

**YAŞAR UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCE**

(MASTER'S THESIS)

**A HOME DECORATING APPLICATION USING
AUGMENTED REALITY**

Uğur Çağrı DOĞAN

Thesis Advisor: Asst. Prof. Dr. Korhan KARABULUT

Department of Computer Engineering

Bornova-İZMİR

2013

This study titled “A Home Decorating Application Using Augmented Reality” and presented as Master’s Thesis by Uğur Çağrı DOĞAN has been evaluated in compliance with the relevant provisions of Y.U Graduate Education and Training Regulation and Y.U Institute of Science Education and Training Direction and jury members written below have decided for the defense of this thesis and it has been declared by consensus of votes that the candidate has succeeded in thesis defense examination dated 27/09/2013.

Jury Members:**Signature:**

Head : Asst. Prof. Dr. Korhan KARABULUT

Rapporteur Member : Asst. Prof. Dr. Mete EMİNAĞAOĞLU

Member : Assoc. Prof. Dr. Aybars UĞUR
(Ege University)

ÖZET

ARTIRILMIŞ GERÇEKLİK KULLANAN EV DEKORASYON UYGULAMASI

DOĞAN, Uğur Çağrı

Yüksek Lisans Tezi, Bilgisayar Mühendisliği
Tez Danışmanı: Yrd. Doç. Dr. Korhan KARABULUT
Eylül 2013, 49 sayfa

Artırılmış Gerçeklik, bulunduğumuz gerçek ortamı bilgisayar yardımı ile oluşturulmuş sanal nesnelere ile birleştiren bir sistemdir. Sanal Gerçeklik, nesnelere tamamen sanal bir ortamda sergilediğinden, gerçek ortamda sanal nesnelere sergileyebilmek için Artırılmış Gerçeklik daha etkilidir.

Bu tez, 3 boyutlu mobilya modellerini kullanıcının bulunduğu gerçek ortamda (özellikle kapalı ortamlarda), dekor değişikliklerini önizleme amacıyla görüntüleyen bir Artırılmış Gerçeklik uygulaması sunmaktadır. Tezin amacı, kullanıcılara, buldukları gerçek ortamda 3 boyutlu mobilya modellerini ve ev tasarım eşyalarını göz önüne getirme imkânını verip, satın alma sürecinde kolaylık sağlamaktır. Bu uygulama, 3 boyutlu modellerin görselleştirilmesi için üç teknik kullanmaktadır. İlki, modeli sahneye yerleştirebilmek için işaretleyici takibi yapan, işaretleyici tabanlı bir tekniktir. İkinci teknik de birincisi gibi işaretleyici tabanlı bir tekniktir; ancak birden fazla işaretleyiciyi takip eder. Her bir işaretleyici farklı mobilya modellerine atanmıştır. Ve sonucusu, gerçek ortamdaki herhangi bir nesneyi, kullanıcıların takip işlemi için kullanabilmesini sağlayan işaretleyicisiz bir tekniktir.

Ayrıca, bu tez SIFT ve eş düzlemlili POSIT algoritmalarının sonuçlarını birleştiren bir bakış açısı sağlamaktadır. Önerilen sistem farklı ışıklandırma durumlarında başarıyla test edilmiştir.

Anahtar Kelimeler: Artırılmış Gerçeklik, Ev Dekorasyonu, Sanal Gerçeklik, Mobilya, Görselleştirme, Bilgisayar Grafikleri, Görüntü İşleme.

ABSTRACT**A HOME DECORATING APPLICATION USING AUGMENTED
REALITY**

DOĞAN, Uğur Çağrı

M. Sc. Thesis in Computer Eng.

Supervisor: Asst. Prof. Dr. Korhan KARABULUT

September 2013, 49 pages

Augmented Reality is a system that combines real-world environment with computer generated virtual objects. It is more effective to exhibit an object in its real environment than Virtual Reality, since Virtual Reality places the design into a virtual environment.

This thesis presents an augmented reality application that visualizes 3D furniture models in the user environment (especially indoor scene) for the purpose of previewing decor changes. The objective of the study in this thesis is to allow users to visualize 3D furniture models and home design items in their environment using Augmented Reality and to provide convenience before purchase. The application uses three techniques to visualize the models. The first one is a marker-based technique that tracks a single marker to place the models on the scene. The second one is also a marker-based technique which tracks multiple markers. Every one of these markers is assigned to a furniture model. The final technique is a markerless one that allows the users to use any object in the real environment for tracking.

This thesis also provides a perspective that integrates the outcomes of SIFT and Coplanar POSIT algorithms. The proposed system has been successfully tested in different lighting conditions.

Keywords: Augmented reality, Home Decoration, Virtual Reality, Furniture, Visualization, Computer Graphics, Image Processing.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to my advisor Asst. Prof. Dr. Korhan KARABULUT for his continuous support, patience, motivation, enthusiasm throughout this study and his immense knowledge which added a considerable value to this thesis.

Besides my advisor, I would like to thank my family for all their love and encouragement and the rest of my thesis committee: Assoc. Prof. Dr. Aybars UĞUR (Ege University) and Asst. Prof. Dr. Mete EMİNAĞAOĞLU for their encouragement, insightful comments, and questions.

TEXT OF OATH

I declare and honestly confirm that my study titled “A Home Decorating Application Using Augmented Reality”, and presented as Master’s Thesis has been written without applying to any assistance inconsistent with scientific ethics and traditions and all sources I have benefited from are listed in bibliography and I have benefited from these sources by means of making references.

27/09/2013

Uğur Çağrı DOĞAN

TABLE OF CONTENTS

	<u>Page</u>
ÖZET	v
ABSTRACT	vii
ACKNOWLEDGEMENTS.....	ix
INDEX OF FIGURES	xvi
INDEX OF ABBREVIATIONS.....	xviii
1. INTRODUCTION.....	1
2. AUGMENTED REALITY.....	4
2.1. Augmented Reality versus Virtual Reality	5
2.2. Components of Augmented Reality Systems	6
2.3. Augmented Reality Displays	7
2.3.1. Head-Up displays.....	7
2.3.2. Head-mounted displays	7
2.3.3. Handheld displays.....	7
2.4. A Brief History of Augmented Reality	8
2.5. Why Augmented Reality?.....	14
3. APPLICATION AREAS OF AUGMENTED REALITY	15

3.1. Medical Applications	15
3.2. Advertising and Commercial	16
3.3. Education and Entertainment	17
3.4. Navigation	18
3.5. Military	19
4. TOPICS IN AUGMENTED REALITY	20
5. TECHNOLOGIES USED IN THE THESIS	22
5.1. Windows Presentation Foundation (WPF)	22
5.1.1. WPF architecture	22
5.1.2. Main features of WPF	23
5.2. ARToolKit	25
5.3. Open CV and Emgu CV	26
5.4. AForge.NET Framework	27
5.5. Pose From Orthography and Scaling with Iterations (POSIT) Using Coplanar Feature Points	28
5.6. Scale Invariant Feature Transform (SIFT)	32
6. A HOME DECORATING APPLICATION USING AUGMENTED REALITY	35
6.1. Application Scenario	35

6.2. Features of the Application	36
6.3. The Methodology	38
7. EXPERIMENTAL RESULTS	42
7.1. Results from Marker-Based Tracking	42
7.2. Results from Markerless Tracking	43
8. DISCUSSION	45
BIBLIOGRAPHY	46

INDEX OF FIGURES

<u>FIGURE</u>	<u>Page</u>
Figure 2.1: Reality-Virtuality Continuum.	4
Figure 2.2: The Sensorama Machine.	8
Figure 2.3: The world’s first head-mounted display “Sword of Damocles”.	9
Figure 2.4: A demonstration of Videoplace system.	9
Figure 2.5: A player wearing the ARQuake equipment.	11
Figure 2.6: A game of AR Chess demonstrates collaboration of the stationary user (left) and the mobile user (right) as seen by the "documentation camera" user. ...	11
Figure 2.7: Video see-through example on a consumer cell-phone.	12
Figure 2.8: Google Glass.	13
Figure 2.9: Google search requests for “augmented reality” and “virtual reality” between years 2004 and 2013.	14
Figure 3.1: Bichlmeier et al. system for viewing through the patient’s skin.	16
Figure 3.2: AR-Drone controlled by an iPhone AR application.	18
Figure 3.3: Wikitude Navigation application working on a smartphone.	18
Figure 5.1: The major components of WPF architecture and communication between managed and unmanaged portions.	22
Figure 5.2: How ARToolkit works.	25
Figure 5.3: Perspective projection for object points M_i	28

INDEX OF FIGURES (cont.)

<u>FIGURE</u>	<u>Page</u>
Figure 5.4: The POSIT algorithm for coplanar points.....	30
Figure 5.5: Gradient magnitudes and orientations of a Gaussian blurred image...33	33
Figure 5.6: A sample histogram for SIFT algorithm.	34
Figure 6.1: A sample screen from the application.....	36
Figure 6.2: Multimarker tracking.	37
Figure 6.3: ARToolKit marker samples.	37
Figure 6.4: The sequence diagram shows obtaining video frame process.	38
Figure 6.5: The image converted to binary image based upon a preset threshold.38	38
Figure 6.6: The sequence diagram shows the marker tracking process.	39
Figure 6.7: The 3D chair overlaying the marker.	39
Figure 6.8: ARToolKit calibration patterns.....	40
Figure 6.9: A sample result of using SIFT algorithm.....	41
Figure 7.1: The overlapped marker is not detected.	43
Figure 7.2: The marker cannot be found because of the camera angle.	43
Figure 7.3: A sample result of SIFT tracking. The selected area was found on the real-time image in average 78ms.....	44
Figure 7.4: A sample result of SIFT tracking. The selected pattern was found on the real-time image in average 117ms.	44

INDEX OF ABBREVIATIONSAbbreviations

6DOF	Six Degrees of Freedom
AR	Augmented Reality
AV	Augmented Virtuality
BARS	Battlefield Augmented Reality System
GA	Genetic Algorithms
GEP	Gene Expression Programming
GP	Genetic Programming
GPS	Global Positioning System
GPU	Graphics Processing Unit
GUI	Graphical User Interface
HMD	Head-Mounted Display
HMS	Helmet Mounted Sight
HUD	Head-Up Display
MIL	Media Integration Layer
MILCORE	Media Integration Library Core
PC	Personal Computer
PDA	Personal Digital Assistant

INDEX OF ABBREVIATIONS (cont.)Abbreviations

RFID	Radio Frequency Identification
SIFT	Scale Invariant Feature Transform
UI	User Interface
VR	Virtual Reality
WPF	Windows Presentation Foundation
XAML	Extensible Application Markup Language

1. INTRODUCTION

When costumers want to buy furniture, they keep thinking about whether the furniture would go well with their current décor. One of the most annoying things related with purchasing furniture is that it sometimes looks worse than how the customer imagined it to look when they buy the item. For this reason, many customers buy the furniture after a long time of thinking. This problem is aimed to be eliminated in this study.

Using one of the fastest expanding and hottest trends, which is Augmented Reality (AR) technology (most of the leading brands and technology providers have already embraced this technology to enhance their customers' experiences in exciting ways), this thesis provides three techniques to superimpose the computer generated 3D furniture models on the real environment in real time, so the customers will see the item from any angle and position and save their precious times.

The first technique is aimed to provide a solution that relies on a single marker. The marker will be detected and the 3D models will overlay the marker. The second technique allows detecting multiple markers. And finally, a third alternative solution that doesn't require a black and white marker for tracking will be presented.

There are several applications available with similar functionality. ViewAR is one of them. It is a worldwide AR catalog system which lets users visualize the 3D furniture models in their environment to support buying decision. Gyroscope tracking and framed marker tracking are available in the application. The system is released for iOS (ViewAR 2013).

uDecore is a customizable 3D Room Planner for iOS devices. The users can make use of a photo of their room or visualize detailed 3D furniture models using augmented reality with the camera live preview mode, but this system doesn't include a tracking method. While using the camera live preview mode, the user needs to rotate and position the furniture every time when the viewpoint is changed (Viutek 2013).

Most recently (August 2013), IKEA¹ has released a mobile application for Android and iOS platforms that uses augmented reality for placing most of the furniture in their 2014 catalog. This application can also use the IKEA catalog as a marker. The users place the printed IKEA catalogue where they want to put the furniture in their room, choose a product and see how it will look in their home (Souppouris 2013).

All of these applications have different features to place the 3D furniture in the real environment or to track the marker or objects. For instance, while uDecore doesn't use a tracking system, but the other applications have different tracking systems such as marker or gyroscope tracking. uDecore and IKEA's application can support hundreds of furniture models but ViewAR includes more than 2000 products. This thesis gathers up most of the prominent features from these applications and doesn't use a constant viewpoint. The user can move around the marker or the tracked object freely. In some situations, it is hard to use black and white markers because of the lighting. Unlike the other applications, the application in this thesis uses the three techniques previously in order to use the environment and lighting efficiently.

The thesis consists of eight chapters:

In Chapter 1, the aim of the project is given. The techniques used in the thesis to superimpose the 3D models on the real environment in real time are explained. The structure of the thesis is also described.

In Chapter 2, the description, requirements and a brief history of Augmented Reality, as well as the most important advances achieved in the history are reviewed.

In Chapter 3, the main application areas in which AR is most beneficial are described.

In Chapter 4, some of the issues of AR including technical and social challenges are discussed.

In Chapter 5, the technologies used in the thesis, including Windows Presentation Foundation, ARToolKit, Open CV, Emgu CV, AForge.Net

¹ IKEA is a worldwide leading home furnishing retailer.

Framework, POSIT algorithm using coplanar feature points, and scale invariant feature transform algorithm are explained.

In Chapter 6, a home decorating application using augmented reality is presented and in the following sections, the features of the application and the methods used in the application are described.

In Chapter 7, the results of the experiments of the presented system are discussed in two sections. The first section summarizes the results of the marker-based tracking and the second section summarizes the results of the markerless tracking.

Finally, Chapter 8 provides a brief overview of the thesis, including a statement of the problems and the major methods involved.

2. AUGMENTED REALITY

Augmented Reality (AR) can be defined as the interaction of virtual computer-generated information, audio and other sense enhancements over a real-world environment that is displayed in real-time. AR combines real and virtual objects. An AR system is both interactive (runs interactively, in three dimensions, and in real time) and registered (aligned) in 3D. The surrounding environment is replaced by virtual one. In contrast, AR provides local virtuality (Carmigniani and Furht 2011).

There are three important aspects of this definition. First, AR is not restricted to particular display technologies. Secondly, AR can be applied to all senses, including touch, hearing and smell. Finally, removing real objects by taking digital or computer generated information (whether it be images, audio, video, and touch or haptic sensations) and overlaying them in a real-time environment is also considered AR (Krevelen and Poelman 2010).

In 1994, Paul Milgram and Fumio Kishino defined Milgram's Reality-Virtuality Continuum (Figure 2.1). They describe a continuum that spans between the real environment and the pure virtual environment. And in-between these environments, there are Augmented Reality (closer to the real world) and Augmented Virtuality (closer to a pure virtual environment) (Milgram and Kishino 1994).

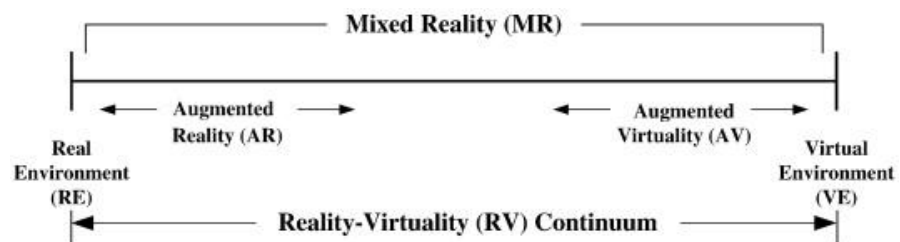


Figure 2.1: Reality-Virtuality Continuum.

2.1. Augmented Reality versus Virtual Reality

In this section, the technical differences and similarities between Augmented Reality and Virtual Reality are discussed.

First, let's begin with giving the definition of Virtual Reality (VR). VR is the computer generated scene where the users are completely immersed in a synthetic world without seeing the real world. The user represents an actor who plays an interactive role within the context of the computer generated three dimensional world.

Based on the previous definition, the most important difference between AR and VR is that virtual reality completely blocks out the real world whereas augmented reality allows the user to see three dimensional virtual objects superimposed upon the real world.

Standard methods of input devices are not adequate for user needs within a virtual environment. For example, a mouse is a 2D input device only allows two degrees of freedom. The user needs special gloves or joysticks, wearable sensors and head-mounted displays to interact with the virtual environment, whereas AR is not restricted to a specific display technology such as head-mounted display. AR applications can even be used with handheld displays.

The similarity between the AR and VR is that they both use various sources of information and programming to create visual, or other sensory, simulations for creating an experience (Carmigniani and Furht 2011).

2.2. Components of Augmented Reality Systems

An augmented reality system can contain a number of necessary components to make the whole process work and the different types of platforms that can be used for Augmented Reality. Kipper and Rampolla (Kipper and Rampolla, 2012) summarize the core components needed for both fixed and mobile environments as follows:

Hardware Components:

- A computer (either a PC or a mobile device)
- Display device
- A camera
- Tracking and sensing systems (GPS, compass, accelerometer etc.)
- A network infrastructure
- A marker that the computer identifies as the place where digital information is to be presented.

Software Components:

- An application or program running locally
- Web services
- A content server

Display devices, input devices, tracking systems and computers could be categorized and be described as the main devices for AR according to the components listed above.

2.3. Augmented Reality Displays

2.3.1. Head-Up displays

Head-Up Display (HUD) is an environmentally transparent fixed display. It can be mounted in a car or an aircraft. Head-Up means that the user doesn't have to look away to view the necessary information. A HUD consists of three primary components: A computer which provides the interface between HUD and the system, a projector unit is supplied with data by the computer (projects it onto the combiner) and a combiner unit which is usually made of glass with a special coating that reflects the monochromatic light from the projector (Carmigniani and Furht 2011).

2.3.2. Head-mounted displays

Head-Mounted Display (HMD) is defined as a display device that allows the user to see the virtual objects superimposed on the real world by video or optical technologies (Zhou, Duh and Billinghamurst 2008).

2.3.3. Handheld displays

Handheld displays provide small computing devices, which are highly mobile and supplied with a display unit. The users can hold them in their hands. Handheld displays use video-see-through techniques to superimpose graphics onto the user's real environment. Smartphones, tablet PCs, PDAs are some examples of these devices.

2.4. A Brief History of Augmented Reality

AR first appeared in 1950s. Morton Heilig thought cinema as an activity that could encompass all the senses in an effective manner (Carmigniani and Furht 2011). He applied his cinematographer experience and invented the machine named “Sensorama” which is a motorcycle simulator in 1957. The Sensorama Machine (Figure 2.2) was patented in 1962. It is one of the earliest known examples of multi-sensory, immersive technology that had visuals, sound, vibration, and smell.

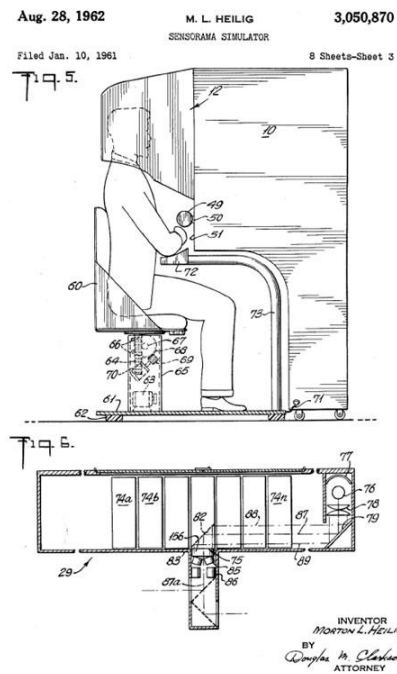


Figure 2.2: The Sensorama Machine a) The image from US patent #3050870, b) Morton Heilig, 1962.

In 1966, Ivan Sutherland invented the optical see-through HMD and by using this display, he created the first Augmented Reality system called The Sword of Damocles (Figure 2.3) in 1968 (Sutherland 1968).

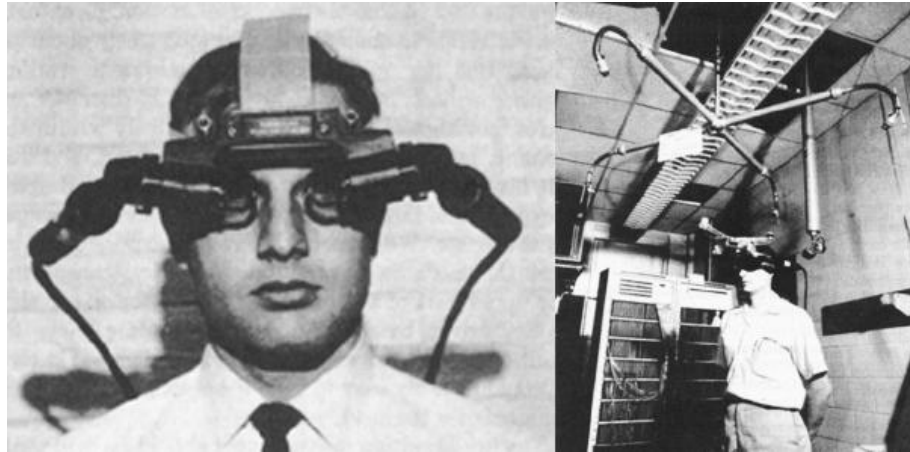


Figure 2.3: The world's first head-mounted display "Sword of Damocles" (Sutherland 1968).

In 1975, Myron Krueger launched a system named Videoplace (Figure 2.4). The system makes it possible for users to interact with graphics. Krueger describes Videoplace as a system that combines a participant's live video image with a computer generated world. It also coordinates the behavior of graphic objects so that they appear to react to the movements of the participant's image in real-time (Krueger 1985).

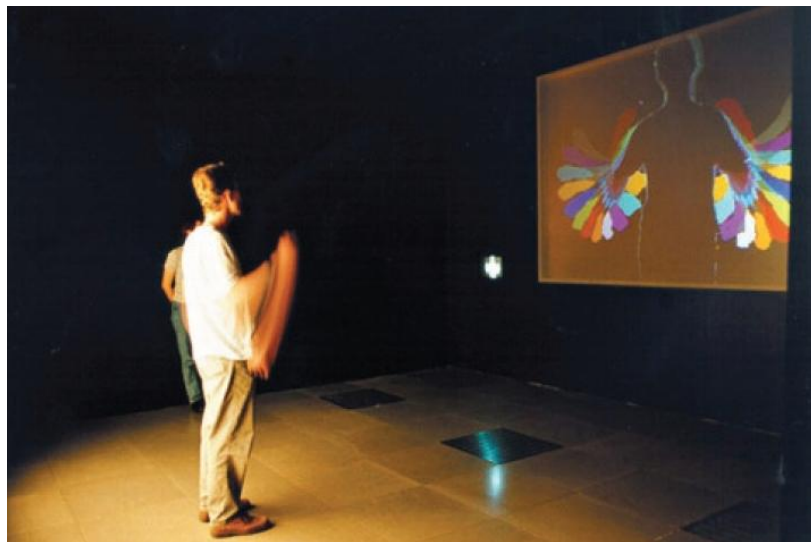


Figure 2.4: A demonstration of Videoplace system.

In 1993, L.B Rosenberg introduced Virtual Fixtures, one of the first functioning AR systems, to be used in telepresence systems and demonstrated that such fixtures can enhance operator performance within remote environments (Rosenberg 1993).

In 1996, Jun Rekimoto presented 2D matrix markers and developed a novel technique for producing augmented reality systems that simultaneously identify real objects and estimate their coordinate systems (Rekimoto 1998). The system was one of the first marker systems to allow camera tracking with six degrees of freedom (6DOF).

In 1997, Ronald Azuma (Azuma 1997) wrote the first survey on AR. Azuma provided the definition of AR, which is widely acknowledged, by identifying it as combining virtual and real environments and being registered in 3D as well as interactive in real time (Carmigniani and Furht 2011).

In 1999, the company named Total Immersion was founded and they entered the market as the first AR solutions provider. The company spent the next decade in research and development which eventually enabled them to become the market leader in AR. Then, in the same year, ARToolKit² was developed by Hirokazu Kato of Nara Institute of Science and Technology and was released by the University of Washington (Lamb 2013). This suite of tools enables video capture in the real world to be combined with virtual objects, to include 3D graphics, and can run on any operating system (Kipper and Rampolla 2012).

In 2000, Bruce Thomas et al. created outdoor/indoor application ARQuake which is an AR version of the popular game Quake. ARQuake (Figure 2.5) is a first-person AR view of the game which incorporated a six degrees of freedom (6DOF) tracking system, GPS, a digital compass, and vision based marker tracking. The keyboard and mouse interaction with the game is completely replaced with user movements and a dual button input device, and is fully functional within the accuracy of this tracking system (Thomas, et al. 2000).

² The detailed information is given in “ARToolKit” section.



Figure 2.5: A player wearing the ARQuake equipment.

In 2001, Gerhard Reitmayr and Dieter Schmalstieg created a mobile collaborative AR system (Figure 2.6) that supports true stereoscopic 3D graphics, a pen and pad interface and direct interaction with virtual objects (Reitmayr and Schmalstieg 2001). The ideas of mobile AR and collaboration between users in augmented shared space were combined and merged into a hybrid system.



Figure 2.6: A game of AR Chess demonstrates collaboration of the stationary user (left) and the mobile user (right) as seen by the "documentation camera" user.

In 2004, the first running video see-through AR system on a consumer cell-phone (Figure 2.7) was presented by Mathias Möhring et al. It supports the detection and differentiation of different markers, and correct integration of rendered 3D graphics into the live video stream via a weak perspective projection camera model and an OpenGL³ rendering pipeline (Möhring, Lessig and Bimber 2004).



Figure 2.7: Video see-through example on a consumer cell-phone.

In 2006, Nokia presented MARA, a multi-sensor mobile phone AR guidance application for mobile phones. To determine orientation, the application uses accelerometers in all three axes. A tilt compensated compass is used for heading and a GPS is used for positioning in the application.

In 2008, Wikitude World Browser with AR was launched. This is a mobile augmented reality software was developed by the Austrian company Wikitude GmbH (formerly Mobilizy GmbH). The application combines GPS and compass data with Wikipedia⁴ entries.

³ OpenGL is a cross-language, multi-platform application programming interface (API) for rendering 2D and 3D computer graphics.

⁴ Wikipedia is a collaboratively edited, multilingual, free Internet encyclopedia supported by the non-profit Wikimedia Foundation.

In 2012, Google has announced a new project named “Project Glass” which aims to build a pair of AR glasses (Figure 2.8). Google also released a concept video that shows some of the ways the AR glasses might be used. It can perform various day to day tasks, without the user ever moving his/her hands. Taking pictures, recording video, showing text messages as well as e-mails received, showing directions/locations by using integrated Google Maps and live chat are some of its features. It uses an Android operating system.



Figure 2.8: Google Glass (Google 2013).

As seen in this brief, history of AR and mobile AR has been around for a long time and many interesting developments are yet to come. In recent years, computers, mobile devices, graphic generators, and HMDs have been reinvented over and over again at a fascinating speed. Improvements on these technologies are promising. Of course, Augmented Reality also continues to develop and works its way into the modern technological landscape of today.

Nowadays, AR systems and AR applications are being produced with the new advances in technology. There exist cheap common devices (cell phones, tablets etc.) which could present AR scene. This means AR system components have made a transition from experimental to commercial domain. As it can be seen in Figure 2.9, normalized global search requests for "augmented reality" and “virtual reality” keywords obviously show that adoption of mobile augmented reality by common users is much faster than the adoption of previous applications.

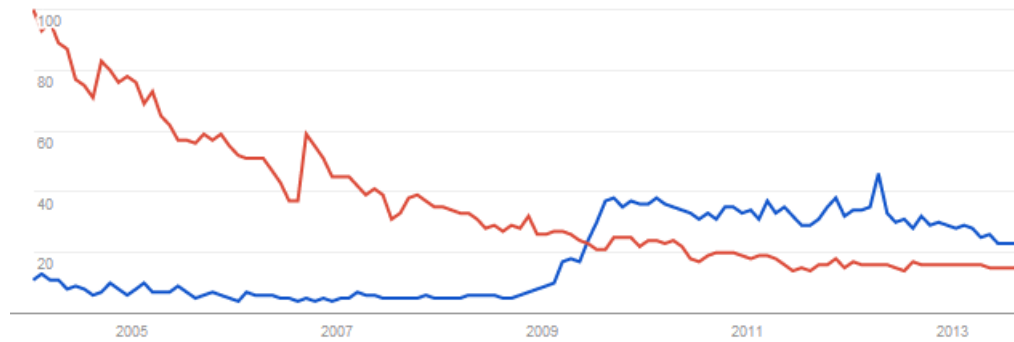


Figure 2.9: Google search requests for “augmented reality” and “virtual reality” between years 2004 and 2013. The red line denotes VR and the blue line denotes AR. The number 100 shows the peak level of search interest (Google 2013).

2.5. Why Augmented Reality?

Virtual Reality (VR) is a computer generated scene which requires high performance hardware to create a sufficient level of realism. Advanced computer, display, interaction and graphics technologies allow us to create photorealistic scene and view, hear, touch and even smell.. However, everything in that artificially created world is computer generated and is not real. VR is the complete immersion into a digital world (Carmigniani and Furht 2011). The VR user cannot sense the real world; but AR blurs the line between what's real and what's computer generated. AR improves the user's perception of the real-world environment. The primary objective of AR is to create the sensation that virtual objects are present in the real world to achieve this effect. AR software combines virtual objects with the real world. Generally speaking, adding virtual 3D object to real time video makes for a more impressive demonstration of AR technology.

3. APPLICATION AREAS OF AUGMENTED REALITY

In the early days, the first systems mostly focused on military, industrial and medical applications, but commercial and entertaining applications appeared soon after. Currently, there are many application areas imaginable for AR from the most serious and demanding areas to the less requiring ones. It is used in almost every field of life.

Azuma (Azuma 1997), Krevelen and Poelman (Krevelen and Poelman 2010) discussed and mainly categorized some application areas of AR in detail. In this section, these areas in which AR is most beneficial are summarized.

3.1. Medical Applications

AR has a big potential in the field of medicine. There are many AR researches and approaches for incorporating AR with devices incorporating the doctor's abilities and medical imaging. AR systems can be used for surgical training, surgery planning, image guided surgery and psychiatry.

In training situations on the simulators, there is no need for costly tissue or organs to be physically present, so surgeons receive training that allow them to gain experience in a realistic AR operating room without risk to actual patients.

In an operation, the risk can be reduced by AR system giving the surgeon improved sensory perception. The surgical team can see both the medical image and the medical data in real time while the operation is progressing. Bichlmeier developed an AR system (Bichlmeier, et al. 2007) (Figure 3.1) that creates a unique effect of observing the human body. It was developed to see through the patient's skin onto virtual anatomy in real time (Carmigniani and Furht 2011).

For treatment of phobias (claustrophobia, acrophobia, agoraphobia etc.), AR is used more than VR, because AR gives a greater feeling of presence than VR. The patient uses the real environment and the elements to interact with the application, whereas VR only simulates this experience (Juan, et al. 2005).



Figure 3.1: Bichlmeier et al. system for viewing through the patient's skin (Bichlmeier, et al. 2007).

3.2. Advertising and Commercial

AR applications promise new opportunities for marketing companies to interact with consumers. Many companies are using magazine advertisements and AR to give the viewer a full 3D view of the product being advertised. In 2008, MINI, the famous car company, placed an AR advertisement in various auto magazines in Germany. When the ad was viewed through a web cam, a virtual 3D model of their MINI Cabrio convertible was produced on screen (Geekologie 2008). This advertisement shows how marketers can integrate offline print ads with interactive online campaigns.

AR is not only used for advertising in online sources or on TV. As smartphones continue to be developed, AR experiences in mobile setting, away from the special environments of research laboratories and special-purpose work areas, become increasingly important (Hammad 2004). The film industry has also taken advantage of AR to promote movies. Marvel Comics created an AR application for Android devices (MarvelComics 2013). The application shows 3D action scenes starring the world's most popular Marvel super heroes by holding out selected Marvel products featuring the Marvel AR logo in front of the device's camera.

AR has not reached yet to its full potential in the industrial market and in advertisement. Some improvements such as technological advances in tracking system are needed.

3.3. Education and Entertainment

Augmented Reality can provide a rich customized learning environment and contents according to the human's input into the AR system. Augmented Reality enables individuals to learn the most when they control their own learning and manipulate virtual objects or representations of real objects that would otherwise be impossible to hold as well as learn tasks and skills in this environment. The advantage of AR learning is that there are no costs for making mistakes and errors, since they are not real. For example, Klopfer (Klopfer, et al. 2005) introduced a collaborative game, named "Mystery at the Museum", and aimed for museum education. The game was designed for synchronous play of groups of parents and children around two to three hour period. The primary design goals were to engage visitors more deeply in the museum and more broadly across museum exhibits, and to encourage collaboration between visitors.

Augmented reality also allows gamers to experience digital game play in a real world environment with animations and other multimedia presentations. AR can change the concepts of games such as; board games, strategic games, puzzle games, and even outdoor gaming are suitable for applying AR. There are currently many AR-enabled games available for both mobile devices and desktop computers.

The Parrot AR-Drone (Figure 3.2) is a radio controlled flying quadcopter⁵. It is produced by the French company Parrot. This flying toy is remote controlled by a mobile AR application. It was initially designed for the Apple platforms but now is available on all Android compatible devices. There is a camera on the Drone which lets the player see things from the Drone's point of view and allowing the player to have virtual dogfights with other Drones. In 2010, Parrot launched AR-Pursuit (Parrot, 2010), a multiplayer augmented reality game that demonstrates the extreme maneuverability and stability of the AR-Drone. One player pursues the other, attempting to bring him down with virtual machine guns and missiles.

⁵ A quadcopter is a helicopter that is lifted and propelled by four rotors.



Figure 3.2: AR-Drone controlled by an iPhone AR application.

3.4. Navigation

Navigation is known to be an existing potential for AR. AR Navigation applications focus more on enhancing reality than on generating a map. City guides that help people find places to eat, drink, and shop have Augmented Reality capabilities that give users real-time visual directions to the places they are looking for. These applications overlay a map through the device's camera. Unlike a GPS, the users can see the road itself with the overlay showing them where to go. Wikitude Navigation (Wikitude, 2010) is one of these applications popular nowadays. It integrates an augmented reality display and presents users with precise turn-by-turn GPS based guidance by eliminating the need for a map. In addition, it displays information about the users' surroundings including image recognition and 3D modeling (Figure 3.3).



Figure 3.3: Wikitude Navigation application working on a smartphone.

3.5. Military

The military is one of the areas where a huge amount of technology is incorporated. AR is one these technologies. The military systems use AR for navigational support, communications enhancement, repair and maintenance and emergency medicine.

HUDs and Helmet Mounted Sights (HMSs) have been used by the military organizations for a long time. An AR enabled head-mounted display could overlay blueprints or a view from a satellite or overheard drone directly onto the soldiers' field of vision. The pilots can aim the targets simply by looking at them. Another benefit of AR for military users may be training in large scale combat scenarios and simulating real time enemy action. A wider range of scenarios could be simulated without the need for extra hardware or manpower. Data or objects can be projected onto transparent helmet mounted displays for use in troop training exercises. However, the factors concerning mobility, usability and availability must be resolved.

The Battlefield Augmented Reality System (BARS) (Julier, et al. 2000) is one of these military systems. The system consists of a wearable computer, a wireless network system for communication and a tracked see-through HMD. The user's perception of the environment is enhanced by AR. The system also includes a database system containing all the important information to make available information to the military users.

4. TOPICS IN AUGMENTED REALITY

In this section, some of the current topics in AR including the technical and social challenges are described.

Object recognition or the “registration” problem is one of the basic problems recently limiting AR applications. The combined objects in the real and virtual worlds must be aligned correctly and be kept registered in real time. Otherwise, the sensation that virtual objects exist in the real environment, which is the essential goal of AR, is affected.

AR systems use one or more of the following tracking technologies that offer varying levels of accuracy and precision: digital cameras and/or other optical sensors, accelerometers, GPS, gyroscopes, solid-state compasses, RFID, and wireless sensors. Important sensor accuracy issues for the location and orientation sensors are found in current mobile devices. For example, smartphone location sensors (using Assisted GPS) exhibit errors with means of 10-30 m, depending on the surrounding buildings (Blum, Greencorn and Cooperstock 2012). Errors like this may affect an AR system uses that technology (for instance, a military AR system) in a bad way.

People might not want to depend on their mobile devices such as smartphones and tablets which have small screens on which to superimpose information. For this reason, wearable devices such as HMDs become more prominent, but the price of these devices is a constraining factor.

Another issue of AR is privacy. As described previously, a camera is one of the core components of an AR system, it is clear that the camera sees everything the user decides to point the camera at. Using image recognition software and facial recognition technology combined with geolocation and augmented data will potentially lead to a seamless integration of our online and offline activities. This means people will be able to point their phones at people and instantly see their digital profile on social networks and other information the people chooses to make available online.

The risk of real physical safety is another issue. Head-mounted displays like video see through types are essentially modified closed view. If the power is cut off that means the user is effectively blind. This becomes a safety concern in some applications. Mobile phones and tablets are currently a serious distraction while driving a car. A driver may become overwhelmed while trying to read a text message that just appeared on the AR interface. That situation may cause accidents, injuries or even death.

As it has been mentioned in the previous sections, AR has caught the attention of marketers and advertisers. This can cause unauthorized augmented advertising, spams and deceptive advertising techniques.

5. TECHNOLOGIES USED IN THE THESIS

5.1. Windows Presentation Foundation (WPF)

While tech-savvy people care mostly about technology, many software professionals are far more interested in how an application works. User interfaces and how they interact with users are more important for the users. An application's user interface is a major part of the entire user experience with that software. Providing a better user experience means more productivity. WPF provides that experience through better interfaces. Microsoft describes WPF as a next-generation presentation system for building Windows client applications with visually stunning user experiences.

5.1.1. WPF architecture

The major components of WPF architecture are Presentation Framework and Presentation Core (Managed WPF Layers) and Media Integration Layer (Unmanaged WPF Layer) (Ramteke 2012) (Figure 5.1).

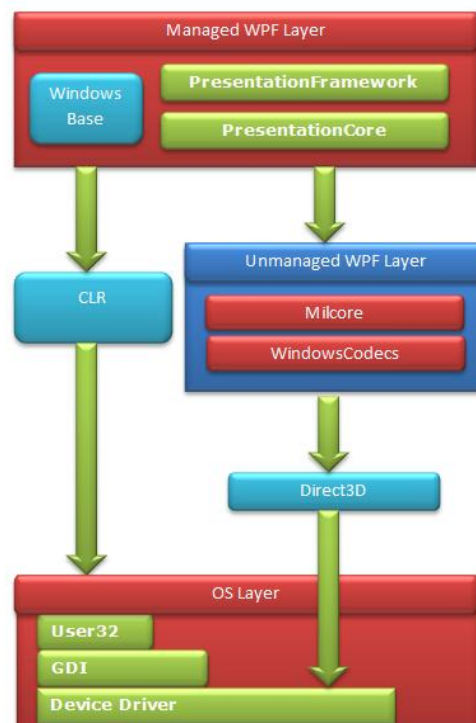


Figure 5.1: The major components of WPF architecture and communication between managed and unmanaged portions.

Managed layer contains WindowsBase, Presentation Framework and Presentation Core assembly. Presentation Framework is a managed component that holds the top level WPF and is the primary API for creating WPF applications. It provides functionality that we need to build the WPF applications such as layout panels, controls, styles and more. Presentation Core is a managed wrapper around Media Integration Layer (MIL) and provides public interface for MIL. It provides classes for creating application visual tree and holds base types such as UI Element and Visual System.

Media Integration Layer contains MilCore (Media Integration Library Core) and WindowsCodecs modules and both of them are unmanaged code. MilCore provides low level rendering functionality for WPF application. It is written in unmanaged code in order to enable a tight integration with DirectX. WindowsCodecs provides support for visualization like image processing, image displaying and scaling etc.

5.1.2. Main features of WPF

WPF is built on a vector based rendering engine that aims to take advantage of modern graphics hardware. It's easy to build scalable user interfaces. WPF also separates the appearance of a user interface from its behavior. The appearance is specified in the Extensible Application Markup Language (XAML) that forms a user interface markup language to define UI elements, data binding and other features. The behavior is generally implemented in programming languages such as C# and Visual Basic and it is associated with markup. This separation has some benefits as follows (Microsoft 2013):

- Application development and maintenance costs are reduced.
- Development becomes more efficient, because the developers can cooperate simultaneously with the designers while they implement an application's behavior.
- Globalization and localization for WPF applications is greatly simplified.

In this study, the most important feature of WPF is observed to be the interactive 3D. 3D elements can be fed with input, focus and events. WPF includes 3D rendering capabilities that integrate 2D graphics, so more astonishing UIs can be created.

All of the display activities in WPF are done through the DirectX engine which enables efficient hardware and software rendering. Graphics, including desktop items like windows, are rendered using Direct3D that allows Windows to offload some graphics tasks to the GPU.

WPF graphics system uses logical unit⁶ as the basic measurement unit. This provides resolution independent and device independent rendering. If the resolution of the screen is increased, then the user interface stays the same size.

Visual elements are not restricted to a uniform outfit such as a rectangular shape. They might have irregular borders. The display of an element can be defined directly via its properties or indirectly by defining a control template or a style.

WPF contains a handful of 2D transforms which enables changing the size, position, rotation angle and allows skewing. The following 2D Transform classes for common transformation operations are as follows:

- RotateTransform: Rotates an element by the given angle.
- ScaleTransform: Scales an element by the given ScaleX and ScaleY values.
- TranslateTransform: Moves an element by the given X and Y values.
- SkewTransform: Skews an element by the given AngleX and AngleY values.

WPF provides two classes for creating more complex transformations which are also used in this study. These classes are defined as follows:

- TransformGroup: Groups multiple TransformGroup objects into a single transform that you can then apply to transform properties.
- MatrixTransform: Creates an arbitrary affine matrix transformation that is used to manipulate objects or coordinate systems in a 2-D plane.

WPF supports time-based animations, in contrast to the frame-based approach. An efficient timing system that is exposed through managed code and XAML is included and that is deeply integrated into the WPF framework.

⁶ A logical unit is 1/96th of an inch.

Controls in WPF are extremely composable. Almost any type of controls could be defined as content of another.

WPF provides an integrated system to use the common media elements like vector images, raster images, audio, and video. WPF also provides an animation system and a 2D/3D rendering system. So if 3D graphics or multimedia is needed for an application, then there is no need to use different APIs. WPF provides all the necessary features.

5.2. ARToolkit

ARToolkit is a software library written in C programming language⁷ that can be used to calculate camera position and orientation relative to physical markers in real time. It allows developers to easily create Augmented Reality applications. It is distributed as an open source library for noncommercial or research applications. ARToolkit was originally developed by Dr. Hirokazu Kato and its development is supported by the Human Interface Technology Laboratory (HIT Lab) at the University of Washington, HIT Lab NZ at the University of Canterbury, New Zealand, and ARToolworks, Inc, Seattle.

ARToolkit uses video tracking capabilities to calculate the real camera position and orientation relative to physical markers in real time to solve the problem of tracking the user's viewpoint. If we know the real camera position, we can place a virtual camera at the same exact position, so a computer generated 3D model or image is placed to overlay the ARToolkit markers (Figure 5.2).

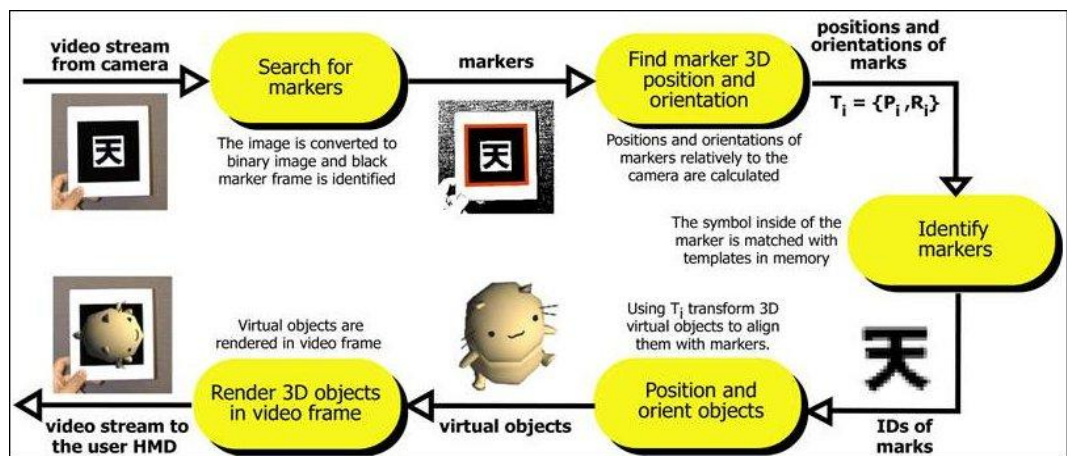


Figure 5.2: How ARToolkit works (Lamb 2013).

⁷ In this thesis, a C# wrapper code is used to access to ARToolkit library from Visual Studio.

While developing an augmented reality application that uses ARToolKit, the following steps must be taken in the main code:

1. Video capture is to be initialized, and the marker pattern (or patterns) and camera parameters need to be read.
2. A video input frame is captured from the camera.
3. The markers and recognized patterns in the video input frame are detected.
4. The camera transformation relative to the detected patterns is calculated.
5. The virtual objects are placed on the detected patterns.
6. Video capture is closed down.

The steps 1 and 6 are only performed in initialization and shutdown of the application respectively. The steps 2 through 5 form the main loop and they are repeated continuously until the application is closed. In addition to these steps, the application might need to respond to mouse, keyboard or other application specific events (Lamb 2013).

5.3. Open CV and Emgu CV

In this section, first Open CV is explained and then Emgu CV is discussed.

Open CV is an open source computer vision and machine learning software library. It was designed especially to provide a common infrastructure for computer vision applications and for computational efficiency on real time applications. It is written in optimized C/C++. Open CV takes advantage of multi-core processing. The library has more than 2500 optimized algorithms, which includes a comprehensive set of both classic and state-of-the-art computer vision and machine learning algorithms. These algorithms can be used to detect and recognize faces, identify objects, classify human actions in videos, track moving objects, extract 3D models of objects, produce 3D point clouds from stereo cameras, stitch images together to produce a high resolution image of an entire scene, find similar images from an image database, remove red eyes from images taken using flash, follow eye movements etc. (OpenCV 2013)

Emgu CV is a cross platform .Net wrapper for the Open CV image processing library and it is written entirely in C#. It allows Open CV functions to be invoked from .NET compatible languages such as C#, VB, VC++ etc. (EmguCV 2013)

Emgu CV has two layers of wrapper and they are defined as follows:

1. The basic layer (Layer 1) contains function, structure and enumeration mappings which directly reflect those in Open CV.
2. The second layer (Layer 2) contains classes that mix in the advantages from the .NET world.

5.4. AForge.NET Framework

AForge.NET is an open source C# framework originally developed by Andrew Kirillov for software developers and researchers in the fields of Computer Vision and Artificial Intelligence. The framework is comprised by the set of libraries and sample applications, which demonstrate their features (Kirillov 2007).

The image processing library contains a set of image processing filters and tools designed to address a lot of different tasks of computer vision and image processing.

The neural network library implements some common popular neural network concepts. It may be applied to a range of problems that can be solved with multi-layer feed-forward networks using supervised learning algorithms, or with self-organizing networks using unsupervised learning algorithms.

The evolution computation library implements several popular algorithms, such as Genetic Algorithms (GA), Genetic Programming (GP) and Gene Expression Programming (GEP). This makes it applicable to many different types of problems.

5.5. Pose From Orthography and Scaling with Iterations (POSIT) Using Coplanar Feature Points

Martins and Batista (Martins and Batista 2008) describes POSIT as an accurate and fast, iterative algorithm for finding the pose of an object or scene from a camera by given a set of 2D image and 3D object points correspondences (Dementhon and Davis 1995). The pose P of a 3D object is a combination of its orientation R (a 3D rotation matrix) and its position T (a 3D translation vector) relative to the camera.

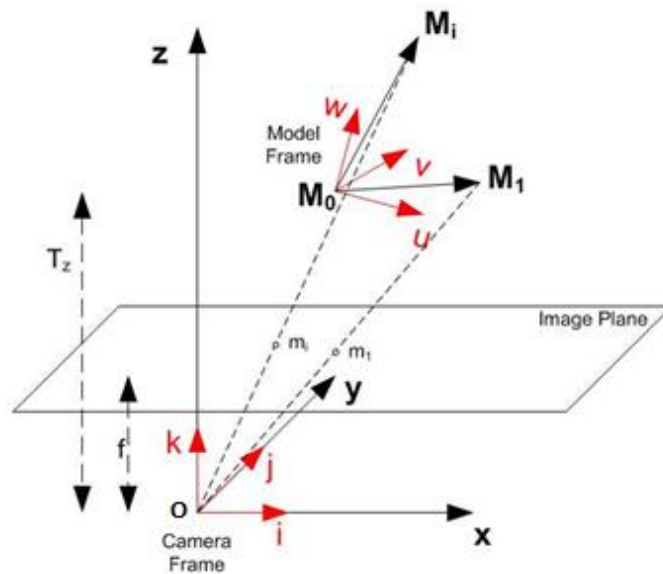


Figure 5.3: Perspective projection for object points M_i (Martins and Batista 2008).

In Figure 5.3, the pinhole camera model and the image plane are shown. O is the model's projection center. It is assumed that the center of the image and the distance f (the focal length) from O are known. i , j and k represents the unit vectors in the camera coordinate frame. $M_0, M_1, \dots, M_i, \dots, M_n$ represent the feature points of an object which is located in the field of view of the camera. M_0 is the center of the model coordinate frame with u , v and w unit vectors. Martins and Batista (Martins and Batista 2008) describe the pose matrix as following:

“An M_i point has known coordinates (U_i, V_i, W_i) in the model frame and unknown coordinates (X_i, Y_i, Z_i) in the camera frame. The projections of M_i are known and called m_i , having image coordinates (x_i, y_i) . The pose matrix \mathbf{P} provides the rigid transformation between the model and the camera frame is as follows:”

$$\mathbf{P} = \left[\begin{array}{c|c} \mathbf{R} & \mathbf{T} \\ \hline \mathbf{0} & 1 \end{array} \right] = \begin{bmatrix} i_u & i_v & i_w & T_x \\ j_u & j_v & j_w & T_y \\ k_u & k_v & k_w & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \mathbf{P}_1 \\ \mathbf{P}_2 \\ \mathbf{P}_3 \\ \mathbf{P}_4 \end{bmatrix}$$

In this resulting matrix, \mathbf{R} (the rotation matrix) represents the orientation of the camera frame with respect to the model frame and \mathbf{T} represents the translation vector from O to M_0 . The pose matrix rows are $\mathbf{P}_1, \mathbf{P}_2, \mathbf{P}_3$ and \mathbf{P}_4 . The rows of \mathbf{R} represent the coordinates of the unit vectors of camera frame expressed in the object coordinate frame (M_{0u}, M_{0v}, M_{0w}) . \mathbf{T} is the vector O to M_0 which is aligned with the vector O_{m0} . This vector has X_0, Y_0 and Z_0 coordinates, so, $T = (Z_0/f) O_{m0}$. If the unit vectors i and j and the coordinate Z_0 are known, the model pose can be defined. In a perspective projection model, a 3D point (defined by X_i, Y_i, Z_i) is projected in the image by

$$x_i = f \frac{X_i}{Z_i}, \quad y_i = f \frac{Y_i}{Z_i}.$$

Under scaled orthographic projection (weak perspective) model, a 3D image point projection can be written as

$$x_i = \frac{f}{(1+\epsilon)} \frac{X_i}{T_z}, \quad y_i = \frac{f}{(1+\epsilon)} \frac{Y_i}{T_z}.$$

In this situation, the depth of an object is assumed to be small compared to the distance of the object from the camera.

From the previous explanations, it seems that POSIT algorithm requires image coordinates of some object's feature points (minimum 4 points). But it should be noted that these points must not be coplanar. However, most AR applications are based on detection of markers which are planar. Fortunately, there is another algorithm that solves this problem which is known as Coplanar POSIT (Dementhon and Davis 1995).

Coplanar POSIT algorithm estimates the pose of an object with coplanar points. The flowchart for the posit algorithm for coplanar scene points is shown in Figure 5.4.

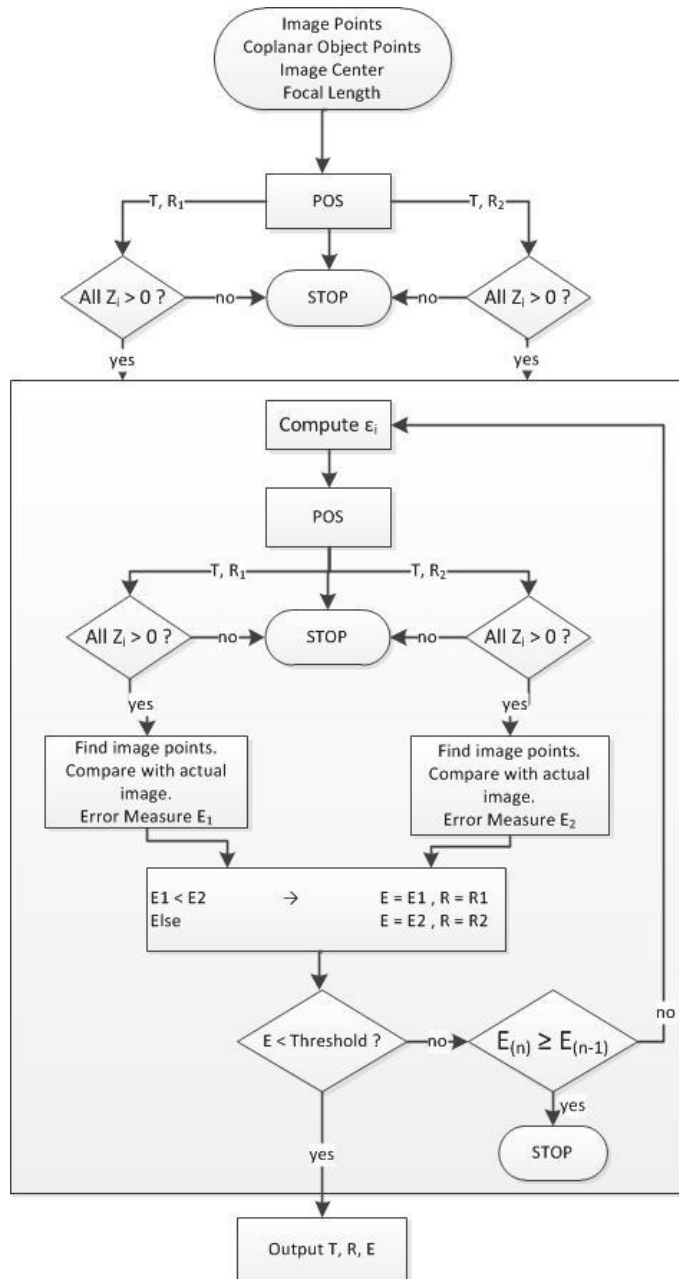


Figure 5.4: The POSIT algorithm for coplanar points (Dementhon and Davis 1995).

As it could be seen in the figure, there are two branches which are produced at the first step. If the z components of some visual points for the corresponding pose are negative (that means the points are behind the camera), one of them is dropped out. At the next and following iteration steps, only the process for the left branch is shown. This process is established to choose the better better pose at each iteration. If the distance measure E has decreased below a predefined threshold, the pose for the branch is the output result; along with the final distance measure E (Dementhon and Davis 1995).

Since coplanar POSIT algorithm estimates pose by using a collection of model points and 2D image projections, we need a way to extract these points. In this thesis, Scale Invariant Feature Transform (SIFT) algorithm (which is described in the following section) is used to find that key points.

5.6. Scale Invariant Feature Transform (SIFT)

Matching features across different images is a common problem in computer vision. Lowe (Lowe 1999) suggests that local image features which are unaffected by nearby clutter are required to recognize an object in cluttered real world scenes. The features must be at least partially invariant to scaling, rotating, common object variations and illumination. The features of the object must also be distinctive enough to identify a specific object among many alternatives.

Scale Invariant Feature Transform (SIFT) provides transforming an image into a large collection of local feature vectors. Each of them is invariant to rotation, scaling, translation and partially invariant to the changes in illumination.

The initial preparation is the first step. Internal representations of the original image are created to assure scale invariance. This is done by generating a scale space. Original image is taken and blurred out images are generated progressively (up to the octave range). Images of the same size form an octave (octave range is given by the user). Then the original image is reduced to half size and blurred out images are generated again from that resized original image.

Blurring process is done by a Gaussian kernel and its derivatives. It is known that the two-dimensional Gaussian function is separable, so if we apply two passes of the one-dimensional Gaussian function in both vertical and horizontal directions, its convolution with the input image is computed (Lowe 1999):

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2}$$

In the formula, x and y presents the location coordinates, and σ is the scale parameter.

The second step is generating another set of images using those blurred images were created in the previous step. Two consecutive images in an octave are selected and one is subtracted from the other. Then the next consecutive pair is taken, and the process repeats until all octaves are done. The resulting images are used to detect key points.

The third step is finding the maxima and minima in the resulting images generated in the previous step. This process is done by comparing neighboring pixels in the current scale, the scale above and the scale below. A pixel is compared to its neighboring pixels at the same level of the octave. If the pixel is a minima or maxima, then at the next lowest level of the octave, the closest pixel location is calculated by taking 1.5 times resample. If the pixel remains higher (or lower) than the closest pixel and its 8 neighboring pixels, then for the level above, the process is repeated.

The previous step produces a lot of key points. Some of them are useless since they lie along an edge or they don't have enough contrast. In the fourth step, the number of key points is reduced by rejecting the useless key points.

After all these steps, the legitimate key points which have been tested to be stable are deduced and scale invariance is achieved. Assigning an orientation to each key point is the next step. Gradient magnitudes and orientations (Figure 5.5) are calculated by using these formulas:

$$m(x, y) = \sqrt{(L(x + 1, y) - L(x - 1, y))^2 + (L(x, y + 1) - L(x, y - 1))^2}$$

$$\theta(x, y) = \tan^{-1}((L(x, y + 1) - L(x, y - 1)) / (L(x + 1, y) - L(x - 1, y)))$$

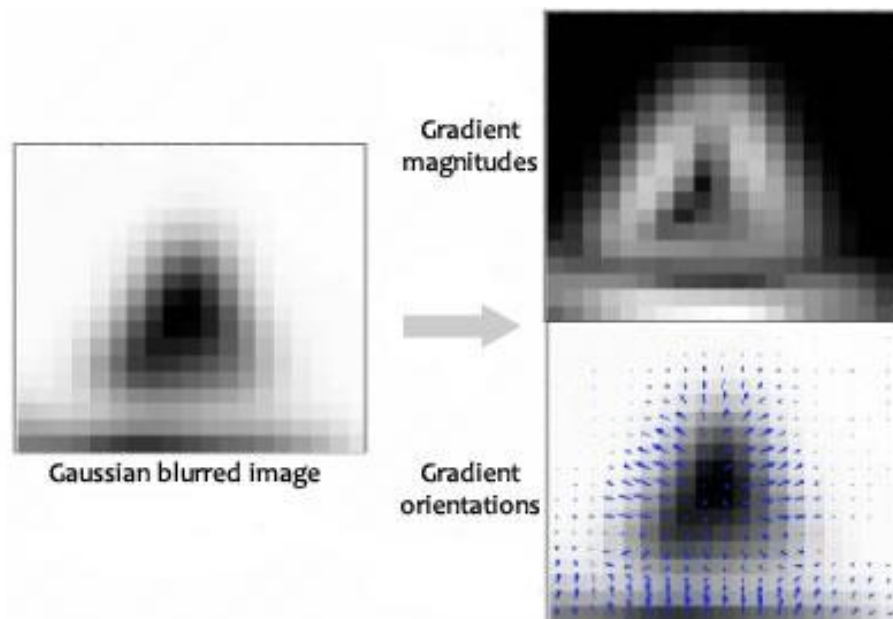


Figure 5.5: Gradient magnitudes and orientations of a Gaussian blurred image.

The magnitude and orientation are calculated for all pixels around the key point. Then, a histogram which has 36 bins is created (Figure 5.6). These bins cover the 360 degree range of rotations (each bin represents 10 degrees). The histogram is smoothed before the peak selection. To assign an orientation, this histogram and a small region in it are used. The most prominent gradient orientations are identified. If there is only one peak, it is assigned to the key point. If there are multiple peaks above the 80% mark, they are all converted into a new key point with their respective orientations.

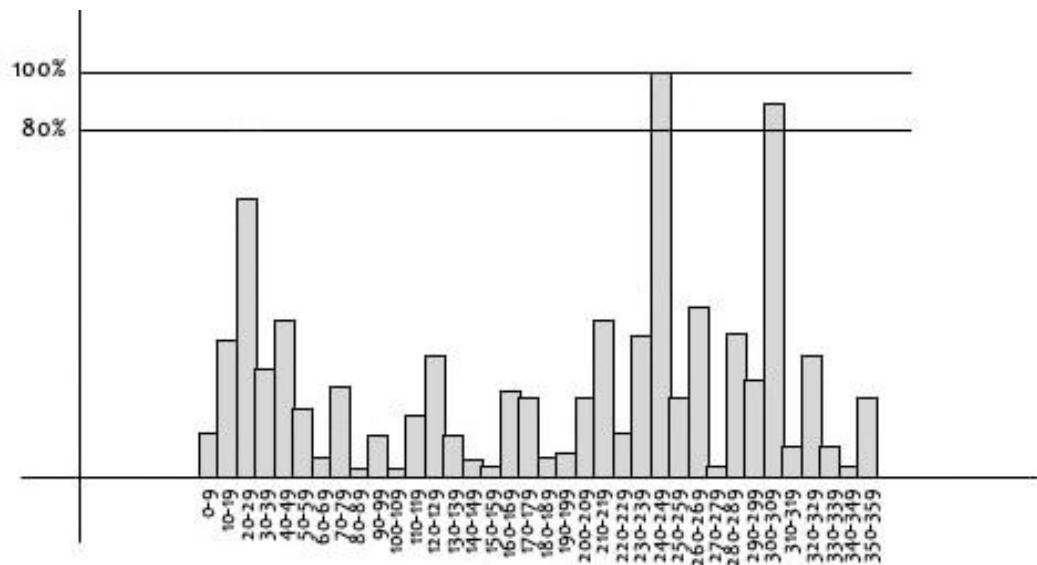


Figure 5.6: A sample histogram for SIFT algorithm.

In a nearest-neighbor indexing method which identifies object matches, the image key points are given as input. By finding a low-residual least-squares solution for the unknown model parameters, final verification of each match is achieved.

6. A HOME DECORATING APPLICATION USING AUGMENTED REALITY

Recently, the demand for augmented reality systems is on the rise due to the advanced computing devices and display technologies, requiring more stable and efficient visualization and user experience. Home decorating applications are one of the opportunities for implementing augmented reality inside home decorations.

Most of these applications use a photo of the user's room as the background to visualize 3D furniture models on it or use a single marker to visualize 3D furniture models in real time with augmented reality.

The most important part of this study is the application that combines and enhances different techniques used in home decorating applications with augmented reality. All these techniques and the details about the application are described in the following sections.

6.1. Application Scenario

The application scenario involves a user who intends to preview furniture models in the real environment before buying them. After setting up the system, a GUI (Figure 6.1) is presented according to the user's selection (marker-based / markerless). On the left hand side, a large view which presents the furniture in the AR interface is placed. Above it, there is a list which includes the cameras installed and available on the computer. On the right hand side, there is a list of different furniture models which is loaded from the database. Below it, the controls, which allow user to rotate the selected furniture model around a specific axis, are placed.

After initiating the interface, the user needs to select a camera from the list to start tracking process. When the process starts, the real time image of the environment which is taken from the selected camera is shown to the user. According to the chosen method, the tracked marker or a specific object which is chosen by the user is overlaid by using 3D furniture model. The user can move the objects by selecting the furniture model and holding the left mouse button down. Changing the rotation of a furniture model around x/y/z axis is also possible.

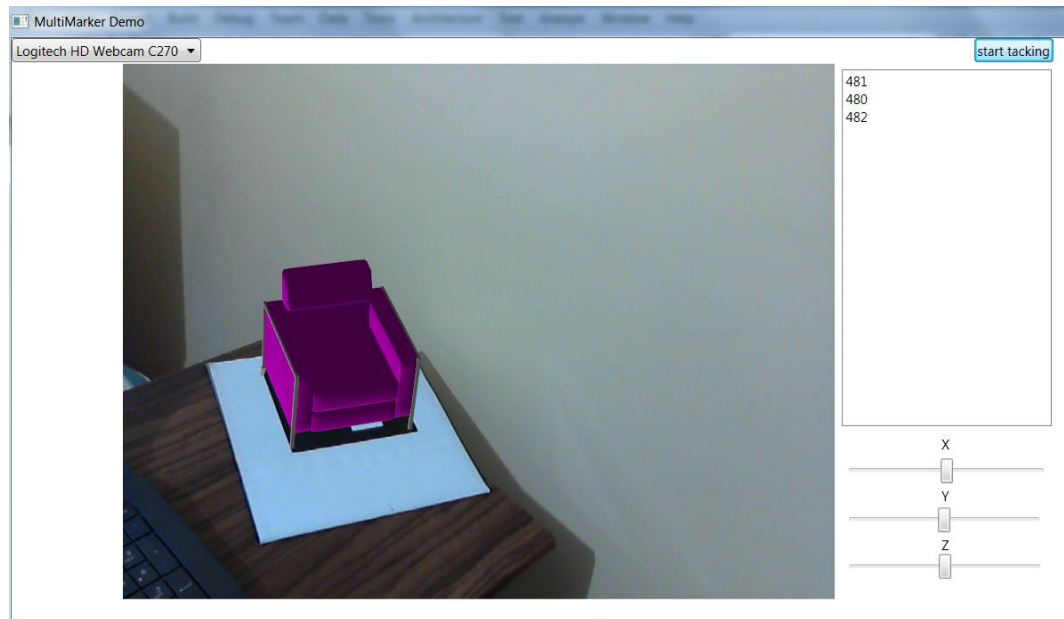


Figure 6.1: A sample screen from the application.

6.2. Features of the Application

The application uses three techniques to visualize the models in real time. The first technique is a marker-based technique that tracks a single marker. Many 3D furniture models can be placed around the single marker. The second one is also a marker-based technique but it tracks multiple markers to visualize the models. Every one of these markers is assigned to a different furniture model. And the last one is a markerless technique that allows the users to use any object in the environment such as a book or a painting for tracking. The detailed specification of objects makes it easier for tracking (more details means better tracking).

The single marker technique relies on a single marker which is found in the captured video frame. The marker is set up in the real-world environment. Then the application finds the marker and the selected 3D models overlay the marker. The models can be moved around depending on the marker location. Since this method relies on a single marker, accuracy may not always be satisfactory when the application runs in large scale environment.

The multiple marker technique allows detecting multiple markers from the captured video frame and provides the position and rotation of each marker (Figure 6.2).

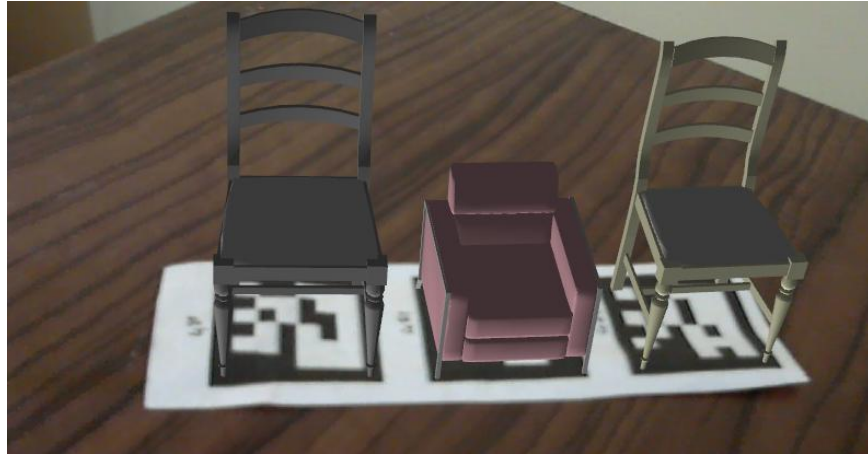


Figure 6.2: Multimarker tracking.

For the marker-based parts of the application, ARToolKit is used. ARToolKit markers (Figure 6.3) can consist of one or more basic shapes made up of black squares against a white background. The markerless technique will be described in the following section.

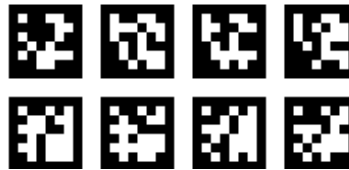


Figure 6.3: ARToolKit marker samples.

The devices needed for the application are just a desktop or laptop computer with Windows platform and a webcam, which are already available in most houses. There is no need for more expensive components.

The graphical user interface (GUI) of the application is designed using the WPF system. The 3D furniture models are exported from 3D studio MAX⁸ and are represented as meshes in XAML files.

The users can browse the furniture catalog and choose a category, pick any of the products on the list and preview it with color variations on the scene. The 3D product models can be moved, rotated and scaled. It is also possible to take a screenshot of the scene with the 3D models.

⁸ 3D Studio Max is a professional 3D modeling, animation, simulation, and rendering software for games, film, and motion graphics artists.

6.3. The Methodology

In order to build and execute AR applications, the minimum hardware requirements are a computer and a camera connected to it. As it is described previously, the first step in the process is capturing video from the camera. A filter graph is needed to achieve the capturing process. The filter graphs are used in multimedia processing. They take input data and process it. A video frame is obtained by running the filter graph. This process is shown in Figure 6.4.

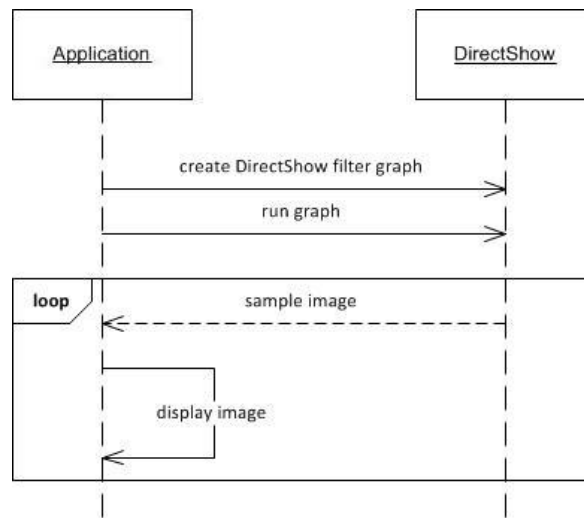


Figure 6.4: The sequence diagram shows obtaining video frame process.

The next step is detecting the markers and recognized patterns in the video input frame. To detect the markers, the image must be converted into a binary (black and white) image based upon a preset threshold value (Figure 6.5). ARToolKit uses binary image to search for square shaped regions.

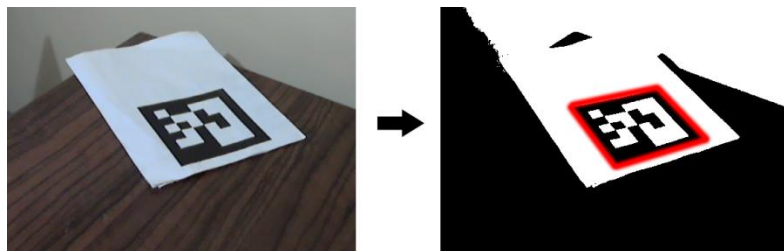


Figure 6.5: The image converted to binary image based upon a preset threshold.

All of the squares can be found in the image, but most of them are not the tracking markers. For every square image found, the pattern inside it is matched against predefined pattern templates. If a match is found, this means ARToolKit has found a tracking marker. Then ARToolKit calculates the position of the real camera relative to the physical marker by using the known square size and pattern orientation. Since the position of the camera is known, the virtual objects can precisely overlay the real marker (Figure 6.7). This process is shown in Figure 6.6.

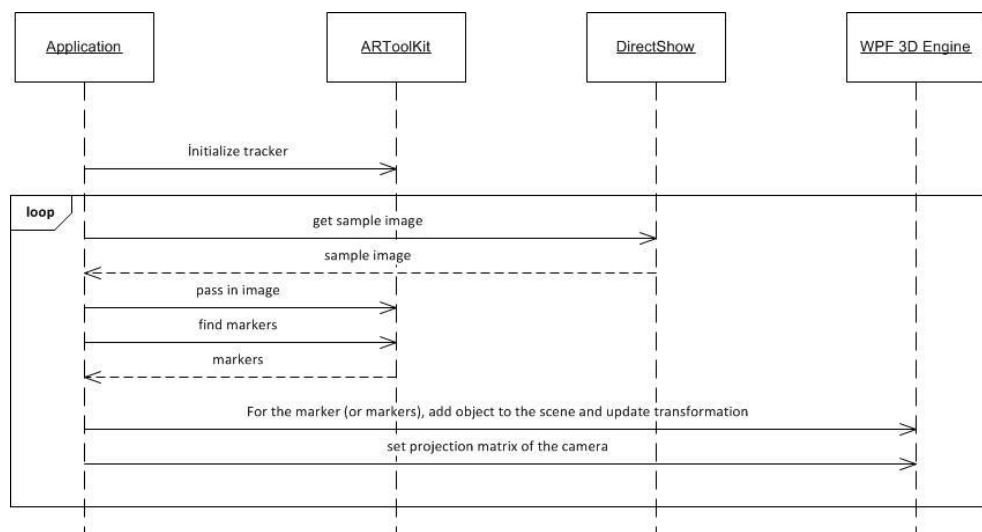


Figure 6.6: The sequence diagram shows the marker tracking process.



Figure 6.7: The 3D chair overlaying the marker.

For a successful and efficient tracking process, a camera calibration procedure is necessary. Since this application uses a camera for tracking, video see-through calibration needs to be established. In the current version of ARToolKit, default camera properties are contained in the camera parameter file `camera_para.dat`, which is read each time an application is started. The default parameters should be sufficient for a wide range of different cameras. The specific camera parameters can be obtained by a relatively simple camera calibration technique. ARToolKit provides two calibration approaches; the first one uses two patterns containing an equally spaced 6x4 dot pattern and a pattern made up from a grid of lines and the second one uses the dot pattern only (Figure 6.8) (Lamb 2013).

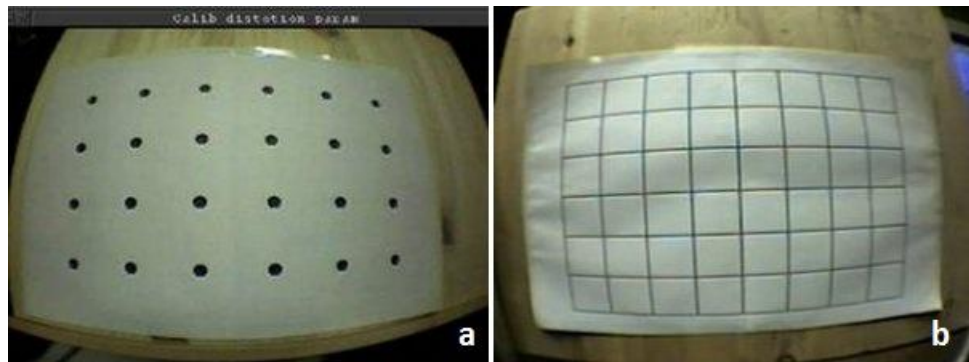


Figure 6.8: ARToolKit calibration patterns a) dot pattern, b) grid of lines

In the markerless technique, there is no need for using black squares against a white background. Any detailed object in the user's environment can be used for tracking. Using detailed objects is not a necessity but using a plain object makes tracking difficult. Choosing suitable features for recognizing and tracking the target is really important. The technique for a successful markerless tracking differs from the one used for marker-based tracking.

The first step of the markerless tracking is the same as described previously (See Figure 6.4). A captured video frame is needed to find suitable feature points for tracking. After capturing the image, a suitable method to extract feature points is required. At this point, SIFT approach is used for detecting and extracting local features. SIFT is one of the most robust and the widely used image matching algorithms based on local features. It is suitable for fast and exact matching in a mass feature database. As it is described before, SIFT feature points are reasonably invariant to changes in illumination, rotation, scaling, image noise and

small changes in viewpoint. A SIFT detector is written in C# by using Emgu CV. Figure 6.9 shows a sample result of using SIFT algorithm which is also used in this study (The image on the left in this figure was produced from a webcam in a darker environment and was blurred. A section surrounding the logo was chosen from that picture as a model image. The image on the right was derived from the same webcam. SIFT algorithm seems successful even some of the model image is covert with another object.).

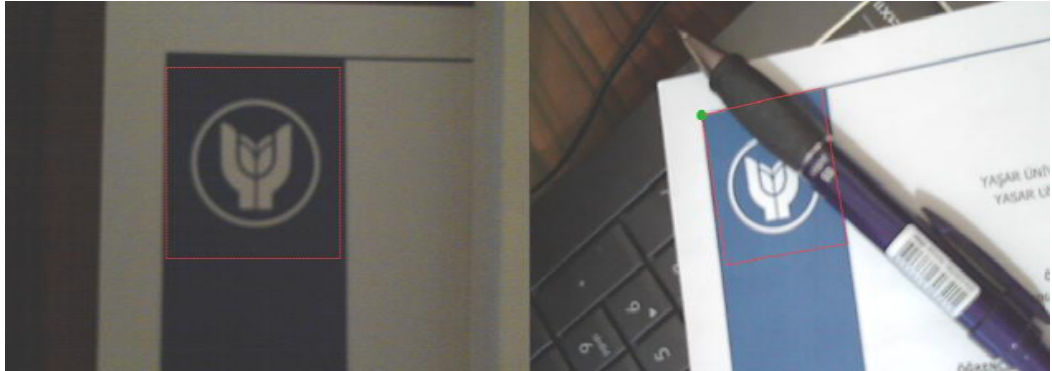


Figure 6.9: A sample result of using SIFT algorithm.

In the previous step, SIFT algorithm was applied to the captured video frame. As a result of this, useless features are discarded and useful feature points are collected. By this way, we can find the corners of the model image by using these feature points, so we can use them to do pose estimation. In this study, Coplanar POSIT algorithm in AForge.NET framework is applied to achieve this process (It should be noted that Coplanar POSIT algorithm requires image coordinates of at least 4 feature points). We need to define model of the selected object and then define the feature points which are already collected by SIFT algorithm. In addition, an effective focal length is also needed (the width of the captured image can be taken as a good approximation of it). When SIFT algorithm is applied with the feature points, model points and the focal length, the rotation matrix and translation vector are obtained as the results. These results are applied to a virtual model. Finally, the model on the real world can be overlaid by the 3D virtual model.

7. EXPERIMENTAL RESULTS

Several experiments consisting of indoor testing environments were carried out to evaluate the proposed system. The application was run on a laptop computer which has the following specifications:

- CPU : Intel(R) Core(TM) i7-2630QM @ 2.00 GHz
- Memory : 4 GB DDR3 1333 MHz SDRAM
- Graphic : Intel HD 3000 + NVIDIA® GeForce GT 550M with 2GB DDR3 VRAM
- HDD : 640GB 5400rpm
- Webcam : Integrated 2.0 Mega Pixel web camera (a Logitech C270 HD 720P webcam was also used)
- OS : Windows 7 Home Premium 64bit

The proposed system was successfully tested in different lighting conditions and sufficient results were observed.

7.1. Results from Marker-Based Tracking

A well-known problem of vision tracking is the control of lighting. Among normal indoor lighting conditions, the markers in the real-time image were tracked successfully and the furniture models were placed into the scene correctly. In some conditions, the images were blurry because of the poor quality of the integrated webcam. The algorithm provided robustness with these blurry images (See Figure 6.2). But in difficult states, such as in poor or extremely bright lighting environments, the main difficulty was finding the markers. The bright light reflected from the marker and the shadows of other objects made it hard to find the black marker frame and it affected the tracking process in a negative manner. The continuity of the representation of the furniture model couldn't be satisfied.

When one marker overlaps with another marker, the overlapped one is not detected (Figure 7.1). Also the markers need to be visible within a specific angle of orientation (Figure 7.2).

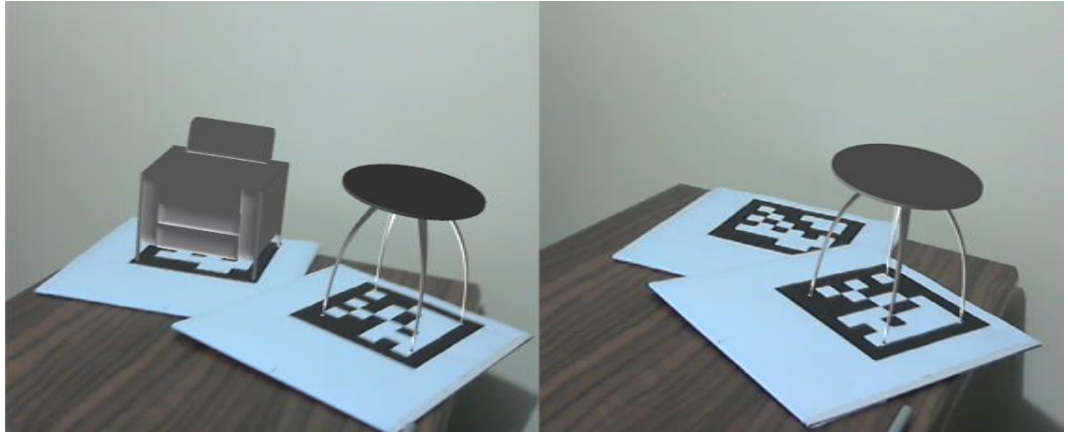


Figure 7.1: The overlapped marker is not detected.



Figure 7.2: The marker cannot be found because of the camera angle.

7.2. Results from Markerless Tracking

The proposed system was experimented with the objects which have different textures. As long as sufficient number of key points is obtained, the selected area can be tracked (Figure 7.3). It was also observed that the change in the number of the key points found on the texture affected the processing time (Figure 7.4). As the number of key points increases, the processing time also increases. Similar key points could also be found in the whole image and it affects the region of interest (similar colored objects, changing lighting conditions, background etc.). This also affects the position and the rotation of the furniture model which overlays the real object.



Figure 7.3: A sample result of SIFT tracking. The selected area was found on the real-time image in average 78ms.



Figure 7.4: A sample result of SIFT tracking. The selected pattern was found on the real-time image in average 117ms.

The similar lighting problem which was observed in the marker-based tracking system also occurred in these experiments. Bright light and shadows affect the number of possible key points on the object. Unlike the marker-based tracking, partial disruption on the object doesn't affect the continuity, because some other key points still could be found.

8. DISCUSSION

In this thesis, Augmented Reality is explained by providing its historical development. Then, AR application areas are introduced with the research done in this field. Afterwards, the technical differences and similarities between Augmented Reality and Virtual Reality are discussed and the issues of AR are mentioned.

In succession, the technologies (WPF, ARToolKit, Open CV, Emgu CV, AForge.NET framework) used while creating the AR home decorating application, POSIT algorithm and SIFT algorithm are described.

It has been shown that SIFT features are reasonably invariant to image scaling, translation, rotation and are partially invariant to illumination changes. These features can be used for matching objects or object recognition. POSIT algorithm works with the points which are not coplanar, but this becomes a limitation which does not allow its usage in most augmented reality applications. Because most augmented reality applications are based on square shaped markers. At this point, Coplanar POSIT algorithm is used. Coplanar POSIT algorithm estimates the pose of an object with coplanar points which are suitable for AR.

A solution for tracking objects and estimating their positions has been proposed in this thesis. The solution integrates the outcomes of SIFT algorithm and Coplanar POSIT algorithm. This approach has been used in the application successfully by obtaining some satisfying test results.

BIBLIOGRAPHY

- Azuma, Ronald T.** "A survey of augmented reality." *Presence: Teleoperators and Virtual Environments*(1054-7460) 6, no. 4 (1997): 355-385.
- Bichlmeier, Christoph, Felix Wimmer, Sandro Michael Heining, and Nassir Navab.** "Contextual anatomic mimesis hybrid in-situ visualization method for improving multi-sensory depth perception in medical augmented reality." *Proceedings of the 2007 6th IEEE and ACM International Symposium on Mixed and Augmented Reality*. 2007. 1-10.
- Blum, Jeffrey R., Daniel G. Greencorn, and Jeremy R. Cooperstock.** "Smartphone sensor reliability for augmented reality applications." *9th International Conference, MobiQuitous 2012. Revised Selected Papers*. Beijing, China, 2012. 127-138.
- Carmigniani, Julie, and Borko Furht.** "Augmented reality: An overview." In *Handbook of Augmented Reality*, 3-46. 2011.
- Dementhon, Daniel F., and Larry S. Davis.** "Model-based object pose in 25 lines of code." *International Journal of Computer Vision*. 1995. 123-141.
- EmguCV.** *Emgu CV: OpenCV in .NET*. 2013. http://www.emgu.com/wiki/index.php/Main_Page (accessed June 5, 2013).
- Geekologie.** *Cool: Augmented Reality Advertisements*. 12 19, 2008. <http://geekologie.com/2008/12/cool-augmented-reality-adverti.php> (accessed June 2, 2013).
- Google.** *Google Glass*. 2013. <http://www.google.com/glass/start/what-it-does/> (accessed June 8, 2013).
- . *Google Trends*. 2013. <http://www.google.com/trends/explore?q=augmented+reality#q=augmented%20reality%2C%20virtual%20reality&cmpt=q> (accessed June 8, 2013).
- Hammad, Amin.** "Mobile augmented reality." Chap. 9 in *Telegeoinformatics: Location Based Computing and Services*, edited by Hassan A. Karimi. London, 2004.
- Juan, M. Carmen, Mariano Alcañiz, Carlos Monserrat, Cristina Botella, Rosa Maria Baños, and Belén Guerrero.** "Using Augmented Reality to Treat Phobias." *IEEE Computer Graphics and Applications*, 2005: 31-37.

Julier, Simon, Yohan Baillot, Marco Lanzagorta, Dennis Brown, ve Lawrence Rosenblum. «BARS: Battlefield augmented reality system.» *NATO Symposium on Information Processing Techniques for Military Systems*. 2000. 9-11.

Kipper, Gregory, and Joseph Rampolla. *Augmented Reality: An Emerging Technologies Guide to AR*. Syngress, 2012.

Kirillov, Andrew. *AForge.NET open source framework*. 2007. <http://www.codeproject.com/Articles/16859/AForge-NET-open-source-framework> (accessed June 5, 2013).

Klopfer, Eric, Judy Perry, Kurt Squire, Ming-Fong Jan, and Constance Steinkuehler. "Mystery at the museum: a collaborative game for museum education." *Proceedings of the 2005 conference on Computer support for collaborative learning: learning 2005: the next 10 years!*, 2005: 316-320.

Krevelen, D.W.F. van, and Ronald Poelman. *A survey of augmented reality technologies, applications and limitations*. Vol. 9, chap. 2 in *The International Journal of Virtual Reality*, 1-20. 2010.

Krueger, Myron W. "VIDEOPLACE - an artificial reality." *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. San Francisco, California, USA, 1985. 35-40.

Lamb, Philip. *ARToolkit*. 2013. <http://www.hitl.washington.edu/artoolkit/> (accessed June 3, 2013).

Lowe, David G. "Object recognition from local scale-invariant features." *Proceedings of the International Conference on Computer Vision*. 1999. 1150-1157.

Martins, Pedro, and Jorge Batista. "Monocular head pose estimation." *Proceedings of the 5th international conference on Image Analysis and Recognition*. 2008. 357-368.

MarvelComics. *Marvel AR*. 07 11, 2013. https://play.google.com/store/apps/details?id=com.aurasma.skinned.marvel_ar (accessed July 19, 2013).

Microsoft. *Introduction to WPF*. 2013. <http://msdn.microsoft.com/en-us/library/aa970268.aspx> (accessed June 9, 2013).

Milgram, Paul, and Fumio Kishino. "Taxonomy of mixed reality visual displays." *IEICE Transactions on Information Systems*, 12 12, 1994: 1321-1329.

Möhring, Mathias, Christian Lessig, and Oliver Bimber. "Video see-through AR on consumer cell-phones." *Proceedings of the 3th IEEE/ACM international Symposium on Mixed and Augmented Reality*. 2004. 252-253.

OpenCV. *About Open CV*. 2013. <http://opencv.org/about.html> (accessed June 04, 2013).

Parrot. *AR.Pursuit*. 2010. <https://itunes.apple.com/gb/app/ar.pursuit/id398459463?mt=8> (accessed June 2, 2013).

Ramteke, Ashish. *WPF Architecture – An overview of Windows Presentation Foundation (WPF) Architecture*. 2012. <http://a1ashiish-csharp.blogspot.com/2012/03/wpf-architecture-overview-of-windows.html> (accessed June 9, 2013).

Reitmayr, Gerhard, and Dieter Schmalstieg. "Mobile collaborative augmented reality." *Proceedings of the International Symposium on Augmented Reality*. 2001. 114-123.

Rekimoto, Jun. "Matrix: A Realtime Object Identification and Registration Method for Augmented Reality." *Proceedings of the Third Asian Pacific Computer and Human Interaction*. IEEE Computer Society, 1998. 63-68.

Rosenberg, Louis B. «Virtual fixtures: Perceptual tools for telerobotic manipulation.» *Virtual Reality Annual International Symposium*. Seattle, WA, USA, 1993. 76-82.

Souppouris, Aaron. *IKEA app projects virtual furniture into your living room*. 2013. <http://www.theverge.com/2013/8/9/4604816/ikea-catalog-augmented-reality-2014> (accessed September 2013).

Sutherland, Ivan Edward. "A head-mounted three dimensional display." In *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, 757--764. San Francisco, California, USA, 1968.

Thomas, Bruce H., et al. «ARQuake: An outdoor/indoor augmented reality first person application.» *Proceedings of the Fourth International Symposium on Wearable Computers*. Atlanta, Ga, USA, 2000. 139-146.

ViewAR. *ViewAR - World Wide Augmented Reality Catalog*. 2013. <http://www.viewar.com/> (accessed September 2013).

Viutek. *uDecore - Home Decoration with Augmented Reality*. 2013. www.viutek.com/udecore.html (accessed September 2013).

Wikitude. *Wikitude Navigation.* 2010.
<http://www.wikitude.com/showcase/wikitude-navigation/> (accessed June 3, 2013).

Zhou, Feng, Henry Been-Lirn Duh, and Mark Billinghurst. "Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR." *Proceedings of the 7th IEEE/ACM International Symposium on Mixed and Augmented Reality.* 2008. 193-202.