius measurement method is reasonable, an experiment is conducted to evaluate the blur radius estimation method.

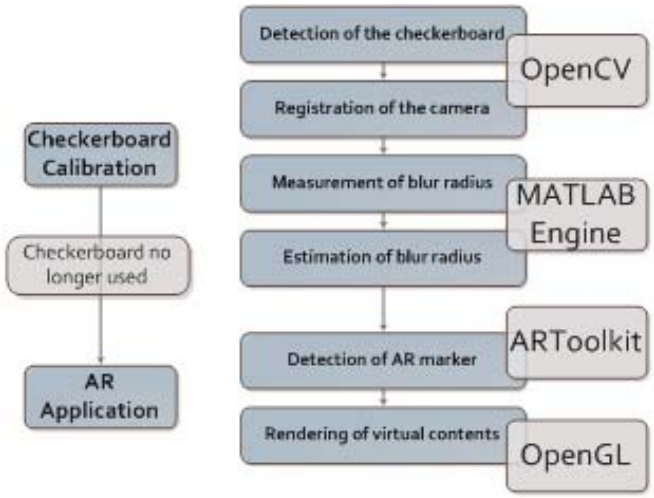


Figure 4.8: Implemented process flow chart.

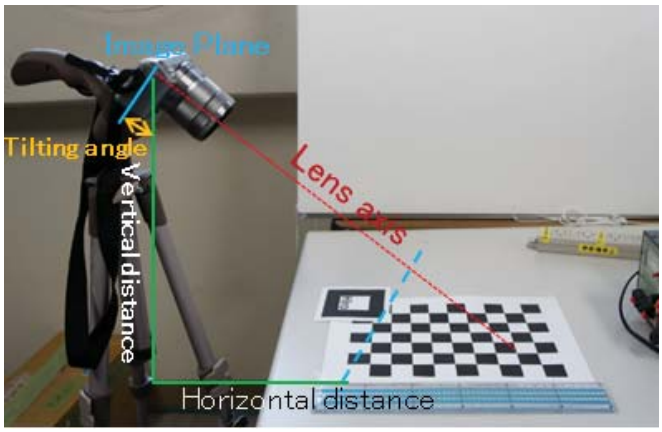


Figure 4.9: Implemented prototype AR system.

Table 4.1: Prototype AR system specs.

Computer	HP Pavilion dv4 Processor: Intel Core i5 2.40Hz Installed memory: 4.00 GB
Graphic Card	AMD Radeon HD 6750M 1GB
Camera	SONY NEX-5N Image Dimensions: $4912 \times 3264 pixels^2$ Size of CCD sensor: $23.4 \times 15.6 mm^2$
Lens	SONY Optical SteadyShot D/f : $3.5 - 5.6/18 - 55mm$
Checkerboard	Number of checkers: 7 10 Size of checkers: $25 \times 25 mm^2$

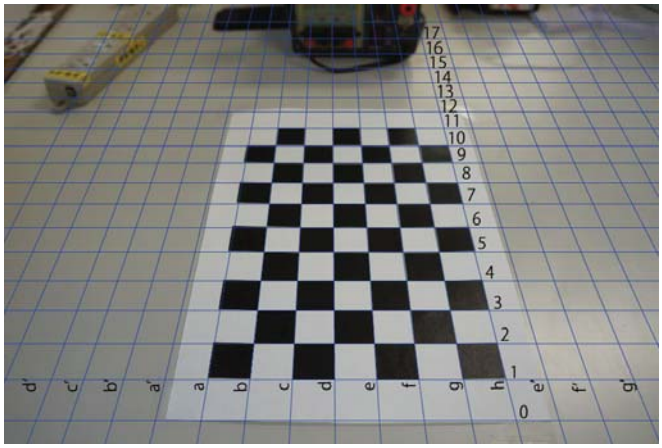


Figure 4.10: Camera view in the prototype AR system. The scene is divided into checker-sized sections. Each row and column is labeled with a number or an alphabet.

Table 4.2: Calibration Parameters for the experiment.

Aperture diameter (D)	$f/5.6mm$
Focus Length (f)	$21mm$
Image resolution	$4,592 \times 3,056 pixels^2$
	$5.1 \times 10 \exp(-3) mm/pixel$
Tilting Angle)	38
Vertical distance	$0mm$
Horizontal distance	$17mm$

4.7.1 Experiment Setup

The 7×10 checkerboard contains 11 linear edges which represent 11 discrete light planes in the imaging system parallel to the image plane (Edge 1 to Edge 11 in Figure 4.10). From one shot, the blur parameter on the 11 edges could be measured respectively. Based on the blur parameter of one of the edges, the blur parameter of the other edges could be estimated using Function (3.7). By comparing the measured blur parameter and the estimated blur parameter inside the area of checkerboard, an evaluation of the estimation algorithm could be conducted.

According to the prototype system in Figure 4.9, the focus is manually set to the nearest edge in the scene (Edge 1 in Figure 4.10). The blur parameter of this edge is brought into Function (3.7) as r_1. The necessary parameters for calculation are listed in Table 4.2 and Table 4.3. The tilting angle is the angle between the image plane and the vertical plane. The blur on any point in the camera scene (if the position is known) could be calculated now.

4.7.2 Results and Analysis

In Figure 4.11, the estimated blur parameter r and the measured blur parameter r is compared. The measured blur parameter is the average value of the 7 sections on an edge (each edge

Table 4.3: Distance from measured edges to the image plane.

Edge	1	2	3	4	5	6
u/mm	34.23	35.96	37.70	39.43	41.17	42.90
Edge	7	8	9	10	11	/
u/mm	44.63	46.37	48.10	49.83	52.73	/

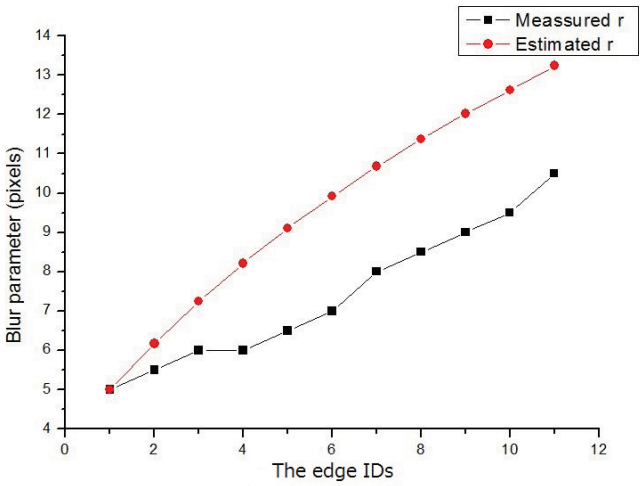


Figure 4.11: A set of comparison of the measured blur parameter and the estimated blur parameter

from Edge 1 to 11 is divided to 7 sections by Edge a to h). Although the estimated r remains larger than the measured r, the measured data shows an ascending trend that is similar to the estimated data. The imperfection of the imaging system is most plausible reason to explain the difference of the data. In Figure 4.12, the 3D graph shows the raw blur parameter data measured in the area of checkerboard. Figure 4.13 shows the blur parameter estimated on the space of checkerboard. The x-y plane of the 3D charts indicates the position of the black-white edge sections. With only the blur parameter of the first edge, the blur circle radius of the whole scene could be estimated by extending the value of u_2 to the red-colored region in Figure 4.14.

With the increasing distance to the focus plane, the difference of the value increases. Comparing to the calculated method, the estimated method is 25% over on the farthest edge (Edge 11) of the checkerboard, which is nearly 4-pixel difference in blur radius. The imperfection of the imaging system is the most plausible reason to explain this difference. Although 25% is pretty big as the error rate, we wondered whether the 4-pixel difference in blur radius would affect the user perception of blur effect for a high resolution image. To figure out the answer, we conducted three user tests shown in next section.

In Figure 4.15, there is an abnormal data on Edge 2 between the Edge b and Edge c. This data is caused by some dark spots on the checkerboard image which is circled in red. The effect of the dark spots is clearly seen on both the raw data and the gradient curve. Since there is

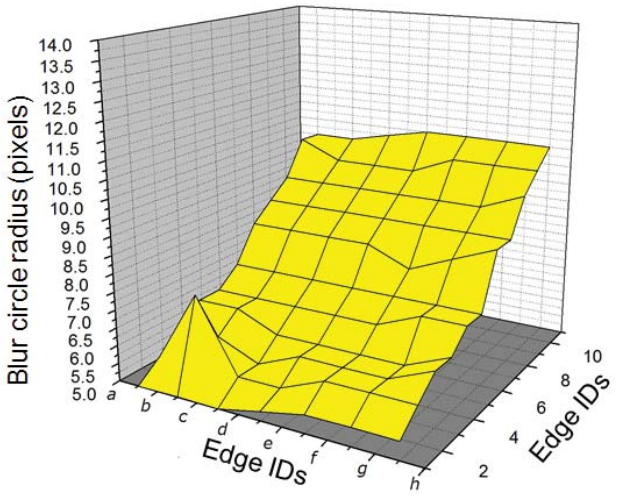


Figure 4.12: Raw data measured on the space of checkerboard.

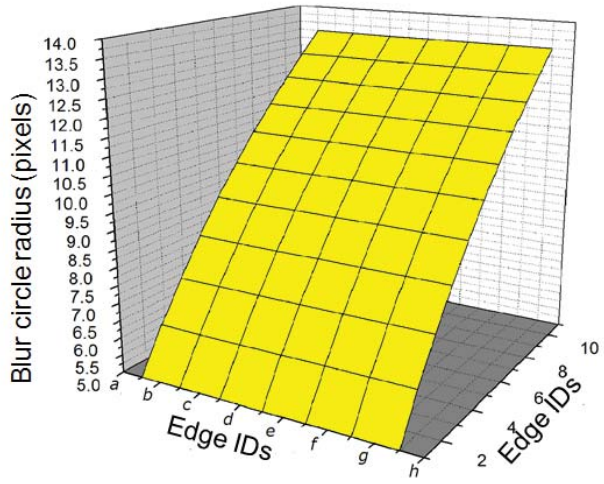


Figure 4.13: Blur parameter estimated on the space of checkerboard.

no algorithm to eliminate the abnormal data, the dark spots led to a bigger PSF radius. It is most likely that the dark spots are printing flaws of the checkerboard. Two possible solutions to this problem can be: 1) to improve the algorithm by incorporating abnormality-handling functionalities; 2) to use a more accurate checkerboard in experiment. In this thesis, a more accurate checkerboard is used in the later implementations.

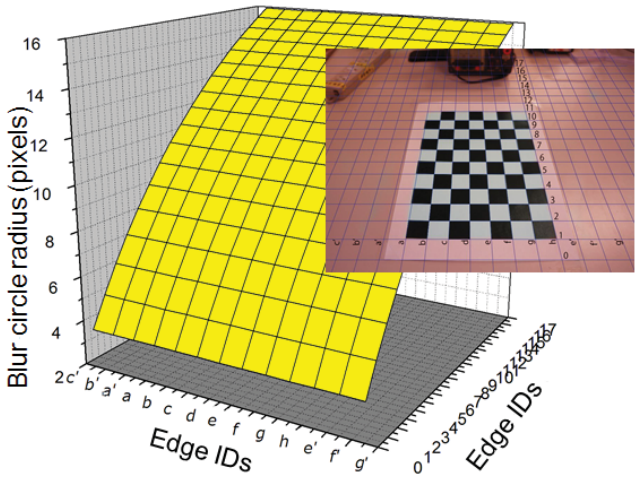


Figure 4.14: The relation of the estimated blur parameter and the spatial information.

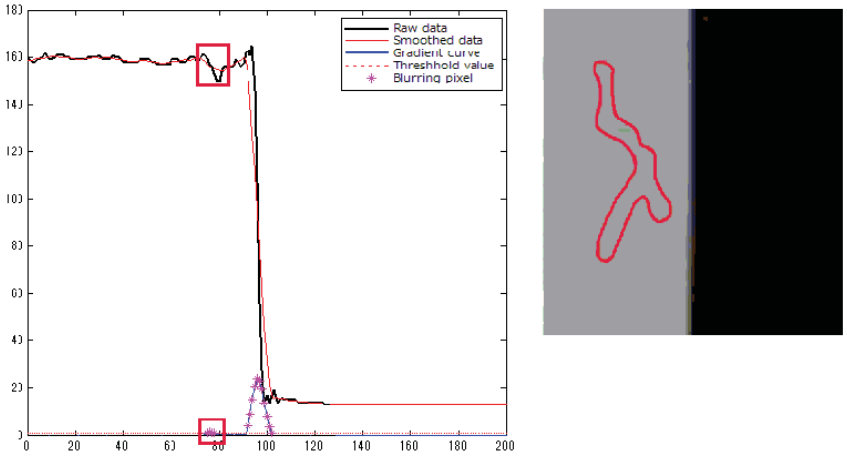


Figure 4.15: Analysis of the abnormal data in estimation. The abnormal parts are marked in red in the intensity chart (the left image) and the edge image (the right image).

4.8 Discussion and Conclusions

In this chapter, the prototype system design and implementation is introduced. The prototype system implemented is based on the blur estimation model proposed in Chapter 3. The implementation of the prototype system attempts to give an answer to the question: how to render the certain degree of blur onto the virtual objects. The evaluation of the prototype system shows some problem in accuracy. The imperfection of the optic system makes the ideal blur estimation model different from the practice one. Whether the difference between the estimated blur radius and the measured blur radius would affect the users' perception and how the users

would perceive the blurred virtual contents as a depth cue would be discussed in Chapter 5.

Chapter 5

USER STUDY

5.1 Overview

In the previous chapter, a prototype system based on the blur estimation model is implemented. In this chapter, several user tests are conducted using the prototype system in order to solve the three questions listed below.

1. Would the user prefer a clearer virtual contents or a seamless virtual-real scene?
2. Will the difference between the blur estimation method and the blur measurement method affect user perception?
3. Is the rendering method able to make the virtual object blur in a way natural to user perception?

The tests would be conducted in various conditions especially when there are moving virtual objects. The test results are discussed after each test.

5.2 User Test 1: Static 2D Virtual Contents

5.2.1 Test Settings

The augmented reality would is made up of a real scene image as the background and a 2D virtual content. The real scene composes a 7×10 checkerboard, an AR marker and some randomly placed objects such as an oscilloscope meter and wiring boards. The AR marker is used to coordinate the perspective using ARToolkit. The checkerboard and other objects are used to help the user understand the depth in the real scene (see Figure 5.1). As discussed in the previous chapter, there are many factors which might affect the depth interpretation of the virtual content. Therefore, in this test, a 2D virtual content is used so that the lighting, shading

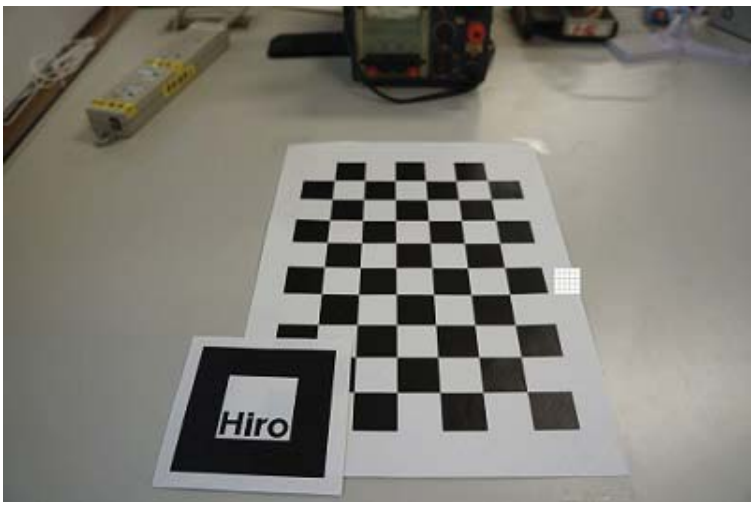


Figure 5.1: Virtual contents pattern for user test 1. A group of two images with virtual contents rendered by different methods were given to the observers. The virtual contents were placed outside the checkerboard aligning one of the edges. The observers were asked to choose the more natural one out of the two.

or other 3D rendering factors will hardly affect the users perception. The 2D virtual content is a 5×5 grid board. The sharpness of the grid pattern is easy for the observers to observe.

The virtual contents were placed outside the checkerboard aligning one of the edges. Since there were 11 parallel edges on the checker, 11 groups of rendered images from edge 1 to edge 11 (edge IDs could be referred to Figure 4.10) as the reference groups were set. In each group, there are two images. The virtual contents were set on the same position but rendered using the measured method and the estimated method separately.

5.2.2 Task

The observers were required to compare the two images and choose which one of the virtual contents seems more natural and fit into the scene. The observers could also indicate that they cannot tell the difference of the two virtual contents. The observers were not informed of the difference between the two images. The number of the observations were collected for each test pattern.

5.2.3 Observers

17 observers between 20 and 30 years old were selected for the test. 85% of the observers were male. The observers all came from unrelated background other than computer vision and computer graphics and had little knowledge in image processing or rendering. Visual impairments such as myopia or astigmatism were not considered since the observers were wearing their corrective devices (glasses or lenses).

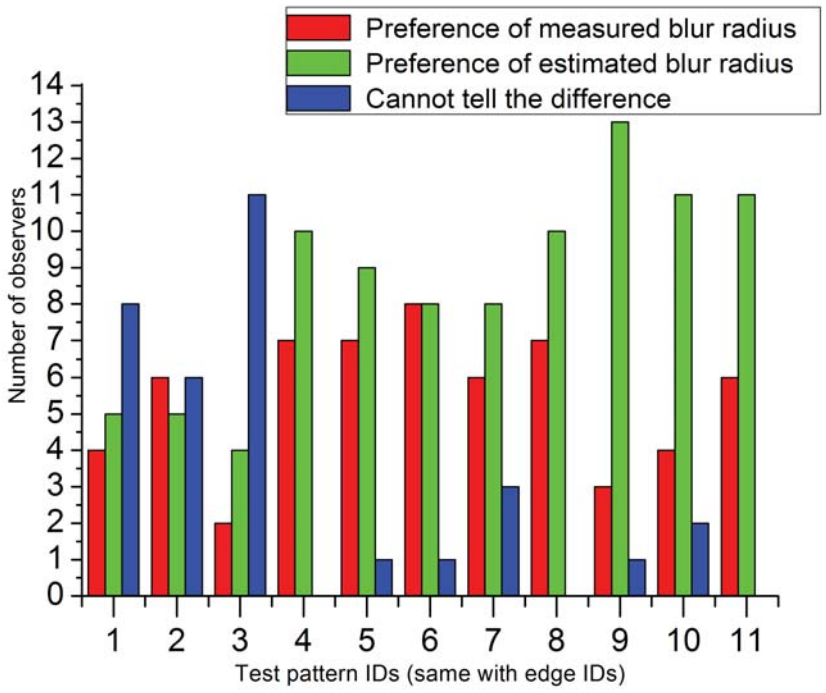


Figure 5.2: Test result of user test 1: with the increasing distance to the focus plane, the population who found the difference between the measured method and the estimated method increased. In the father region (from Edge 9 to Edge 11), the observers were more likely to feel that the virtual contents rendered with the estimated blur radius fit into the background better.

5.2.4 Test Results

In Figure 5.2, the result of user test 1 is shown. The graph showed that with the increasing distance to the focus plane, the population who found the difference between the measured method and the estimated method increased. In the father region (from Edge 9 to Edge 11), the observers were more likely to feel that the virtual contents rendered with the estimated blur radius fit into the background better. During the experiment process, some observers used such kind of words, like "farther in depth" or "consistent" to describe their perception and explain why they made the choice.

5.2.5 Discussion

In Figure 5.2, the result of observers' perception can be divided into 3 parts. Part 1: from Edge 1 to Edge 3. In this part, the observers are more likely to choose the "Cannot tell the difference". This is because the difference of the blur radius was quite small and the observers could hardly figure it out. In the same time, the background of the image still keeps quite sharp. The observers seemed to be confused by the motivation of the user test since they found little

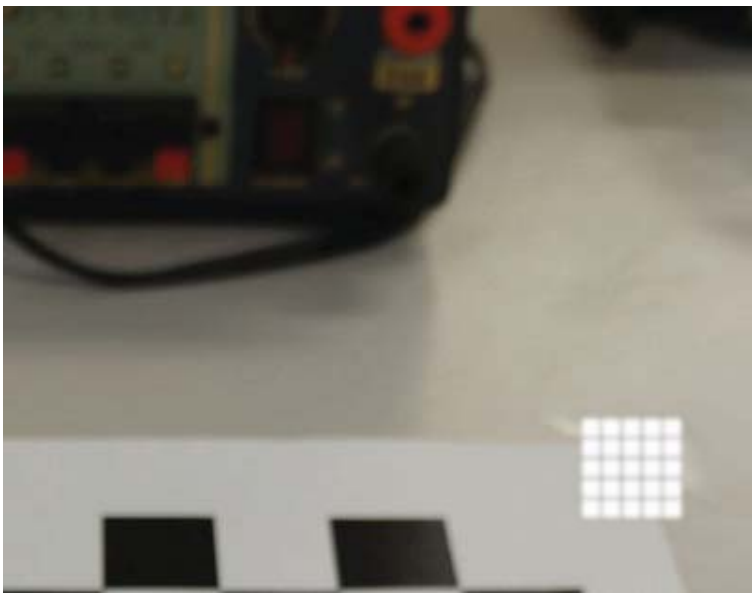


Figure 5.3: Test pattern in the farthest region. The virtual contents in this image is rendered by estimated blur. During the test, many observers reported that they like to compare the virtual contents with the oscilloscope which was in the farthest place in the scene instead of comparing the pattern to the checkerboard around it.

difference in a reference group. Part 2: from Edge 4 to Edge 7. In this part, the population chose "no difference" decreased. Most of the observers claimed that they begin to understand that the virtual contents are gradually blurred but they could still hard to find the difference between the two rendering methods. The number of the observers who choose the "estimated blur radius" were almost the same with the the number who choose the "calculated blur radius". The observers chose either side claimed that the other side was either too blur or too clear. Part 3: from Edge 8 to Edge 11. The observers who choose the "Estimated blur radius" increased. Most of the observers claimed that they feel the more blurred ones seem better consistent to the background.

In Part 1 and Part 2, most of the observers reported that they compared the virtual pattern with the checkerboard to figure out which virtual pattern was more consistent or fit into the scene. However, in Part 3, many observers reported that they were more likely to compare the virtual contents with the oscilloscope which was in the farthest place in the scene (as shown in Figure 5.3). That might be the reason why many of the observers would like to choose the estimation method (a little more blurred) instead of the measurement method in the farther region.

The observers also found that the brightness of the virtual texture (the contrast and relative brightness of the virtual texture changed after blurring) affected their perception to some extent. It has to be noticed that the contrast of the virtual contents will decrease while the blur radius increases. Since the test virtual pattern was a black and white pattern, the change of the contrast

Table 5.1: Experiment patterns used in user test 2. The left numbers refer to the pattern IDs.

	Speed(cm/s)	Rendering method 1	Rendering method 2
1	3	Measured	No- blur
2	1.8	Measured	No- blur
3	3	Estimated	No- blur
4	1.8	Estimated	No- blur
5	3	Measured	Estimated
6	1.8	Measured	Estimated

seemed to affect observers' perception more obviously. In the later user tests, the contrast of the virtual pattern should be adjusted so that the affect of decreased the brightness will not affect the test results much.

It should be noticed that, since the white grid board is not a nature pattern in the image, it was a little hard to convince the users that the board is consistent to the scene. In the later user tests, some more natural virtual contents would be used.

5.3 User Test 2: Moving Virtual Objects

This test focuses on how the observers might perceive the blur effect rendered to a moving virtual object in AR system. Two tasks were conducted using the same test settings.

5.3.1 Test Settings

In user test 2, two moving 3D ball were placed in real image background. The radius of the virtual balls is 10mm and the size of the ball is a little bit smaller than the checker (25mm in length) on the checkerboard. The texture of the balls is a soccer pattern. Two balls lining side by side were moving from Edge 1 to Edge 11 on the checkerboard. As shown in Table 1, the experiments were divided into 6 patterns. Besides the different rendering methods, the speed of the moving object was also taken into consideration. Two different speeds were set for the balls: 1.8 cm/s and 3 cm/s (nearly two times of the previous one, the speed is calculated based on the frame rate). The balls seemed rolling quite fast when the speed was set to 3 cm/s and rolling much slower when the speed is set to 1.8 cm/s. It was supposed that the observers might be easier to compare the virtual contents with the checkerboard in the background while the fast rolling ball could make the observers gave an answer more intrinsically. To eliminate the bias in the experiment, the positions of the virtual balls, the content of the texture pattern and the contrast and relative brightness of the texture pattern were considered.

5.3.1.1 Prejudice of the positions

In the previous user tests, some observers show a prejudice on the positions of the balls. For example, some observers thought that the left balls were always more blurred than the right

balls. In other to eliminate the possible prejudice on the position, the test pattern is doubled by inverting the left-right positions of the balls. In this way, 20 answers (20 observations as shown in the graphs) for each of the six patterns were collected.

5.3.1.2 Texture pattern of the virtual objects

Two patterns were chosen as the texture for the virtual ball. One is a wood texture pattern as shown in Figure 5.4. The other is the soccer pattern (after color balanced). Patterns other than the two might also be reasonable. In this study, patterns were chosen from the available high quality patterns in the laboratory. 10 observers were asked to conduct a pre-test to see whether the texture of the virtual object were set reasonably. 5 observers were given the "wooden" virtual ball and 5 observers were given the "soccer" ball. The test process was similar to test 1. Besides the general questions, the observers were asked to answer one more question. The question was what factor on the rendering of the virtual ball affects their choice of "which ball is more natural" most. Three factors which might affect the observers choice were listed up and the observers could offer their own answer too. The first factor is the blur of the texture pattern. The second is the contrast of the texture pattern. The third is the blur of the edge. These three factors were all the main factors which might affect the observers' perception as introduced in the previous sections. In test 2, it would be ideal that the observers perceive the blur of the texture pattern as the main factor affect their choice in the test.

When the observers were tested with the ball textured with the wood texture pattern, four out of five claimed that the blur of the edge of the ball was the main reason that made them think whether the virtual ball was "natural". When the observers were tested with the ball textured with the soccer pattern, four out of five claimed that the blur of the texture pattern was the main reason that affect their choice on which ball was more nature. The observers claimed that it was hard to tell whether the wood texture pattern was clear or blurred since there was no standard for a clear wood pattern. They also claimed that comparing to the wood pattern, the soccer ball was easy to tell when it was clear or when it was blurred. On the other hand, the observers thought that the "wooden" ball was more convincing (easy to believe that it was a real object in the scene) than the "soccer" ball since the size of the "soccer" ball is somehow unreasonable. In order to ruling out the other perceptual factors, the soccer pattern was chosen as the texture pattern of the moving virtual ball in test 2.

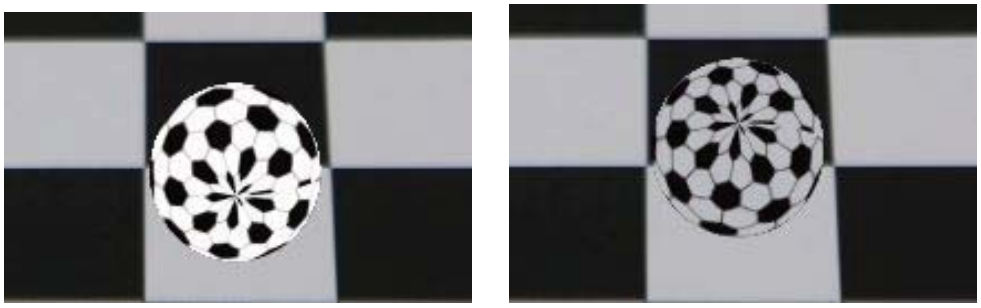
5.3.1.3 Contrast and relative brightness

In order to eliminate the influence from the inconsistency of the contrast and relative brightness of the virtual object, the contrast and relative brightness should be adjusted. The relative brightness of the virtual ball was adjusted by the blur shader. By changing the variable, *coefficientSum* in the blur shader (see Appendix-A), the relative brightness of the virtual ball was adjusted.

The contrast of the virtual texture pattern should also be adjusted, especially the "white

(a) Wood texture pattern (b) Virtual ball rendered with wood pattern

Figure 5.4: Moving ball textured with wood texture pattern.



(a) Before contrast adjustment (b) After contrast adjustment

Figure 5.5: Moving ball textured with soccer texture pattern.

balance". Figure 5.4 shows the virtual ball before and after contrast adjustment. The test environment was set an uniform distribution of light, so the contrast and relative brightness affected by lighting could be ignored.

5.3.2 Task 1

The observers were required to compare the two balls and choose which one of the virtual balls seemed more natural and fit into the scene. The observers could also indicate that they cannot tell the difference of the two virtual balls. The observers were not informed of the difference between the two balls. The number of the observations were collected for each test pattern.

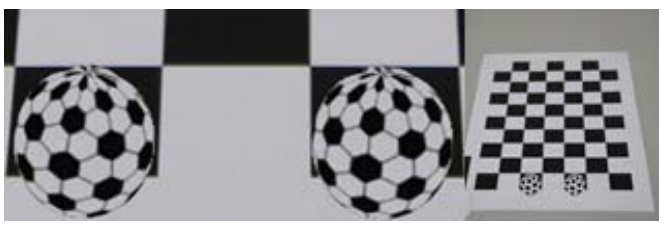


Figure 5.6: Screen shot of moving virtual balls in near region.

Figure 5.7: Screen shot of moving virtual balls in middle region.

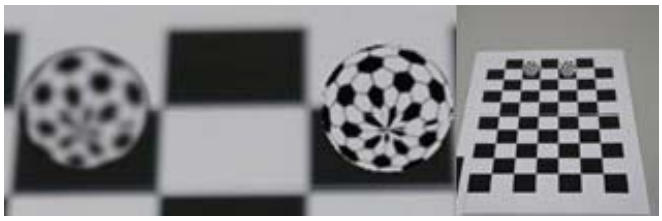


Figure 5.8: Screen shot of moving virtual balls in far region.

5.3.2.1 Observers

10 observers between 20 and 30 years old were selected for the test. 85% of the observers were male. The observers all came from unrelated background other than computer vision and computer graphics and had little knowledge in image processing or rendering. Visual impairments such as myopia or astigmatism were not considered since the observers were wearing their corrective devices (glasses or lenses). 20 observations were collected for each test pattern.

5.3.2.2 Test results

Figure 5.8 showed the result of user test 2. The results of the same group of rendering methods with different speed did not vary a lot (see the odd pattern ID results and the even pattern ID results). In the first four patterns, the observers claimed that they could easily tell which of the two ball was blurred even the ball was rolling in a fast speed. From the feedback, the observers preferred the blurred balls because the no-blur ball made them fill that the ball was "floating" on the screen. In the later two patterns, the observers were hardly able to tell the difference between the balls rendered by the measured blur method and the estimated blur method.

5.3.3 Task 2

The observers were asked to compare the details in the test pattern 2, 4, 6 listed in Table 1. The moving process in task 1 were separated to 10 moving sections: Section 1: virtual ball moves from Edge 1 to Edge 2; Section 2: virtual ball moves from Edge 2 to Edge 3 cSection 10: virtual ball moves from Edge 10 to Edge 11. When the balls reached one of the edge, the moving process was stopped and the balls were wiped off. The observers were required to focus on the moving process and do not compare the texture on the ball with the background checkerboard

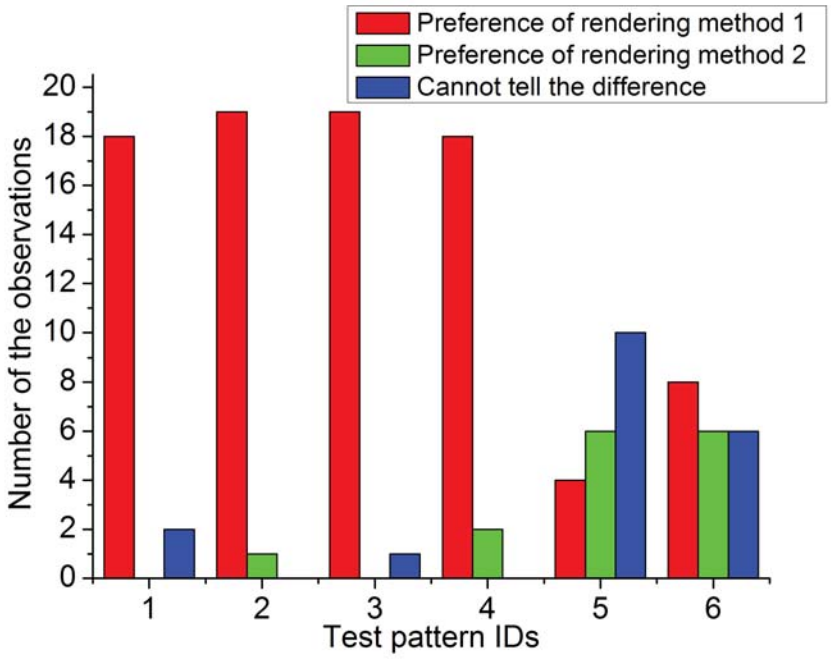


Figure 5.9: Test result of task 1. The pattern IDs and its contents could be referred in Table 1.

when the ball is stopped. The observers were asked which one of the virtual contents seems more natural and fit into the scene. The observers could also indicate that they cannot tell the difference of the two virtual contents. Three groups of tests were conducted for test pattern 2 (the measured blur radius and no-blur effect), 4 (the estimated blur radius and no-blur effect), and 6 (the measured blur radius and the estimated blur radius).

5.3.3.1 Observers

12 observers between 20 and 30 years old were selected for the test. 85% of the observers were male. The observers all came from unrelated background other than computer vision and computer graphics and had little knowledge in image processing or rendering. Visual impairments such as myopia or astigmatism were not considered since the observers were wearing their corrective devices (glasses or lenses). 12 observations were collected for each test pattern.

5.3.3.2 Test results

Figure 5.10 and Figure 5.11 show that in the nearer region in the scene, such as the area from Section 1 to Section 3, observers could hardly tell the difference between the blurred ball and the no-blur ball (see the blue columns in the graphs). This might because the blur parameter still kept small at this region. In farther region, most of the observers preferred the blurred balls

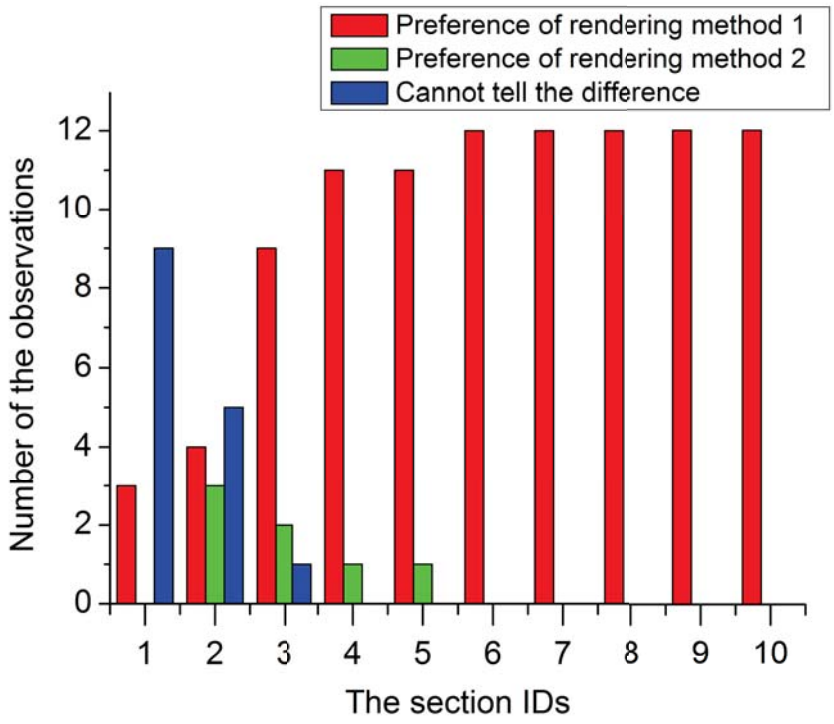


Figure 5.10: Test result of test pattern 2: the measured blur radius and no-blur effect. The observers compared balls rendered by the measured blur radius and no-blur effect. (Section 1 means moving process from Edge 1 to Edge 2cSection 10 means moving process from Edge 10 to Edge 11).

since they thought that the blurred ball seems to fit into the scene better. There were some of the observers chose the no-blur balls before Edge 6 (see the green columns in the graphs). These observers indicated that although they felt the blur could make the scene more consistent, they thought that the blur at the nearer part might be over and they expected to see clearer virtual objects.

In Figure 5.10, in the near region of the scene, the blue columns are high because the measured method and the estimated method were in nearly the same value. From edge 4, the number of the observers who chose the ball rendered by estimated method increased. From the feedback of the observers, some of them claimed that they chose the more blurred ball because they expected that it should be more blurred in the farther region.

5.3.4 Discussion

Three points could be concluded from this test. First, the moving speed would not affect the observers' perception. Comparing to the user test 1, it could be confirmed that no matter

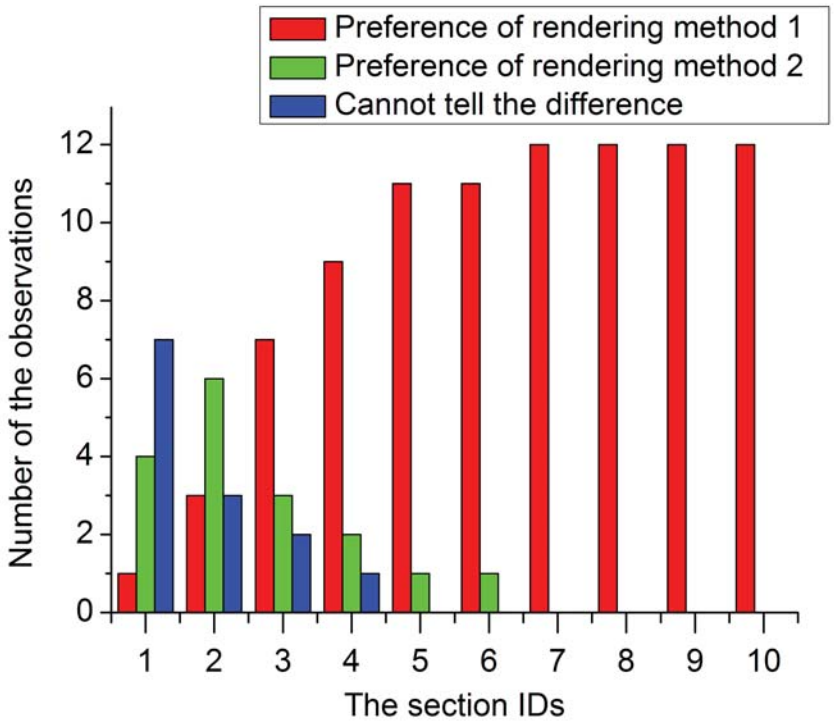


Figure 5.11: Test result of test pattern 4: the estimated blur radius and no-blur effect. The observers compared balls rendered by the estimated blur radius and no-blur effect.

the virtual object was static or moving, the test results did not change much. Second, the test results showed that the estimated blur method could help the observers to feel a better depth interpretation. The result of test pattern 1 to test pattern 4 showed that the estimated blur method made the observers feel the consistency between the virtual objects and the real scene. The result of test pattern 5 and 6 showed that in farther region, the observers could found the difference between the measured blur method and the estimated blur method. However, they showed no strong preference to either of the methods. The observers might make their choice due to personal perceptual habits.

5.4 User Test 3: Moving Virtual Objects with No Checkerboard Background

5.4.1 Test Settings

In user test 3, two moving 3D ball were placed in real image background. The radius of the virtual ball is 10mm, the same as the one in test 2. The texture of the balls is a soccer pattern. Two balls lining side by side were rolling from near region to far region. This time, there is no

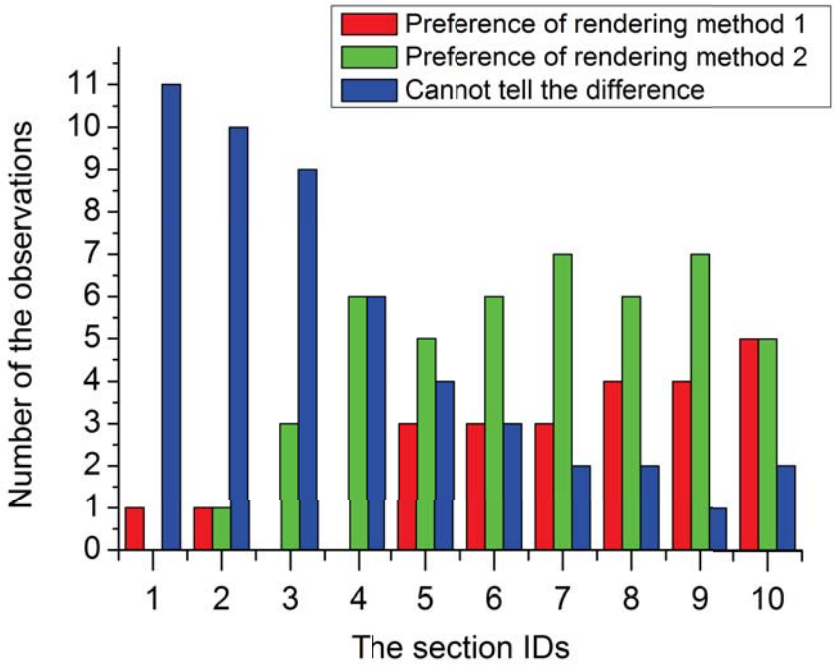


Figure 5.12: Test result of test pattern 6: the measured blur radius and the estimated blur radius. The observers compared balls rendered by the measured blur radius and the estimated blur radius.

checkerboard in the background. The track of the virtual balls is covered with black paper. The test patterns and other settings are all the same as the ones in test 2.

5.4.2 Task

The observers were required to compare the two balls and choose which one of the virtual balls seemed more natural and fit into the scene. The observers were told that the virtual balls were desired to move far away from them. The observers could also indicate that they cannot tell the difference of the two virtual contents. The number of the observations were collected for each test pattern.

5.4.3 Observers

6 observers between 20 and 30 years old were selected for the test. 85% of the observers were male. The observers all came from unrelated background other than computer vision and computer graphics and had little knowledge in image processing or rendering. Visual impairments such as myopia or astigmatism were not considered since the observers were wearing their corrective devices (glasses or lenses). 12 observations were collected for each test pattern.

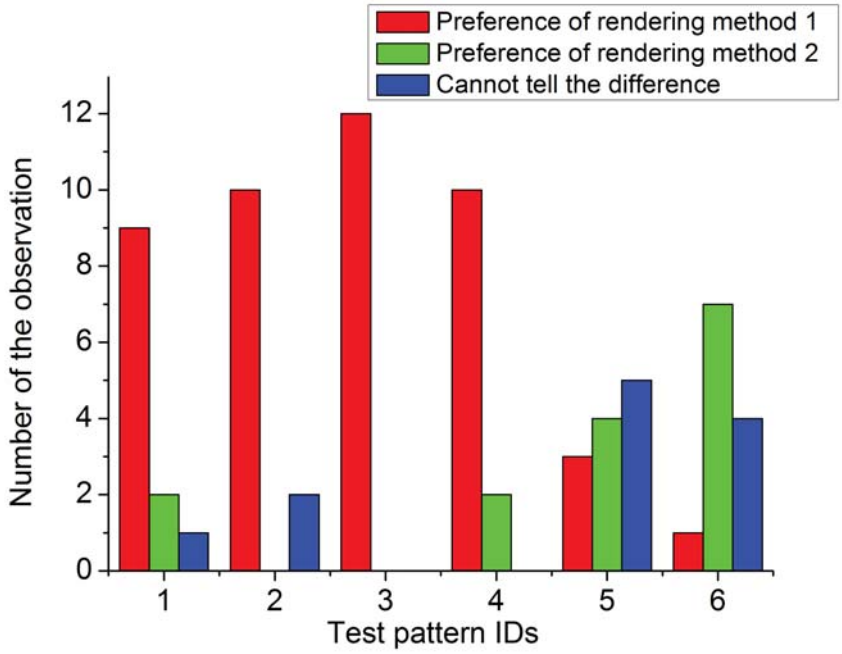


Figure 5.13: Test result of user test 3. The pattern IDs and its contents could be referred in Table 1.

5.4.4 Test Results

The test result showed that in the first four test patterns, most of the observers prefer the virtual balls rendered with the blur effect. In test pattern 5 and 6, many observers claimed they could tell the difference of the two rendering methods while the number of the observations show that they did not have a strong preference for either of the method.

5.4.5 Discussion

The test results show that even when the checkerboard was moved away, the estimated blur method still made the observers feel a better depth interpretation comparing to rendering no-blur to the virtual objects. The test results of test pattern 5 and 6 show that the number of the observations who chose the estimated method is larger than the number of the observations who chose the measured method. This might because that the observers expect motion blur in the moving process of the virtual ball.

5.5 Discussion and Conclusions

In this chapter, three user tests were conducted for different situations: 1. static 2D virtual contents with checkerboard background; 2. moving 3D virtual objects with checkerboard background; 3. moving 3D virtual objects without checkerboard in the background. From the test results, it could be confirmed that the user would prefer the more seamless scene created by blur effect. Although there is a difference between the value of the estimated blur and the measured blur, both blur method could help the user to feel a better depth interpretation in augmented reality systems.

The test results could not show which of the blur methods is more natural to user perception. In the original assumption, the more accurate the estimated method is, the more seamless scene could be rendered; the more seamless scene is rendered, the better depth interpretation the observers would perceive. However, the test results show that the observers sometimes prefer a more blurred virtual contents or a less blurred virtual contents comparing to the image quality of the real scene. This might because in certain situations, the observers would expect motion blur or just expect more detailed information. In order to know how to build up a blur estimation model natural to human perception, further tests on how the blur effect match the depth interpretation in quantity should be conducted.

Chapter 6

CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

In this thesis, a further discussion is conducted on how to use blur effect as depth cue in augmented reality applications. A new depth cue method is proposed. In the proposed method, the blur parameter of any point on the whole screen could be estimated according to its spatial information in the real world coordinate. The proposed method works by estimating the degree of blur based on the point spread function (PSF) parameter all over the scene of view. The proposed method also need to refer to the thin lens camera model and the intrinsic parameters of the camera, such as the focal length, the aperture diameter and the size of CCD sensor (Charge-coupled Device, the image sensor for digital imaging). The evaluation of the proposed method showed that the method could determine the degree of blur which should be rendered on the virtual objects in AR applications.

A prototype AR system was designed and implemented in order to solve the problem, how to render a proper degree of blur onto the virtual objects. The processing flow of the prototype system is divided into five steps: 1. detect the marker; 2. register the intrinsic and extrinsic parameters of the camera; 3. measure the defocusing (blur) parameters on the marker; 4. estimate the blur parameters in the whole view; 5. render the virtual objects with blur effect. The algorithm of the blur shader is also discussed.

Three user tests were conducted to solve the problem how the users would perceive the blur effect rendered by different blurring methods. The user tests confirmed that the proposed depth cue method could help the user to gain a better depth interpretation in AR system. In the user tests, a comparison between the estimated blur method (the proposed method) and the measured blur method (render blur effect on the position whose blur radius is measured) is addressed. However, the test results could not confirm which method is more natural to user

perception.

6.2 Future Work

In the future work, three aspects should be considered. First of all, the proposed depth cue method should be improved by taking the motion blur into account. Besides the blur effect caused by defocusing of the camera lens, the motion blur is also the key factor to cause the blur effect in augmented reality. The moving virtual object in a static scene might also cause some extent of motion blur. Therefore, the motion blur caused by the moving camera as well as the motion blur caused by the moving virtual object should both be studied.

For the prototype system implementation, a moving camera should be introduced and the real-time detecting and rendering should also be discussed. The camera view of the prototype system should no longer be limited to a desk but expanded to a broader view, for example, a ego-centric distance of 1.5m to 4.0m to the user. For the desk-top AR system (the prototype system in this thesis), it is hard to measure the observers' perception quantitatively. This is because the blur in the real scene of the desk-top AR system is not natural to human perception. The focal length of the prototype system is short and the focus depth is small. This made the camera view seem more blurred in farther region while the user could actually focus on anything on the desk. If the prototype could be expanded to a broader view, observers might feel a more natural depth perception both on the screen and out of the screen. In this situation, it is easier to require the observers to conduct tasks to match the blur effect and depth interpretation in quantity.

Last but not least, more user tests should be addressed to investigate on how the blur effect would match depth interpretation in quantity. In this thesis, the problem of how the users would perceive the blurred virtual contents as a depth cue is only partially solved. To solve this problem, more statistics should be collected. It is also possible to build up a blur model based on user tests results to see whether the user prefer a seamless blur or maybe other more contrastive blur. Meanwhile, whether the Gausssian blur shader is the best shader model or not should also be discussed by doing further user studies.

Appendix A

Blur Shader in GLSL

```
#ifdef GL_ES
precision mediump float;
precision mediump int;
#endif
#define PROCESSING_TEXTURE_SHADER

uniform sampler2D texture;
uniform vec2 texOffset;
varying vec4 vertColor;
varying vec4 vertTexCoord;
uniform int blurSize;
uniform int horizontalPass; // 0 or 1 to indicate vertical or horizontal pass
uniform float sigma;

const float pi = 3.14159265;

void main() {
  float numBlurPixelsPerSide = float(blurSize / 2);
  vec2 blurMultiplyVec = 0 < horizontalPass ? vec2(1.0, 0.0) : vec2(0.0, 1.0);
  vec3 incrementalGaussian;
  incrementalGaussian.x = 1.0 / (sqrt(2.0 * pi) * sigma);
  incrementalGaussian.y = exp(-0.5 / (sigma * sigma));
  incrementalGaussian.z = incrementalGaussian.y * incrementalGaussian.y;
  vec4 avgValue = vec4(0.0, 0.0, 0.0, 0.0);
  float coefficientSum = 0.0;

  // Take the central sample first
```

```
avgValue += texture2D(texture, vertTexCoord.st) * incrementalGaussian.x;

if( avgValue.a == 0.0 ){
         gl_FragColor = vec4(0.0, 0.0, 0.0, 0.0);
}else{

coefficientSum += incrementalGaussian.x;
incrementalGaussian.xy *= incrementalGaussian.yz;
for (float i = 1.0; i <= numBlurPixelsPerSide; i++) {
  avgValue += texture2D(texture, vertTexCoord.st - i * texOffset *
                        blurMultiplyVec) * incrementalGaussian.x;
  avgValue += texture2D(texture, vertTexCoord.st + i * texOffset *
                        blurMultiplyVec) * incrementalGaussian.x;
  coefficientSum += 2.0 * incrementalGaussian.x;
  incrementalGaussian.xy *= incrementalGaussian.yz;
}

gl_FragColor = avgValue / coefficientSum;
}
```

Bibliography

[1] Ariely, D. Occlusion edge blur: a cue to relative visual depth.

[2] Azuma, R., Baillot, Y., Behringer, R., Feiner, S., Julier, S., and MacIntyre, B. Recent advances in augmented reality. *Computer Graphics and Applications, IEEE 21*, 6 (2001), 34–47.

[3] Azuma, R. T., et al. A survey of augmented reality. *Presence 6*, 4 (1997), 355–385.

[4] Baba, M., Asada, N., Oda, A., and Migita, T. A thin lens based camera model for depth estimation from defocus and translation by zooming. In *Proceedings of* (2002), vol. 107, pp. 274–281.

[5] Bradski, G., and Kaehler, A. *Learning OpenCV: Computer vision with the OpenCV library*. O'reilly, 2008.

[6] Breen, D. E., Whitaker, R. T., Rose, E., and Tuceryan, M. Interactive occlusion and automatic object placement for augmented reality. In *Computer Graphics Forum* (1996), vol. 15, Wiley Online Library, pp. 11–22.

[7] Burdea, G., and Coiffet, P. Virtual reality technology. *Presence: Teleoperators and virtual environments 12*, 6 (2003), 663–664.

[8] Drascic, D., and Milgram, P. Perceptual issues in augmented reality. In *Electronic Imaging: Science & Technology* (1996), International Society for Optics and Photonics, pp. 123–134.

[9] Fischer, J., Cunningham, D., Bartz, D., Wallraven, C., Bülthoff, H., and Strasser, W. Measuring the discernability of virtual objects in conventional and stylized augmented reality. In *Proceedings of the 12th Eurographics conference on Virtual Environments* (2006), Eurographics Association, pp. 53–61.

[10] Fuhrmann, A., Hesina, G., Faure, F., and Gervautz, M. Occlusion in collaborative augmented environments. *Computers & Graphics 23*, 6 (1999), 809–819.

[11] Haller, M., and Landerl, F. A mediated reality environment using a loose and sketchy rendering technique. In *Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality* (2005), IEEE Computer Society, pp. 184–185.

[12] http://endigy.com/$2.7_A R_m edicine.php. Medicalardeveloper, endigyltd.$

[13] http://ros.org/wiki/camera$_p ose_c alibration. Api : camera_p ose_c alibration.$

[14] http://www.3ders.org/articles/20130105-inition-develops-ipad-based-augmented-3d printing.html. Inition develops ipad-based augmented 3d-printing, 1 2013.

[15] http://www.infsoft.de/navigation/. Navigation ar application by infsoft.

[16] Huang, W., Alem, L., and Livingston, M. *Human factors in augmented reality environments*. Springer, 2012.

[17] Kato, H. Artoolkit: library for vision-based augmented reality. *IEICE, PRMU* (2002), 79–86.

[18] Kato, I. P. H., Billinghurst, M., and Poupyrev, I. Artoolkit user manual, version 2.33. *Human Interface Technology Lab, University of Washington* (2000).

[19] Kessenich, J., Baldwin, D., and Rost, R. The opengl shading language. *Language version 1* (2004).

[20] Kessenich, J., Baldwin, D., and Rost, R. The opengl shading language. *Language version 1* (2004).

[21] Kim, S. K., Park, S. R., and Paik, J. K. Simultaneous out-of-focus blur estimation and restoration for digital auto-focusing system. *Consumer Electronics, IEEE Transactions on 44*, 3 (1998), 1071–1075.

[22] Kruijff, E., Swan, J., and Feiner, S. Perceptual issues in augmented reality revisited. In *Mixed and Augmented Reality (ISMAR), 2010 9th IEEE International Symposium on* (2010), IEEE, pp. 3–12.

[23] Leroy, L., Fuchs, P., and Moreau, G. Real-time adaptive blur for reducing eye strain in stereoscopic displays. *ACM Transactions on Applied Perception (TAP) 9*, 2 (2012), 9.

[24] Li, Y., Li, D.-X., Wang, L.-H., and Zhang, M. Real time stereo rendering for augmented reality on 3dtv system. In *Systems and Informatics (ICSAI), 2012 International Conference on* (2012), IEEE, pp. 2125–2129.

[25] Loftus, G. R., and Harley, E. M. Why is it easier to identify someone close than far away? *Psychonomic Bulletin & Review 12*, 1 (2005), 43–65.

[26] Mather, G. The use of image blur as a depth cue. *PERCEPTION-LONDON- 26* (1997), 1147–1158.

[27] Mather, G., and Smith, D. R. Depth cue integration: stereopsis and image blur. *Vision research 40*, 25 (2000), 3501–3506.

[28] McIlhagga, W. H., and May, K. A. Optimal edge filters explain human blur detection. *Journal of Vision 12*, 10 (2012).

[29] Neider, J., Davis, T., and Woo, M. *OpenGL. Programming guide*. Addison-Wesley, 1997.

[30] Okumura, B., Kanbara, M., and Yokoya, N. Augmented reality based on estimation of defocusing and motion blurring from captured images. In *Proceedings of the 5th IEEE and ACM International Symposium on Mixed and Augmented Reality* (2006), IEEE Computer Society, pp. 219–225.

[31] Okumura, B., Kanbara, M., and Yokoya, N. Image composition based on blur estimation from captured image for augmented reality. In *Proc. of IEEE Virtual Reality* (2006), pp. 128–134.

[32] O'Shea, R. P., and Govan, D. G. Blur and contrast as pictorial depth cues1. *Perception 26* (1997), 599–612.

[33] Pentland, A. P. A new sense for depth of field. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 4 (1987), 523–531.

[34] Pentland, A. P. A new sense for depth of field. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 4 (1987), 523–531.

[35] Phillips, L., Ries, B., Interrante, V., Kaeding, M., and Anderson, L. Distance perception in npr immersive virtual environments, revisited. In *Proceedings of the 6th Symposium on Applied Perception in Graphics and Visualization* (2009), ACM, pp. 11–14.

[36] Rheingold, H. *Virtual Reality: Exploring the Brave New Technologies*. Simon & Schuster Adult Publishing Group, 1991.

[37] Rossmann, K. Point spread-function, line spread-function, and modulation transfer function tools for the study of imaging systems. *Radiology 93*, 2 (1969), 257–272.

[38] Rost, R. J. *Open GL: Shading Language*. Addison-Wesley Professional, 2004.

[39] Sanches, S. R. R., Tokunaga, D. M., Silva, V. F., Sementille, A. C., and Tori, R. Mutual occlusion between real and virtual elements in augmented reality based on fiducial markers. In *Applications of Computer Vision (WACV), 2012 IEEE Workshop on* (2012), IEEE, pp. 49–54.

[40] Selesnick, I. W., and Burrus, C. S. Generalized digital butterworth filter design. *Signal Processing, IEEE Transactions on 46*, 6 (1998), 1688–1694.

[41] Shah, M. M., Arshad, H., and Sulaiman, R. Occlusion in augmented reality. In *Information Science and Digital Content Technology (ICIDT), 2012 8th International Conference on* (2012), vol. 2, IEEE, pp. 372–378.

[42] Surdick, R. T., Davis, E. T., King, R. A., and Hodges, L. F. The perception of distance in simulated visual displays- a comparison of the effectiveness and accuracy of multiple depth cues across viewing distances. *Presence: Teleoperators and Virtual Environments 6*, 5 (1997), 513–531.

[43] Sutherland, I. E. A head-mounted three dimensional display. In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I* (1968), ACM, pp. 757–764.

[44] Veas, E. E., Mendez, E., Feiner, S. K., and Schmalstieg, D. Directing attention and influencing memory with visual saliency modulation. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2011), ACM, pp. 1471–1480.

[45] Vishwanath, D., and Blaser, E. Retinal blur and the perception of egocentric distance. *Journal of Vision 10*, 10 (2010).

[46] Watt, S. J., Akeley, K., Ernst, M. O., and Banks, M. S. Focus cues affect perceived depth. *Journal of Vision 5*, 10 (2005).

[47] Weghorst, S. J., Sieburg, H. B., and Morgan, K. S. *Medicine meets virtual reality*. IOS Press, 1996.

[48] Xiong, Y., and Shafer, S. A. Depth from focusing and defocusing. In *Computer Vision and Pattern Recognition, 1993. Proceedings CVPR'93., 1993 IEEE Computer Society Conference on* (1993), IEEE, pp. 68–73.

[49] Yang-Mao, S.-F., Lin, Y.-T., Lin, M.-H., Zeng, W.-J., and Wang, Y.-l. Evaluation of mono/binocular depth perception using virtual image display. In *Human-Computer Interaction. Towards Intelligent and Implicit Interaction*. Springer, 2013, pp. 483–490.