DEVELOPMENT OF VIRTUAL FITTING ROOM SYSTEM

THESIS

Submitted to International Program
Faculty of Industrial Technology in Partial Fulfillment of
the Requirements for the degree of Sarjana Teknik Industri at
Universitas Islam Indonesia

By

Name : Rahito

Student Number : 07 522 115

INTERNATIONAL PROGRAM

DEPARTMENT OF INDUSTRIAL ENGINEERING
FACULTY OF INDUSTRIAL TECHNOLOGY
UNIVERSITAS ISLAM INDONESIA
YOGYAKARTA
2012

UNIVERSITAS ISLAM INDONESIA

UNIVERSITAS ISLAM INDONESIA

DECLARATION LETTER

Demi Allah, I declare this research was conduct by myself except the citation. If someday this declaration letter to be proved plagiarism, Universitas Islam Indonesia had right to revoke to its confession.

Yogyakarta, April 2012

Rahito
07 522 190

DEDICATION PAGE

I dedicate my final project to my parents, my big family, someone beside me and everyone who always support me until this time.

MOTTO

فَبِأَيِّ آلَاءِ رَبِّكُمَا تُكَذِّبَانِ

"Then which of the Blessings of your Lord will you both (jinns and men) deny?"

(QS. Ar-Rahman:13)

فَبِمَا رَحْمَةٍ مِّنَ ٱللَّهِ لِنتَ لَهُمْ ۖ وَلَوْ كُنتَ فَظًّا غَلِيظَ ٱلْقَلْبِ لَٱنفَضُّوا۟ مِنْ
حَوْلِكَ ۖ فَٱعْفُ عَنْهُمْ وَٱسْتَغْفِرْ لَهُمْ وَشَاوِرْهُمْ فِى ٱلْأَمْرِ ۖ فَإِذَا عَزَمْتَ
فَتَوَكَّلْ عَلَى ٱللَّهِ ۚ إِنَّ ٱللَّهَ يُحِبُّ ٱلْمُتَوَكِّلِينَ ﴿١٥٩﴾

"And by the Mercy of Allah, you dealt with them gently. And had you been severe and harsh-hearted, they would have broken away from about you; so pass over (their faults), and ask (Allah's) Forgiveness for them; and consult them in the affairs. Then when you have taken a decision, put your trust in Allah, certainly, Allah loves those who put their trust (in Him)"

(QS. Ali `Imran : 159)

ACKOWLEDGMENT

Assalamualaikum Wr. Wb.

This final research report was made as part of requirement to reach "Sarjana Teknik" degree Industrial Engineering, Universitas Islam Indonesia. I would to thank Allah SWT because I have been blessed to finish this final research in time. Many parties contribute to this final research; it is a pleasure to thank those who made this final research possible. I owe my deepest gratitude to:

1. My parents who always pray for me, my big family and someone beside me.
2. Muhammad Ridwan Andi Purnomo., ST., M.Sc., Ph.D for the supervision.
3. Secretary of International Program of Industrial Engineering Department, Universitas Islam Indonesia
4. For all my Brain Futura company patner
5. For all my friends of International Program of Industrial Engineering
6. For all my family in SIMAN laboratory

Wassalamu 'alaikum Wr. Wb.

Yogyakarta, April 2012

Author

Abstract

Servicing customer is one of the important things that should be considered by business subject. Technology will become one of the tools to improve customer services. In conventional online apparel shop, there is some missing in providing the fitting room. Usually an apparel shop just provide online catalog and allow logged user to browse the catalog without any chance to try some shirts. This research focuses on the development of Virtual Fitting Room (VFR) for Cloud Computing (CC) based online apparel shop. In the developed VFR, a fast and efficient motion detection algorithm is developed as interactive point of the VFR with the user. Based on the conducted experiment, it shows that the developed VFR system is able to work like real fitting room and also able to be embedded inside a CC based online apparel shop system.
Keyword: virtual fitting room, motion detection, image overlay, flash action script.

TABLE OF CONTENT

COVER i

APPROVAL PAGE OF SUPERVISOR ii

APPROVAL PAGE OF EXAMINER COMMITEE iii

DECLARATION LETTER iv

DEDICATION v

MOTTO vi

ACKNOWLEDMENT vii

ABSTRACT viii

TABLE OF CONTENT x

LIST OF FIGURE xii

CHAPTER I INTRODUCTION 1

1.1 Background 1

1.2 Problem Formulation 2

1.3 Problem Boundaries 2

1.4 Research Objective 2

1.5 Research Benefit 2

1.6 Thesis Structure 3

CHAPTER II LITERATUR REVIEW 4

2.1 Theoritical Review 4

2.1.1 Virtual Fitting Room 4

2.1.1.1 VFR Application 4

2.1.2 Motion Detection....8
2.1.2.1 Sensor....8
2.1.2.2 Motion Detection Algorithm....9
2.1.3 E-commerce....10
2.1.3.1 Tipe of E-commerce....10
2.1.4 Augmented Reality....13
2.1.4.1 Category of Augmented Reality....15
2.2 Related Research....16
CHAPTER III RESEARCH METHOD....19
3.1 Problem Identification....19
3.2 Literatur Review....19
3.3 System Development....19
3.3.1 Software Platform Analysis and Selection....19
3.3.2 Development of Camera Image Capture Module....21
3.3.3 Development of Motion Detection Module....21
3.3.4 Development of Image Overlay Module....21
3.4 System Integration....21
3.5 System Testing....21
3.6 Discussion....21
3.6 Conclussion and Suggestion....22
CHAPTER IV DATA COLLECTING AND PROCESSING....23
4.1 Software Platform and Selection....23
4.1.1 *Microsoft Silverlight*....23
4.1.2 Adobe Flash....24

4.1.3 Analysis and Selection 25

4.2 Development of Camera Image Capture Module 26

4.3 Development of Motion Detection Module 27

4.4 Development of Image Overlay Module 29

4.5 Modules Integration 30

4.6 System Integration 31

CHAPTER V SYSTEM TESTING AND DISCUSSION 32

5.1 System Testing 32

5.2 Discussion 35

CHAPTER VI CONCLUSION AND SUGGESTION 36

6.1 Conclusion 36

6.2 Sugestion 36

References

Appendices

LIST OF FIGURE

Figure 2.1 Dummy Example of Robotic VFR 6
Figure 2.2 Ilustration of VFR 7
Figure 2.3 Motion Detection Algorithms 9
Figure 2.4 Scheme of Virtual Continoum 14
Figure 2.5 Basic Principle of AR 16
Figure 3.1 Research Flowchart 20
Figure 4.1 Process of Video Compression 25
Figure 4.2 Pseudo cCode of Camera Capture Module 26
Figure 4.3 Camera Capture Image 27
Figure 4.4 The Process of Substraction 28
Figure 4.5 Thresholding Algoriths to Detect Motion 29
Figure 4.6 JPG image and PNG image 30
Figure 4.7 Pseudo Code of Image Overlay Module 30
Figure 4.8 User Interface 32
Figure 5.1 Virtual Button Position 33
Figure 5.2 Pseudo Code to Manage The Display of Loaded Shirt Image 33
Figure 5.3 The VFR System When Used by a User 34

CHAPTER I

INTRODUCTION

1.1 Background

In conventional apparel shop, customers need to come to the shop in order to buy apparel products. When entering the apparel shop, the customers will have a look to apparel collection which is displayed in some medium. Then, the customers will choose the apparel they like and will go to fitting room in order to fit the chosen cloth with their body. From the producer point of view, the above business model needs huge capital to build all of the infra structures, and unfortunately, has limited market. The apparel product just shown to the customers who visited the shop. Another disadvantage of the conventional apparel business model is cloth fitting process that usually takes a long time.

On the other hand, information technology (IT) has been growing fast. Almost all of business today supported by the development of information technology (IT), including apparel industry. There are so many technologies to develop online apparel shop that able to help producers to run their business 24 hours per day, 7 days per week. Not only that, that, the apparel product also can be shown and offered to wider market. The problem that still exists in the online apparel shop is there is no feature as fitting room, customers who come to the web page are able to choose the apparel products but they cannot fit it to their body. It is noted that virtual fitting room (VFR) is needed to be developed and embedded inside an online apparel shop.

In our country, especially in Jogjakarta, batik industry received a great attention from the producer and also the government. Usually, batik industry is established as

small and medium enterprise (SME). Currently, the focus of SME is more on increasing market and efficient production process. Because using this technology, apparel shop can put it inside their web so that there will be have more customer, not only inside their region, also apparel shop can put their extra shop in some place like airport or train station, of course with VFR tools there. But the use of technology, especially the IT, is not the main focus of SME and it leads to no capital allocation for the IT infra structure. Fortunately, currently there is a technology, called cloud computing (CC) that enables the owner of a SME to use IT with minimum cost. The owners of SME are allowed to rent IT infra structures including several application software to run their business. They don't need to provide IT infra structures and buy expensive software license as support system to success their business.

1.2 Problem Formulation

Based on the background above, the problem to be solved is how to develop a VFR that able to be embedded inside an online apparel shop under CC system.

1.3 Problem Boundaries

To focus the research, following boundaries are used in this research

a. The developed VFR system is 2D system and testing conducted by researcher.
b. The developed VFR is not final system. It is just a prototype but able to be developed further.
c. Development cost is ignored.

1.4 Research Objective

The objective of this research is to develop a VFR that able to be embedded inside an online apparel shop under CC system.

1.5 Research Benefits

Main benefit of this research is to enrich the knowledge on online business system. It includes the benefit of using CC system and the application of new technology, especially on Augmented Reality (AR) for online business system.

1.6 Thesis Structure

In order to get a scientifically-structured thesis report, this thesis will be continued by containing:

CHAPTER II LITERATURE REVIEW

This chapter is the backbone to determine the current study from the previous related ones. It contains information about the result of related previous studies and supporting literatures underlying the research.

CHAPTER III RESEARCH METHODOLOGY

This chapter describes in detail about the method used in this research.

CHAPTER IV DATA COLLECTING AND PROCESSING

It explains the data collecting and processing related to the case study.

CHAPTER V DISCUSSION

This chapter consists of result analysis and discussion, in conformity with the research background which pointing to the conclusion of research.

CHAPTER VI CONCLUSION AND RECOMMENDATION

Conclusion gives short and precise statements described from the previous chapters. Recommendations related to the current study in purpose of the advancement in future study are given as well. Some suggested improvements for the company might be given also in this chapter.

REFERENCES

APPENDICES

CHAPTER II

LITERATURE REVIEW

2.1 Theoretical Review

2.1.1 Virtual Fitting Room

Virtual dressing room is a relatively new concept which is becoming more available on various fashion websites. Taking form in different ways, the virtual dressing room allows the at home shopper to virtually try on dresses and other fashions online. This allows the consumer to gauge if the style and the fit are an appropriate match before adding it to the virtual shopping cart of a web store. There are currently several different online virtual dressing rooms, but brick and mortar stores are also starting to use virtual technology in the physical dressing room. In October 2010, Macy's introduced their new virtual mirror dressing room, allowing shoppers to select merchandise from a computer using computer tablet and a touch screen mirror. With password secured internet access, the shopper has the option of sharing the image of the merchandise with friends via the internet and then enlarging the clothing for a life size virtual fit.

2.1.1.1 Virtual Fitting Room Application

Some find it tiresome and time consuming to try clothes in clothing stores. It could be easier if one could see whether or not the clothes would fit without having to take off one’s own clothes to try out the new ones, and without having to wait in a long queue outside the fitting rooms. It is possible to order clothes on the internet but you never know whether the selected garments fit until you try them on at home. A size medium e.g. can differ from one brand to another and it is difficult to judge the quality of the textile by seeing it in picture on the computer. A better scenario is to see the clothes on

one's own body, but in order to save time this may now be done without having to actually put them on. The alternative to a fully virtual try on is a mixed reality where the body is real but the clothes are digital models shaped to fit the individual body. Instead focus is on selecting the clothes and seeing how it fits. Trying clothes becomes more immediate and spontaneous.

The concept of Virtual Fitting Room is to allow users to try on clothing items virtually in front of an augmented mirror, which displays the user wearing a 2D model of the chosen clothing. This way the customers can try multiple clothing items virtually and see how they fit physically and aesthetically without having to try them on. When customers have selected their items they hang them on the bar in the closet to the left of the mirror and place their mobile phone in the box to the right. To try a new item, the users just need to put the item at the yellow side of the bar.

Krista et al (2005) classified Virtual Fitting Room specification as:

1. Robotic Virtual Fitting Room

   The first robotic virtual dressing room was invented by a startup company in Estonia called Fits. Me. The robotic prototype was in development for three years and as of 2010 is capable of changing into 2000 body types. Robotic program requires various measurements in order to accurately present the shopper with a robotic body style match. Once the match is made, the consumer can virtually try on different sizes of clothes to determine the best fit. Males need to enter the measurements for their height, neck, chest, waist, arm length, and torso length in order to find their robotic counterpart. The robot is currently only programmed for the male body; although its female counterpart is scheduled to be released in April 2011. Hawes & Curtis launched the male robotic virtual dressing room in May 2010.

   Customers enter their body measurements and visually examine how different sizes of garments fit their body. Different brands size their garments differently. In fashion, the size of clothing is as important as the fit. Fits.me does

not only fit the customers' size it fits their styles. Many people prefer to wear their clothes more fitted than suggested by size charts. Fits.me allows the customers to see how fitted or generous the garment's cut is and make the choices they feel comfortable with.

Figure 2.1 will explain mannequin example of robotic virtual fitting room as a dummy trial while the system running.

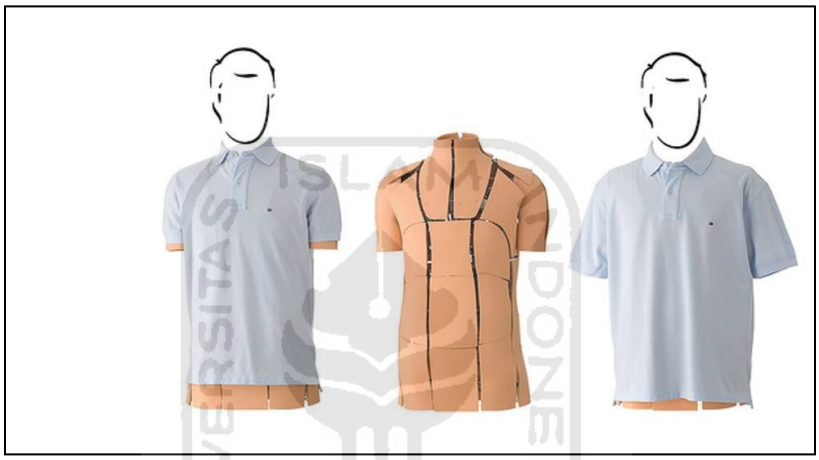

Figure 2.1 Dummy Example of Robotic Virtual Fitting Room

2. Motion Detection Virtual Fitting Room

In 2006, Eyemagnet launched an innovative method of utilizing motion detectors to create a touch free screen. Using that technology, they were able to create a virtual dressing room for Hallenstein's retail shop in New Zealand in 2007. A plasma or rear projection TV is needed to display the store's inventory. Then, acting on the concept of a motion detector, a web camera is used to place the consumer into a virtual dressing room, thereby turning the monitor into a digital mirror. Using hand motions, a consumer can wave different apparel items onto the screen to, virtually, try them on. A hand motion can add items to a virtual shopping cart that tallies the items and totals it on the screen. One more wave of the hand lets the consumer text a photo of the virtual outfit to a cell phone.

3. Web Cam Virtual Fitting Room

The first online web cam virtual dressing room was introduced by Zugara in June 2009. Tobi.com was the first online retail store to launch the Fashionista product in November 2009, which allows at home consumers to virtually try on clothes on their own beings using a web cam. In order to use the Fashionista program, an online shopper needs to print out a marker and then follow the online steps to align the actual position with the web cam so that the clothes are sized correctly on the monitor.

Zugara also released a streaming reality program called ZugSTAR which can be integrated with the Fashionista program. The streaming capability allows virtual contact between two people so that they can, in essence, shop together in a virtual world regardless of their real life locations.

Figure 2.1 will explain more about the application of Virtual Fitting Room in a general condition.

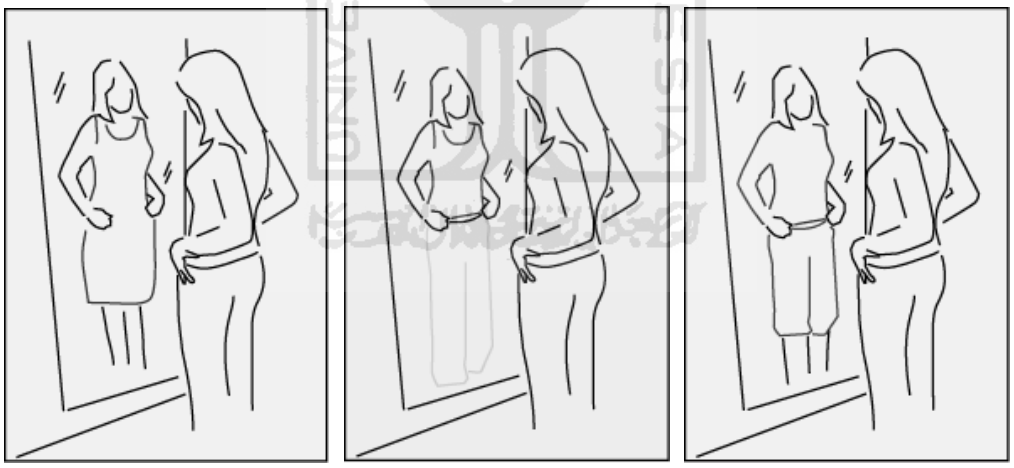

Figure 2.2 Illustration of Virtual Fitting Room

2.1.2 Motion Detection

Motion detection is a process of confirming a change in position of an object relative to its surroundings or the change in the surroundings relative to an object. This detection can be achieved by both mechanical and electronic methods. In addition to discrete, on or off motion detection, it can also consist of magnitude detection that can measure and quantify the strength or speed of this motion or the object that created it. When motion detection is accomplished by natural organisms, it is called motion perception.

Motion can be detected by: sound (acoustic sensors), opacity (optical and infrared sensors and video image processors), geomagnetism (magnetic sensors, magnetometers), reflection of transmitted energy (infrared laser radar, ultrasonic sensors, and microwave radar sensors), and electromagnetic induction (inductive-loop detectors), and vibration (triboelectric, seismic, and inertia-switch sensors). Acoustic sensors are based on: electret effect, inductive coupling, capacitive coupling, triboelectric effect, piezoelectric effect, and fiber optic transmission. Radar intrusion sensors have the lowest rate of false alarms.

An electronic motion detector contains a motion sensor that transforms the detection of motion into an electric signal. This can be achieved by measuring optical changes in the field of view. Most motion detectors can detect up to 15 – 25 meters (50–80ft). A motion detector may be connected to a burglar alarm that is used to alert the home owner or security service after it detects motion. Such a detector may also trigger a red light camera or outdoor lighting. An occupancy sensor is a motion detector that is integrated with a timing device. It senses when motion has stopped for a specified time period in order to trigger a light extinguishing signal. These devices prevent illumination of unoccupied spaces like public toilets. They are widely used for security purposes.

2.1.2.1 Sensor

There are basically four types of sensors used in motion detectors spectrum:

1. Passive infrared sensors (Passive)

Looks for body heat. No energy is emitted from the sensor.

2. Ultrasonic (active)

Sends out pulses of ultrasonic waves and measures the reflection off a moving object.

3. Microwave (active)

Sensor sends out microwave pulses and measures the reflection off a moving object. Similar with police radar gun.

4. Tomographic Detector (active)

Senses disturbances to radio waves as they travel through an area surrounded by mesh network nodes.

2.1.2.2 Motion Detection Algorithm

There were several algorithm used in order to make motion detection system. Motion detection algorithm used inside software development engine. Figure 2.3 will explain the algorithm of motion detection in action script cs5.

```
private function onEnterFrameHandler(event : Event) : void
{
    // Tell the MotionTracker to update itself
    _motionTracker.track();

    // Move the target with some easing
    _target.x += ((_motionTracker.x + _bounds.x) - _target.x) / 10;
    _target.y += ((_motionTracker.y + _bounds.y) - _target.y) / 10;

    _video.draw(_motionTracker.input);

    // If there is enough movement (see the MotionTracker's minArea property) then continue
    if ( !_motionTracker.hasMovement ) return;

    // Draw the motion bounds so we can see what the MotionTracker is doing
    _bounds.graphics.clear();
    _bounds.graphics.lineStyle(0, 0xFFFFFF);
    _bounds.graphics.drawRect(_motionTracker.motionArea.x, _motionTracker.motionArea.y,
    _motionTracker.motionArea.width, _motionTracker.motionArea.height);
    _xcoord.text = "x coord: " + _motionTracker.motionArea.x;
    _ycoord.text = "y coord: " + _motionTracker.motionArea.y;
```

Figure 2.3 Motion Detection Algorithms

2.1.3 E-commerce

E-commerce is a mechanism to trade or sale made on the internet where buyers and sellers found in cyberspace. E-Commerce will change all marketing activities and will

cut operating costs for commercial activities. Some of the processes that exist in e-commerce are as follows (January, et al, 2008):

1. Electronics presentation (Web site creation) for the products and services.
2. Direct reservations and availability of bills
3. Automation Customer accounts are safe (both account numbers or credit card)
4. Payment is made directly (online) and handling the transaction.

2.1.3.1 Type of E-commerce

E-commerce can be divided into several types with different characteristics, namely (Januari, et al, 2008):

a. Business to Business (B2B)

B2B has the following characteristics:

1. Trading partners are already known and generally have a pretty long time so this type of information transmitted can be adjusted according to the needs and beliefs. Information is only exchanged with the partner.
2. Data exchange takes place repeatedly and at regular intervals, eg every day with a data format that has been agreed that the service being used is determined. This facilitates the exchange of data for the two entities that use the same standard.
3. One of the actors can take the initiative to submit data, do not have to wait for his partner.
4. The model used is a peer-to-peer, where processing intelligence can distribute to both businesses players.

b. Business to Consumer (B2C)

B2C has the following characteristics:

1. Open to the public where information is disseminated to the public.
2. General nature of the services provided by a mechanism that can be used by the general public. For example, because the system is commonly used web service is provided using a web base.
3. Services are provided on request (on demand). Initiative and customer service providers should be prepared to respond in accordance with the request.
4. Client-server approach is often used where the consumer is taken assuming the client) uses a minimal system (web based) and process (business procedures) placed on the server side.

c. Consumer to Consumer (C2C)

C2C is a transaction in which a person selling a product or service to others. Can also be called as a customer to a customer that is, those that sell products and services to others.

Sales and purchases in the C2C auction site is very much available. Most of the auctions conducted by intermediaries, such as eBay.com. Customers can also use special sites such as buyit.com or bid2bid.com. In addition, many customers who conduct themselves as greatshop.com auction software provider to create an online auction community.

d. Consumer to Business (C2B)

Inform consumers on the type C2B requirements for a particular product or service and then the suppliers compete to provide products or services to consumers

In general there are four sections in e-commerce business processes are:

a. Information sharing

This process serves to locate and attract potential buyers as much as principle buyer is doing everything possible to find a product that needs

service tau and try to figure out the assessment or responses of others towards a product or service.

b. Most products

Ordering a product or service is done electronically. Two interacting parties should carry out a specific activity of the previous agreement, so the buying process can be done legally, properly and safely. Purchases between two business entities is usually done through a network such as EDI (Electronic Data Interchange) or extranet. There are four schools of entities that must be managed well in this business process, namely:

1. Flow of goods (product flow)
2. Flow of information (information flow)
3. Flow of money (cash flow)
4. Flow of document (document flow)

c. Distribution of products

The distribution of the product made after the transaction between the two sides completed.

d. After sales activities

At this stage the seller and the buyer do a variety of activities or communications such as:

1. Complaints on product quality
2. Questions or requests for information about other products
3. Notices will be new products being offered
4. Discussions about how to use the product properly

The target of this interaction is that the sustainable transaction business for future between the two parties based on customer satisfaction.

2.1.4 Augmented Reality

In the beginning, virtual fitting room was an application that tries to see picture of our self using any choosing clothes inside a multimedia way. Inspired by virtual game that can change any clothes to the object inside that game, people start to find a way to virtually change clothes that people used without should really change. Current condition of virtual fitting room field was already used augmented reality technology. The characteristic of augmented reality that can combine between virtual object into real object will be main reason why this application become so applicable to the virtual fitting room. Using augmented reality people can input virtual clothes, and after the marker captured by camera, virtual clothes will appear directly on the peoples body.

Some find it tiresome and time consuming to try clothes in clothing stores. It could be easier if one could see whether or not the clothes would fit without having to take off one's own clothes to try out the new ones, and without having to wait in a long queue outside the fitting rooms. It is possible to order clothes on the internet but you never know whether the selected garments fit until you try them on at home. A size medium e.g. can different from one brand to another and it is difficult to judge the quality of the textile by seeing it in picture on the computer. A better scenario is to see the clothes on one's own body, but in order to save time this may now be done without having to actually put them on. The alternative to a fully virtual try on is a mixed reality where the body is real but the clothes are digital models shaped to fit the individual body. The vision of AR virtual fitting room is to provide these qualities. AR virtual fitting room could improve the shopping experience by avoiding spending a lot of time changing clothes and waiting in line. Instead focus is on selecting the clothes and seeing how it fits. This makes AR virtual fitting room an experience based AR application. Trying clothes becomes more immediate and spontaneous. AR virtual fitting room does not eliminate the physical presence of the customer in the store. This decision about not making a substitute for selecting the clothes is made due to the fact that we find it important that the customer is able to touch, see and smell the real garment in order to make the best judgments. When the customer has selected some cloths it is easy to walk up to an AR virtual fitting room and see if the clothes fit. This does not have to bring several sizes of the same clothes. At AR virtual fitting room it can see the different sizes

and colors of the garment and it is also possible to get suggestions for which cloths that would go with the selected. This why the customer is not dependent on the sales associate finding other sizes etc., moreover there is less clothes to put back on the hangers when the customer has left the fitting room.

Based on Azuma (1997) augmented reality can be defined as a multimedia technology that mixed between virtual object into reality. Using augmented reality user are possible to add a virtual object that already made before into reality word using computer programming. Azuma et.al (1999) also mentions that ideally, it appears to the user as if the virtual 3-D objects actually exist in the real environment. One of the key requirements for accomplishing this illusion is a tracking system that accurately measures the position and the orientation of the observer's location in space. Technically the application work with library as a set of data, and also virtual preparation and integrated development environment to build a computer programming. Figure 2.1 will visualize more about AR.

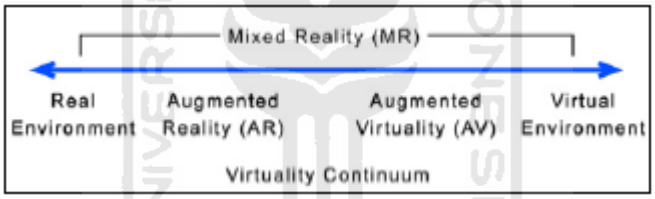

Figure 2.4 schema of a virtual continuum. From (Milgram and Kishino,1994).

After accomplish with the preparation data and also computer programming, this application need to be captured a kind of marker in order to be placed for virtual object appear into reality one. The marker can be various types, it can be marker with specific pattern, face, or a movement. After this marker already embed into the program, and then if the camera capture reality object and also the marker, it will be directly appear a virtual object on the marker.

2.1.4.1 Category of Augmented Reality

Basically the display of augmented reality itself will be separrated between 2. The different is base on the way to show the virtual object toward reality. T.P. (Caudell and Mizell, 1992) mention that those 2 categories are:

1. See trough AR display (ST-AR display)

Most commonly display augmentation is achieved by using mirrors to superimpose computer generated graphics optically onto directly viewed real-world scenes. Such displays are already a mature technology in some (mostly military) aviation systems, as either panel-mounted or head-mounted displays (HMD's), but are currently finding new applications as a Virtual Reality (VR) related technology, especially in manufacturing and maintenance environments. Several research and development issues have accompanied the advent of optical see-through (ST) displays. These include the need for accurate and precise, low latency body and head tracking, accurate and precise calibration and viewpoint matching, adequate field of view, and the requirement for a snug (no-slip) but comfortable and preferably head-mount.

Other issues which present themselves are more perceptual in nature, including the conflicting effects of occlusion of apparently overlapping objects and other ambiguities introduced by a variety of factors which define the interactions between computer generated images and real object images. Perceptual issues become even more challenging when ST-AR systems are constructed to permit computer augmentation to be presented stereoscopically. Some of these technological difficulties can be partially alleviated by replacing the optical ST with a conformal video-based HMD, thereby creating what is known as "video see-through". Such displays present certain advantages, both technological and perceptual, even as new issues arise from the need to create a camera system whose effective viewpoint is identical to that of the observer's own eyes.

2. Monitor Based AR Display

Basically concept of monitor based AR display is the opposite of HMD display where the user can directly see the virtual object with it. Monitor based display will show virtual object display on the screen of the monitor. Means that the user need to have a webcam tools in order to capture the real picture as field of Augmented Reality. When the user captures the reality object using web camera, then it will appear the virtual one on the screen.

The basic principle of AR is shown in Figure 2.5.

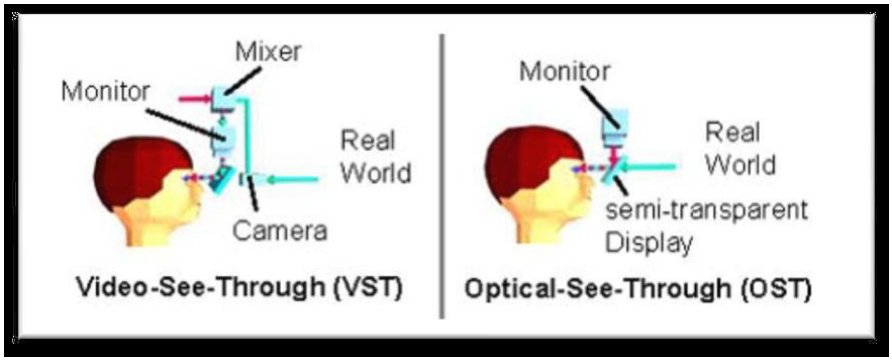

Figure 2.5 Basic Principle of AR

2.2 Related Research

The previous researches that related with this research are:

Leebmann (2002) analyze the calibration of see-through head-mounted displays (STHMD) for Augmented Reality. An Augmented Reality system (ARS) is based on several components that are affected by stochastic and random errors. One important component is the tracking system. An overview of different calibration approaches for optical see through displays is given and a quantitative comparison of the different models is made based on the derived accuracy information.

Dangelmaier et al (2004) research the virtual and AR Support for Discrete Manufacturing System Simulation. They try to find out a solution for complex problem inside manufacturing, where the dynamic of innovation always increase and product life cycle becomes shorter. They describe the architecture of a system which uses the technologies of augmented and virtual reality to support the planning process of complex manufacturing systems.

Yuan et al (2005) propose a generalized registration method using projective reconstruction technique in computer vision. The proposed method is simple, as only four points need to be specified at the embedding stage, and the virtual object can then be easily augmented onto a real scene from a video sequence.

Chen et al (2007) develop a system that enable car production planner get visualize of their prototype using AR. The updated car image is constructed on the fly on a PC server and sent back to the HMD and the PDA. The synchronicity of viewing the virtual prototype of a car, and controlling the prototype through PDA allows the designers to better evaluate the car design in full-scale.

Portalés et al (2009) develop an Augmented Reality Photogrammetric system which synergy to visualize physical and virtual city environments. The augmented environment that is presented herein requires for visualization a see-through video head mounted display (HMD), whereas user's movement navigation is achieved in the real world with the help of an inertial navigation sensor. They enable user's positioning and orientation in real time is achieved after combining an inertial sensor and a vision-based tracking technique. Which means the user can visualize building model in the real world to help user make a decision of their future job toward the buildings.

Juana and Joele (2011) compare an invisible marker versus a marker augmented reality system for the treatment of phobia towards small animals. AR system used to display small animals on the marker. With ask 24 correspondences they get a result that Moreover, 83.33% of the participants reported that they did not have the same sensations/surprise using the two systems, and they scored the advantage of using the invisible marker-tracking system.

2.3 Chapter Conclusion

Based on the previous researches, the research of AR in Virtual Fitting Room does not study yet. In this research, an image of cloths will be overlaid on the user's body. This system will be jointed with Cloud Computing system in order to launch to the web. This system will be applied in the batik's shop.

CHAPTER III

RESEARCH METHODOLOGY

This chapter discusses about the research methodology. The steps of this research will be illustrated in the research flowchart as shown in Figure 3.1. All of the step will be explained in the following sub sections.

3.1 Problem Identification

The first step of this research is identifying the problem in the apparel company. Sometimes the problem cannot be identified based on what really happened in the company. It will become suggestions in order to improve the performance of the business.

3.2 Literatures Review

In this step, several previous researches will be reviewed in order to discover the novelty of the proposed research. Basic theory as supporting knowledge of the proposed research will be presented as well.

3.3 System Development

3.3.1 Software Platform Analysis and Selection

In this step, several software platforms to develop VFR will be analyzed and selected. The analysis and selection will be conducted based on the ability to works online. Since the VFR system will be integrated with existing online apparel shop then compatibility level with the existing online apparel shop will be very important factor and must be considered as well in the VFR system development.

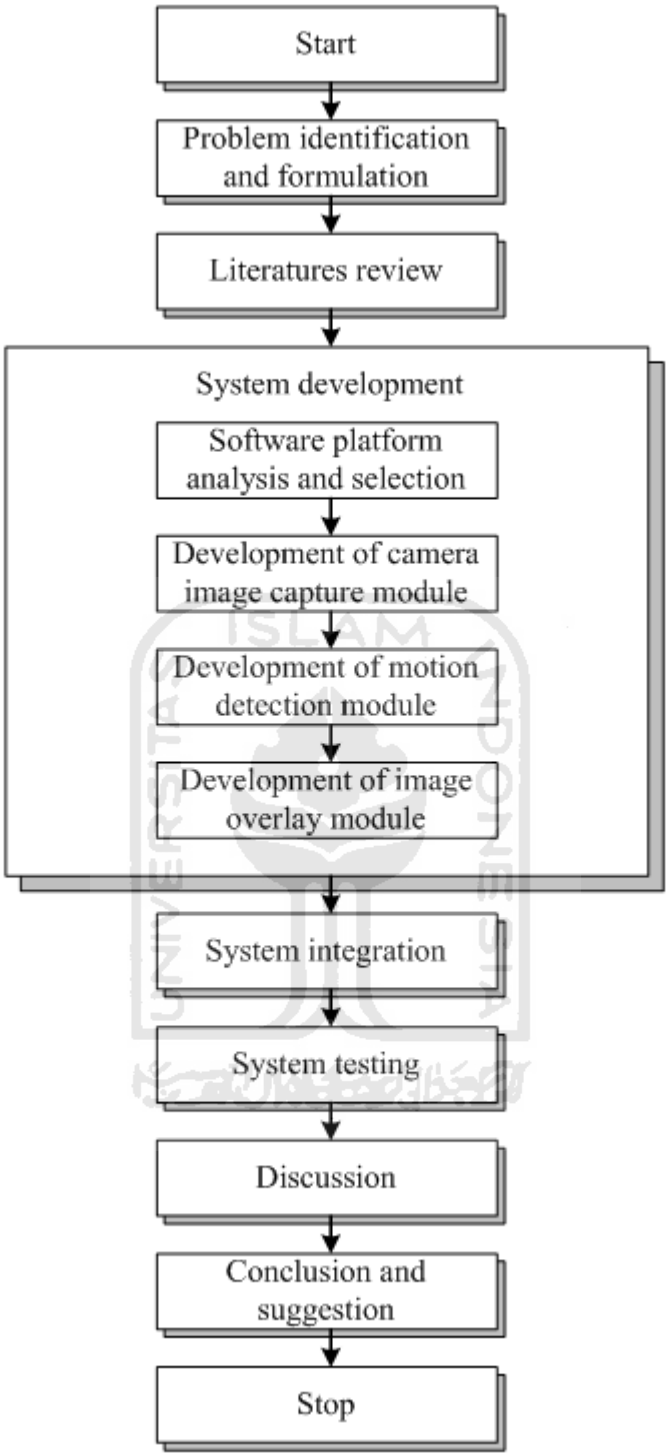

Figure 3.1 Research Flowchart

3.3.2 Development of Camera Image Capture Module

In this step, a camera image capture module will be developed. Since the proposed VFR system will be online system, then the image will be captured from webcam that today almost all of personal computer, such as laptop, already equipped with it.

3.3.3 Development of Motion Detection Module

In this step, a motion detection module will be developed. This module will be embedded behind virtual buttons in the VFR system. The virtual button is a feature to enable the user to change batik cloth image to be fitted with their body.

3.3.4 Development of Image Overlay Module

In this step, an image overlay module will be developed to augment batik image object on the camera streaming image. It will create effect that enables user to imagine they are wearing batik cloth and see them self in the mirror.

3.4 System Integration

There are two steps in the system integration phase. The first step is integrating image capture, motion detection and image overlay modules and the second step is integrating the VFR system with the existing online apparel shop system.

3.5 System Testing

In this step, after the VFR system is integrated with the existing online apparel shop system, then it is need to be tested. In this research, system testing will be conducted by the author without involving external customer or user.

3.6 Discussion

A critical discussion will be presented in this step. It covers several things that have not been explained yet in the previous chapter. This step to enrich the understanding of the VFR development process.

3.7 Conclusion and Suggestion

In this step, a clear conclusion will be made and also several suggestions for further research will also be presented. The suggestion part to generate another idea for further research so that the developed VFR system can be improved towards perfect system.

CHAPTER IV

SYSTEM DEVELOPMENT

This chapter explains the development process of VFR system. The developed VFR will then be integrated with existing online batik apparel shop system that developed under CC system. There are several steps in the VFR system development which will be explained in detail in the following sub sections.

4.1 Software Platform Analysis and Selection

The basic requirement of the VFR system to be developed is able to be integrated with existing online batik apparel shop that developed based on CC system. It means that the VFR system must be able to work online and support multi user processing. Technically, required fundamental features of the technology to be used to develop VFR system are camera image capturing and bitmap overlaying. Based on the above features, there are mainly two technologies that can be used as platform to develop the VFR system, which are Microsoft Silverlight and Adobe Flash.

4.1.1 Microsoft Silverlight

Microsoft Silverlight is an application framework for writing and running rich Internet applications, with features and purposes similar to those of Adobe Flash. The run-time environment for Silverlight is available as a plug-in for web browsers running under Microsoft Windows and Mac OS X. While early versions of Silverlight focused on streaming media, current versions support multimedia, graphics and animation, and give developers support for Common Language Infrastructure (CLI) languages and

development tools. Silverlight is also one of the two application development platforms for Windows Phone as Operating System for smart phones.

Microsoft Silverlight uses VC-1 Codec for video compression when displaying streaming video. Even though Microsoft already provide Software Development Kit (SDK) to develop it, but it still need a driver to compress the streamed data, hence the file size of the Silverlight application will become slightly bigger. Thus, the quality of the video picture is good. However, it is need to be considering that the file size of an online application must be small to deal with low internet bandwidth and low personal computer specification.

4.1.2 Adobe Flash

Adobe Flash is a multimedia platform used to add animation, video, and interactivity to web pages. Adobe Flash is frequently used for advertisements, games and flash animations for broadcast. More recently, it has been positioned as a tool for Rich Internet Applications (RIAs).

In displaying graphics, Adobe Flash manipulates vector and raster graphics to provide animation of text, drawings, and still images. It supports bidirectional streaming of audio and video, and it can capture user input via mouse, keyboard, microphone, and camera. Adobe Flash contains an object-oriented language called Action Script and supports automation via the JavaScript Flash language (JSFL). Flash content may be displayed on various computer systems and devices, using Adobe Flash Player, which is available free of charge for common web browsers, some mobile phones, and a few other electronic devices (using Flash Lite).

Flash supports **multiple video formats**. The latest codec, which is the video compression technique, is very high quality, and the bandwidth usage is reasonable. Figure 4.1 explains generally the process of video compression.

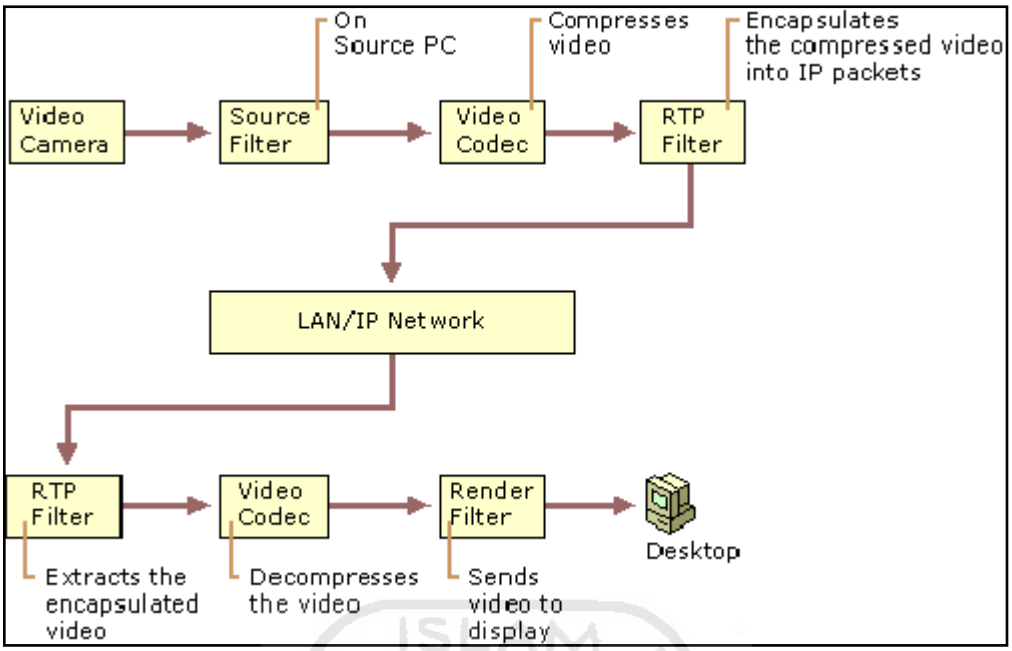

Figure 4.1 Process of video compression

(Source: http://www.smashingmagazine.com)

4.1.3 Analysis and Selection

Based on analysis above and further analysis, the comparison between Microsoft Silverlight and Adobe Flash can be concluded as shown in Table 4.1.

Table 4.1 Features comparison between Microsoft Silverlight and Adobe Flash

Feature	Microsoft Silverlight	Adobe Flash
File size		Better
Video / Audio	Better	
Accessibility		Better
Platform compatibility		Better
Supported image format		Better
Media streaming	Better	

Based on the comparison above, Adobe Flash has more features that support VFR system requirement, then it can be concluded fairly that Adobe Flash will be used to develop VFR system.

4.2 Development of Camera Image Capture Module

The basic function to be understood to capture image from client camera are conversion of streamed video data format, encoding for sending the data over internet network, and framing to manage data decoding to output handle device. All of the functions above have been supported by Adobe Flash. The required class in the camera image capture module are "`flash.media.camera`", "`flash.media.video`" and "`flash.events.*`" to instantiate image capture and format conversion, data encoding and frame management function respectively.

In Adobe Flash, the calling of required class can be carried out without considering the calling sequence. However, order of implementation of the called class must be carried out based on the requirement. Figure 4.2 shows the pseudo code of the camera image capture module and Figure 4.3 shows the result of the camera image capture module.

```
private function initTracking() : void
{
        var camW : int = 420;
        var camH : int = 320;

        // Create the camera
        var cam : Camera = Camera.getCamera();
        cam.setMode(camW, camH, stage.frameRate);

        // Create a video
        var vid : Video = new Video(camW, camH);
        vid.attachCamera(cam);

        // Create the Motion Tracker
        _motionTracker = new MotionTracker(vid);

        // We flip the input as we want a mirror image
        _motionTracker.flipInput = true;

        /*** Create a few things to help us visualise what the MotionTracker is doing... ***/

        _matrix = new ColorMatrix();
        _matrix.brightness = _motionTracker.brightness;
        _matrix.contrast = _motionTracker.contrast;

        // Display the camera input with the same filters (minus the blur) as the MotionTracker is using
        _video = new BitmapData(camW, camH, false, 0);
        _source = new Bitmap(_video);
        _source.scaleX = -1;
        _source.x = 10 + camW;
        _source.y = 10;
        _source.filters = [new ColorMatrixFilter(_matrix.toArray())];
        addChild(_source);
```

Figure 4.2 Pseudo Code of Camera Capture Module

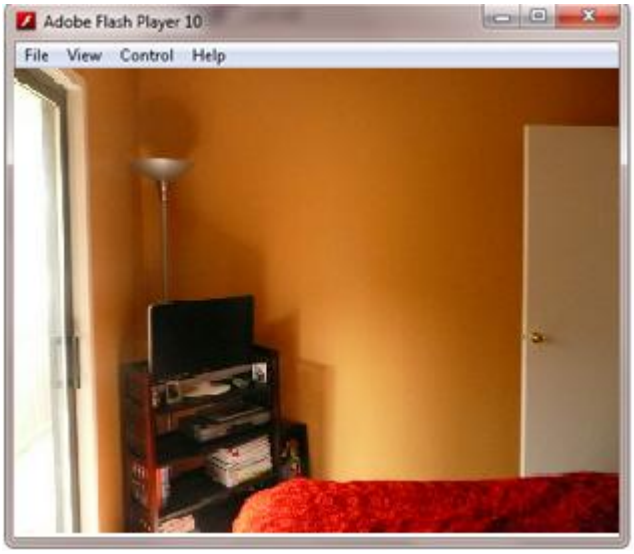

Figure 4.3 Camera Image Capture

4.3 Development of Motion Detection Module

In order to attract user awareness, the VFR is designed to be interactive. The interaction point is the user allowed to change the batik shirt image to be overlaid on his or her body image. The changing process is conducted through pressing of a virtual button which is overlaid on the upper left and right side of the camera image. The pressing process is conducted by touching virtually the virtual button. In the developed VFR system, the user is not allowed to control the VFR using external devices interface such as keyboard or mouse since it will not allow the user to move freely. For the purpose above, a motion detection module based on image processing is developed and embedded inside the VFR system.

Basically, motion detection technique is developed based on image subtraction. An image frame from the sequence image will be captured (image frame at time $t+1$) and will be subtracted with the previous image frame (image frame at t). If the resulted image is not black image, then a motion will be detected. Figure 4.4 shows the image subtraction process.

image frame at *t* image frame at *t-1* resulted image

Figure 4.4 The process of image subtraction

Basically, motion detection can be carried out based on background modeling. This technique, conceptually, will capture the background of the camera image it will be used as still image. The next step is capturing current image to be compared with the still image. The comparison is carried out based on subtraction operation. If resulted image is not totally black, then motion is detected. This background modeling is not deal with the real system due to uncontrollable user's environment. In the real condition, the VFR system will be used by users in natural environment. The still image is impossible to be modeled since static background is impossible to get as well.

In the real condition, the stability of the image is imperfect. There are several factors cause this imperfection, such as camera resolution, speed of image capture and also ambient light. These factors influence the quality of the image and it leads to unstable image subtraction result; if an image frame is subtracted with the previous image frame, the resulted image will not totally black even though there is no motion. To overcome that problem, a technique based on thresholding concept will be applied. This technique will ignore the small change in the sequence image, only big change will be considered as motion. Actually, this technique is similar with background modeling technique explained before, but in this technique, the still image is modeled dynamically, which is the image at *t-1*. Detail of thresholding algorithm is shown in Figure 4.5.

Step 1 : Capture frame image and save the pixel data at a matrix, say *ft*.

$$ft = \begin{pmatrix} p_{11} & p_{12} & p_{13} \cdots & p_{1c} \\ p_{21} & p_{22} & p_{23} \cdots & p_{2c} \\ & & \cdots & \\ p_{r1} & p_{r2} & p_{r3} \cdots & p_{rc} \end{pmatrix},$$

where c = number of columns and r = number of row

Step 2 : Capture frame image and save the pixel data at a matrix, say *ft+1*.

$$ft+1 = \begin{pmatrix} p_{11} & p_{12} & p_{13} \cdots & p_{1c} \\ p_{21} & p_{22} & p_{23} \cdots & p_{2c} \\ & & \cdots & \\ p_{r1} & p_{r2} & p_{r3} \cdots & p_{rc} \end{pmatrix},$$

where c = number of columns and r = number of row

Step 3 : Calculate image difference by formula:

$$diff = \frac{(ft+1)-(ft)}{ft} \times 100\%$$

Step 4 : Make a decision, if *diff* ≥ a threshold value then motion is detected.

Figure 4.5 Thresholding algorithms to detect motion

Based on preliminary study, the threshold value will fall in a range of 20%-30%. In this research the threshold value is set to 30%.

4.4 Development of Image Overlay Module

The image overlay module is used to augment batik shirt image inside camera image. This technique is used to make effect that user try to fit a batik shirt to his or her body. In computer filing, an image will be stored in rectangle shape. It means that batik shirt image will be overlaid in a colored rectangle. If this image overlaid on the camera image then the colored rectangle image will block the camera image as well. This will lead to unrealistic camera image. To overcome this problem, all of batik shirt image have to be converted to a portable network graphic (*.png*) file format since *png* file allows user to transparent the colored rectangle in an image file. Beside, the *png* file is dedicated as transferrable image over internet network. In the proposed VFR system, it will reduce required bandwidth of the user and also speed up the download process in user's computer. Figure 4.6 shows the comparison between *.jpg* and *.png* image file.

(a) (b)

Figure 4.6 (a). .jpg image; (b). .png image

The basic process of image overlay module is read, save and display. First, the pixel value of the image will be read and the data will be stored in an array memory. In Adobe Flash, all of the object to be managed must be handled in the form of sprite. Hence, the array memory will be embedded inside a sprite. A class is required to form the sprite. The last step is overlay the sprite on the specific position on the image camera. In here, the sprite will be the child of image camera so that it can be managed through camera image but it can be managed independently as well. Figure 4.7 shows the pseudo code of the image overlay module.

```
class bitmapoverlay
     load png file
     read pixel image and store inside an array → px
     construct sprite → sp
     embed px inside sp
     overlay sp on camera image
end class
```

Figure 4.7 Pseudo code of image overlay module

4.5 Modules Integration

After all of required modules in VFR system are developed, then they must be integrated and managed. The image camera capture module will be run as continue process. The image capture speed in this module is set to 25 frame per second (*fps*). This speed is reasonable since in the life video, the image capture speed will be in the range of 25-30fps.

The motion detection module will be executed to process the 10^{th} image frame in a sequence. It means that in a second, there are only 2 motion detection processes.

Actually this module can be ran in real time so that it can process 2 image frames in a row. However, this process is very ineffective since it will utilize high computer memory and the possibility to detect a motion in 2 image frames in a row is very low. Based on preliminary study, a motion can be detected effectively if interval between still image and current image is at least 5 frames.

The image overlay module will be executed by event. If a motion is detected by motion detection module, then there will be some flag to sign that the image overlay module must be ran. However, this technique will make the VFR system very sensitive to motion. A small unintentional motion will cause the change of batik shirt image. To overcome this problem, there is some small procedure embedded in between motion detection and image overlay module, called motion counter procedure. This procedure will count the detected motion and if the counter reach 5, then image overlay module will be executed.

4.6 System Integration

In order to provide technology with lower price, VFR system will be integrated with CC system. CC system will bridge an application provider to the user with internet network. So user only needs to have low specification of PC with internet browser software inside. In the application, user interface of VFR system will be covered inside CC system. Which are choice process of the cloths and determining which cloths will be trying. Combination of VFR and CC will provide cheap price of implementation technology in cloth shop. CC system is allowed users to use that system simultaneously and not interfere each other. This ensure to the user to do not worry when they have to implement this system. Figure 4.8 will show the user interface of VFR inside CC system.

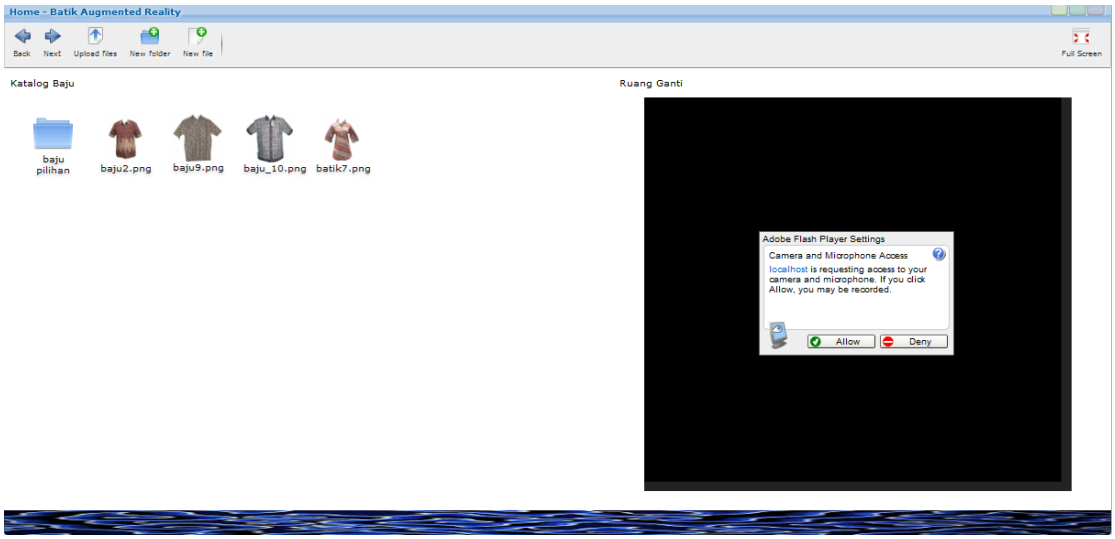

Figure 4.8 User Interface

CHAPTER V

SYSTEM TESTING AND DISCUSSION

5.1 System Testing

As mentioned in the previous chapter, in order to get real life video, the speed of image capture must be in a range of 25-30 fps. The developed camera image capture module has been tried in a personal computer with 2 Gb DDR 3 RAM, 3.6 GHz processor, and 2 Gb for VGA and it shows that this module is able to capture camera image at 25fps. However, since this VFR system is online system, then the speed of camera image capture will vary depends on personal computer of the user.

The developed motion detection module is also working well. In the implementation, this module is applied to the specific region of the virtual buttons to change the batik shirt image overlaid on the user's body. Figure 5.1 shows the position of a virtual button that overlaid on the camera image.

Figure 5.1 Virtual button position

In real apparel shop system, when a customer wants to try shirts, sometimes he or she will take more than one shirts to the fitting room. However, the shop will limit the number of shirts to 5 pieces. The developed VFR system also adopts the same idea; a customer is allowed to select more than one shirt to be tried but the maximum is only 5 pieces. The selected shirts will then be loaded in the VFR system in advance. Management to display the loaded shirt image is conducted based on the motion detection module and indexing technique. Figure 5.2 shows the pseudo code of the technique to display the loaded shirt.

```
if (_motionTracker.motionArea.x > 350 && _motionTracker.motionArea.y < 20)
{
    if (_motionTracker.motionArea.x != 0 && _motionTracker.motionArea.y != 0)
    {
        if (objectType == 10)
        {
            objectType = 1;
        }
        else
        {
            objectType++;
        }
    }
```

Figure 5.2 Pseudo code to manage the display of the loaded shirt images

Based on visual testing, it shows that the procedure above is work well to manage the display of shirt images.

Integration of the developed VFR with the virtual batik shop that developed based on CC system is also successfully carried out. In that system, the developed VFR system can be used independently by every logged user who are the owner of the virtual batik shop. The following figures also show the developed VFR system when used by a user.

Figure 5.3 The VFR systems when used by a user

5.2 Discussion

The critical thing in the developed VFR system is the motion detection module. As explained in the previous chapter, this module is developed based on thresholding

technique. Based on the empirical testing, it found that this method has disadvantages. When the environment of the user is very dynamic, for instance the VFR system is accessed in the shop where there are so many movement objects behind the user, such as customers, and then sometime the movement will trigger the virtual button.

Another disadvantage is the ambient light. When the ambient light is very unstable, sometime it will trigger the virtual button as well. Technically the problems above can be handled by increasing the threshold value of the thresholding technique. However, it is potential to generate another problem which are the sensitivity of the virtual button will decrease. When the threshold value is too high, then to trigger the virtual button, it is need very major movement, and this is not applicable. One of the ideas to overcome this problem is using hand gesture detection system and this could be basic idea for further research.

CHAPTER VI

CONCLUSION AND SUGGESTION

6.1 Conclusion

Based on the research that have been conducted, it can be concluded that the developed VFR system is work fairly good and able to be embedded inside a CC based online apparel shop. Another minor conclusion can be presented that the proposed motion detection algorithm and image overlay module that embedded inside the VFR system are also work well. They are potential to create awareness of the customer to the apparel products. It is potential to increase the sales of the online apparel shop as well.

6.2 Suggestion

Several suggestions can be proposed in order to improve the current research, such as:

a. In order to improve the stability of the motion detection module, another technique such as hand gesture detection can be tried.
b. Current researches only augment 2D shirt images. For further research, 3D model of the shirt objects can be tried to be augmented so that the user can see the shirt from 3 dimensions.

References

Azuma, R.T. 1997. A survey of augmented reality. *Teleoperators and Virtual Environments* **6***:* 355-385.

Chen, K. Chen, L. Shen, S.T. 2008. Development and comparison of a full-scale car display and communication system by applying Augmented Reality, 33–40

Dangelmaier, Wilhelm. F, Matthias. G. Jurgen. 2004. Virtual and augmented reality support for discrete manufacturing system simulation. *Computers in Industry* **56***: 371–383*.

Fernandez, B.C.A. & Sanchez, J.F. 2008. Acceptance of an augmented reality system as a visualization tool for computer-aided design classes. *Interactive Educational Multimedia An on time journal published.* Barcelona.

Fritz, F., Susperregui, A. & Linaza. M.T. 2005. Enhancing cultural tourism experiences with augmented reality technologies. *Asociación VICOMTech, San Sebastian.* Spain.

Hsieh, M.C. & Lin, H.C.K. 2010. Interaction design based on augmented reality technologies for english vocabulary learning. *Proceeding of the 18th International Conference on Computers in Education*.

Juana & Joele. 2011. *Augmented Reality for Phobia Treatment Toward Small Animal*. ISPRS, 34-48

Leebman. Johannes. 2003. A stochastic analysis of the calibration problem for Augmented

Pang, Yan. 2006. Assembly feature design in an augmented reality environment. *Assembly Automation* **26(1)**: 34-43. USA.

Portales, Christina. Lerma, L. Navarro, S. 2010. Augmented reality and photogrammetry: A synergy to visualize physical and virtual city environments, *ISPRS*, 134_142

Reality systems with see-through head-mounted displays, *ISPRS*, 400– 408

Reinhart, G. & Patron, C. 2003. Integrating augmented reality into assembly domain. *Proceedings of ClRP Design Seminar on New Tools and Product Development* 131-143.

Tang, A. 2003. Comparative effectiveness of augmented reality in object assembly. *New Techniques for Presenting Instructions and Transcripts* **5**: 1.

Yuan, M.L. Ong, S.K, Nee, A.Y.C. 2005, A generalized registration method for augmented reality systems, *Computers & Graphics* 29, 980–997

APPENDICES

1. Action Script VFR Module

```
package
{
     import uk.co.soulwire.cv.MotionTracker;

     /**
      * I'm using Grant Skinners fantastic ColorMatrix
class:
      *
http://www.gskinner.com/blog/archives/2007/12/colormat
rix_upd.html
      *
      * And Keith Peters Minimal Components:
      * http://www.bit-101.com/minimalcomps/
      */

     import com.bit101.components.Label;
     import com.bit101.components.Slider;
     import com.gskinner.geom.ColorMatrix;

    import mx.core.BitmapAsset;

     import flash.display.Bitmap;
     import flash.display.BitmapData;
     import flash.display.Shape;
     import flash.display.Sprite;
     import flash.events.Event;
     import flash.filters.ColorMatrixFilter;
     import flash.media.Camera;
     import flash.media.Video;


    import flash.display.Graphics;
    import flash.display.StageAlign;
    import flash.display.StageScaleMode;
    import flash.geom.Matrix;
    import flash.geom.Rectangle;
    import flash.media.Camera;
    import flash.text.TextField;

     /**
      * MotionTrackerDemo
      */
```

```
     [SWF(width="450", height="350",
backgroundColor="#FFFFFF", frameRate="31")]

     public class MotionTrackerDemo extends Sprite
     {

          //   ------------------------------------
-------------------------
          //   PRIVAYE MEMBERS
          //   ------------------------------------
-------------------------

          private var _motionTracker : MotionTracker;

          private var _target : Shape;
          private var _bounds : Shape;
          private var _output : Bitmap;
          private var _source : Bitmap;
          private var _video : BitmapData;
          private var _matrix : ColorMatrix;
          private var _startButton : Bitmap;
          private var masksArray:Array = new Array();
          private var masksType:Array = new Array();

          private var masksSrcbaju1:Array = new
Array();
          private var masksSrcbaju2:Array = new
Array();
          private var masksSrcbaju3:Array = new
Array();
          private var masksSrcbaju4:Array = new
Array();
          private var masksSrcbaju5:Array = new
Array();
          private var masksSrcbaju6:Array = new
Array();
          private var masksSrcbaju7:Array = new
Array();
          private var masksSrcbaju8:Array = new
Array();
          private var masksSrcbaju9:Array = new
Array();
          private var masksSrcbaju10:Array = new
Array();

          private var flagShirt01:int = 0;
          private var flagShirt02:int = 0;
```

```
            private var flagShirt03:int = 0;

            private var objectIndex:int = 0;
            private var objectType:int = 1;

            private var _blurLabel : Label = new
Label();
            private var _brightnessLabel : Label = new
Label();
            private var _contrastLabel : Label = new
Label();
            private var _minAreaLabel : Label = new
Label();
            private var _xcoord : Label = new Label();
            private var _ycoord : Label = new Label();

            private var _blurSlider : Slider = new
Slider();
            private var _brightnessSlider : Slider = new
Slider();
            private var _contrastSlider : Slider = new
Slider();
            private var _minAreaSlider : Slider = new
Slider();


            //masukin source virtual button
            [Embed(source = 'button.png')]
          private var maskSrc:Class;

            //masukin source baju nya
            [Embed(source = 'baju1.png')]
          private var maskSrcbaju1:Class;
            [Embed(source = 'baju2.png')]
          private var maskSrcbaju2:Class;
            [Embed(source = 'baju3.png')]
          private var maskSrcbaju3:Class;
            [Embed(source = 'baju4.png')]
          private var maskSrcbaju4:Class;
            [Embed(source = 'baju5.png')]
          private var maskSrcbaju5:Class;
            [Embed(source = 'baju6.png')]
          private var maskSrcbaju6:Class;
            [Embed(source = 'baju7.png')]
          private var maskSrcbaju7:Class;
            [Embed(source = 'baju8.png')]
          private var maskSrcbaju8:Class;
```

```
            [Embed(source = 'baju9.png')]
          private var maskSrcbaju9:Class;
            [Embed(source = 'baju10.png')]
          private var maskSrcbaju10:Class;



          //   ----------------------------------------
-------------------------
          //   CONSTRUCTOR
          //   ----------------------------------------
-------------------------

          public function MotionTrackerDemo()
          {
               addEventListener(Event.ADDED_TO_STAGE,
onAddedToStage);
          }

          //   ----------------------------------------
-------------------------
          //   PRIVATE METHODS
          //   ----------------------------------------
-------------------------

          private function configureUI() : void
          {
               _blurSlider.minimum = 0;
               _blurSlider.maximum = 40;

               _brightnessSlider.minimum = -100;
               _brightnessSlider.maximum = 100;

               _contrastSlider.minimum = -100;
               _contrastSlider.maximum = 200;

               _minAreaSlider.minimum = 0;
               _minAreaSlider.maximum = 50;

               _blurSlider.x = _blurLabel.x = 10;
               _blurSlider.y = stage.stageHeight - 20;
               _blurLabel.y = _blurSlider.y - 20;

               _brightnessSlider.x =
_brightnessLabel.x = _blurSlider.x + 110;
               _brightnessSlider.y = _blurSlider.y;
```

```
                _brightnessLabel.y =
_brightnessSlider.y - 20;

                _contrastSlider.x = _contrastLabel.x =
_brightnessSlider.x + 110;
                _contrastSlider.y = _blurSlider.y;
                _contrastLabel.y = _contrastSlider.y -
20;

                _minAreaSlider.x = _minAreaLabel.x =
_contrastSlider.x + 110;
                _minAreaSlider.y = _blurSlider.y;
                _minAreaLabel.y = _minAreaSlider.y -
20;

                _xcoord.x = _minAreaLabel.x + 110;
                _xcoord.y = _minAreaLabel.y;

                _ycoord.x = _xcoord.x + 110;
                _ycoord.y = _xcoord.y;

                /*
                addChild(_blurSlider);
                addChild(_blurLabel);

                addChild(_brightnessSlider);
                addChild(_brightnessLabel);

                addChild(_contrastSlider);
                addChild(_contrastLabel);

                addChild(_minAreaSlider);
                addChild(_minAreaLabel);

                addChild(_xcoord);
                addChild(_ycoord);
                */
           }

           private function initTracking() : void
           {
                var camW : int = 420;
                var camH : int = 320;

                // Create the camera
                var cam : Camera = Camera.getCamera();
```

```
                cam.setMode(camW, camH,
stage.frameRate);

                // Create a video
                var vid : Video = new Video(camW,
camH);
                vid.attachCamera(cam);

                // Create the Motion Tracker
                _motionTracker = new
MotionTracker(vid);

                // We flip the input as we want a
mirror image
                _motionTracker.flipInput = true;

                /*** Create a few things to help us
visualise what the MotionTracker is doing... ***/

                _matrix = new ColorMatrix();
                _matrix.brightness =
_motionTracker.brightness;
                _matrix.contrast =
_motionTracker.contrast;

                // Display the camera input with the
same filters (minus the blur) as the MotionTracker is
using
                _video = new BitmapData(camW, camH,
false, 0);
                _source = new Bitmap(_video);
                _source.scaleX = -1;
                _source.x = 10 + camW;
                _source.y = 10;
                _source.filters = [new
ColorMatrixFilter(_matrix.toArray())];
                addChild(_source);

                // Show the image the MotionTracker is
processing and using to track
                _output = new
Bitmap(_motionTracker.trackingImage);
                _output.x = camW + 20;
                _output.y = 10;
                //addChild(_output);
```

```
                // A shape to represent the tracking
point
                _target = new Shape();
                _target.graphics.lineStyle(0,
0xFFFFFF);
                _target.graphics.drawCircle(0, 0, 10);
                //addChild(_target);

                // A box to represent the activity area
                _bounds = new Shape();
                _bounds.x = _output.x;
                _bounds.y = _output.y;
                //addChild(_bounds);

                // Configure the UI

     _blurSlider.addEventListener(Event.CHANGE,
onComponentChanged);

     _brightnessSlider.addEventListener(Event.CHANGE,
onComponentChanged);

     _contrastSlider.addEventListener(Event.CHANGE,
onComponentChanged);

     _minAreaSlider.addEventListener(Event.CHANGE,
onComponentChanged);

                // Get going!
                addEventListener(Event.ENTER_FRAME,
onEnterFrameHandler);
          }

          private function applyFilters() : void
          {
                _blurLabel.text = "Blur: " +
Math.round(_blurSlider.value);
                _brightnessLabel.text = "Brightness: "
+ Math.round(_brightnessSlider.value);
                _contrastLabel.text = "Contrast: " +
Math.round(_contrastSlider.value);
                _minAreaLabel.text = "Min Area: " +
Math.round(_minAreaSlider.value);


                _matrix.reset();
```

```
     _matrix.adjustContrast(_contrastSlider.value);

     _matrix.adjustBrightness(_brightnessSlider.value)
;
               _source.filters = [new
ColorMatrixFilter(_matrix)];
          }

          //   ------------------------------------
-------------------------
          //   EVENT HANDLERS
          //   ------------------------------------
-------------------------

          private function onAddedToStage(event :
Event) : void
          {
               configureUI();
               initTracking();
               applyFilters();
               startdetectionHandler();

          }

          private function onEnterFrameHandler(event :
Event) : void
          {
               // Tell the MotionTracker to update
itself
               _motionTracker.track();

               // Move the target with some easing
               _target.x += ((_motionTracker.x +
_bounds.x) - _target.x) / 10;
               _target.y += ((_motionTracker.y +
_bounds.y) - _target.y) / 10;

               _video.draw(_motionTracker.input);

               // If there is enough movement (see the
MotionTracker's minArea property) then continue
               if ( !_motionTracker.hasMovement )
return;

               // Draw the motion bounds so we can see
what the MotionTracker is doing
```

```
                _bounds.graphics.clear();
                _bounds.graphics.lineStyle(0,
0xFFFFFF);

     _bounds.graphics.drawRect(_motionTracker.motionAr
ea.x, _motionTracker.motionArea.y,
_motionTracker.motionArea.width,
_motionTracker.motionArea.height);
                _xcoord.text = "x coord: " +
_motionTracker.motionArea.x;
                _ycoord.text = "y coord: " +
_motionTracker.motionArea.y;



                /*if (_motionTracker.motionArea.x < 50
&& _motionTracker.motionArea.y < 50)
                {
                     if (_motionTracker.motionArea.x !=
0 && _motionTracker.motionArea.y != 0)
                     {
                          //hasil deteksi gerakan
silakan ditulis di sini
                     }
                }*/

                //if (objectIndex != 0)
                //{
                     if (_motionTracker.motionArea.x >
350 && _motionTracker.motionArea.y < 20)
                     {
                          if
(_motionTracker.motionArea.x != 0 &&
_motionTracker.motionArea.y != 0)
                          {
                               if (objectType == 5)
                               {
                                    objectType = 1;
                               }
                               else
                               {
                                    objectType++;
                               }
                          }
                          shirt01detectionHandler();
                     }
                //}
```

```
          }

          private function onComponentChanged(event :
Event) : void
          {
               switch(event.target)
               {
                    case _blurSlider :

                         _motionTracker.blur =
_blurSlider.value;

                         break;

                    case _brightnessSlider :

                         _motionTracker.brightness =
_brightnessSlider.value;

                         break;

                    case _contrastSlider :

                         _motionTracker.contrast =
_contrastSlider.value;

                         break;

                    case _minAreaSlider :

                         _motionTracker.minArea =
_minAreaSlider.value;

                         break;
               }

               applyFilters();
          }



               // Script bwt virtual button nya
          private function
startdetectionHandler():void
          {
               masksArray[ 0 ] = new maskSrc();
             addChild( masksArray[ 0 ]);
```

```
                masksArray[ 0 ].visible = true;
               masksArray[ 0 ].alpha = 1;
                masksArray[ 0 ].width = 40;
               masksArray[ 0 ].scaleY = masksArray[ 0
].scaleX;
               masksArray[ 0 ].x = 350;
                masksArray[ 0 ].y = 20;

                /*masksArray[ 1 ] = new maskSrcbaju1();
               //addChild( masksArray[ 1 ]);
                //masksArray[ 1 ].visible = true;
               masksArray[ 1 ].alpha = 1;
                masksArray[ 1 ].width = 40;
               masksArray[ 1 ].scaleY = masksArray[ 1
].scaleX;
               masksArray[ 1 ].x = 150;
                masksArray[ 1 ].y = 120;*/

           }
           private function
shirt01detectionHandler():void
         {
                _ycoord.text = _ycoord.text + "(" +
flagShirt01 + "; " + flagShirt02 + "; " + flagShirt03
+ ")";
                for ( var i:int = masksSrcbaju1.length
- 1; i > -1; i-- )
               {
                 masksSrcbaju1[ i ].visible = false;
               }
                for ( var j:int = masksSrcbaju2.length
- 1; j > -1; j-- )
               {
                 masksSrcbaju2[ j ].visible = false;
               }
                for ( var k:int = masksSrcbaju3.length
- 1; k > -1; k-- )
               {
                 masksSrcbaju3[ k ].visible = false;
               }
                for ( var l:int = masksSrcbaju4.length
- 1; l > -1; l-- )
               {
                 masksSrcbaju4[ l ].visible = false;
               }
                for ( var m:int = masksSrcbaju5.length
- 1; m > -1; m-- )
```

```
            {
                masksSrcbaju5[ m ].visible = false;
            }
               for ( var n:int = masksSrcbaju6.length
- 1; n > -1; n-- )
            {
                masksSrcbaju6[ n ].visible = false;
            }
               for ( var o:int = masksSrcbaju7.length
- 1; o > -1; o-- )
            {
                masksSrcbaju7[ o ].visible = false;
            }
               for ( var p:int = masksSrcbaju8.length
- 1; p > -1; p-- )
            {
                masksSrcbaju8[ p ].visible = false;
            }
               for ( var q:int = masksSrcbaju9.length
- 1; q > -1; q-- )
            {
                masksSrcbaju9[ q ].visible = false;
            }
               for ( var r:int = masksSrcbaju10.length
- 1; r > -1; r-- )
            {
                masksSrcbaju10[ r ].visible = false;
            }

               //if (objectIndex >=1 && objectIndex <=
5)
               //{
                    if (objectType >= 1 && objectType
<= 2)
                    {
                         masksSrcbaju1[ 0 ] = new
maskSrcbaju1();
                         addChild( masksSrcbaju1[ 0
]);
                         masksSrcbaju1[ 0 ].visible =
true;
                         masksSrcbaju1[ 0 ].alpha = 1;
                         masksSrcbaju1[ 0 ].width =
160;
                         masksSrcbaju1[ 0 ].scaleX =
masksSrcbaju1[ 0 ].scaleX * 1.5;
```

```
                        masksSrcbaju1[ 0 ].scaleY =
masksSrcbaju1[ 0 ].scaleX * 1.2;
                        masksSrcbaju1[ 0 ].x = 125;
                        masksSrcbaju1[ 0 ].y = 90;
                    }
                    if (objectType >= 3 && objectType
<= 4)
                    {
                        masksSrcbaju2[ 0 ] = new
maskSrcbaju2();
                        addChild( masksSrcbaju2[ 0
]);
                        masksSrcbaju2[ 0 ].visible =
true;
                        masksSrcbaju2[ 0 ].alpha = 1;
                        masksSrcbaju2[ 0 ].width =
140;
                        masksSrcbaju2[ 0 ].scaleX =
masksSrcbaju2[ 0 ].scaleX * 1.5;
                        masksSrcbaju2[ 0 ].scaleY =
masksSrcbaju2[ 0 ].scaleX * 1.2;
                        masksSrcbaju2[ 0 ].x = 125;
                        masksSrcbaju2[ 0 ].y = 100;
                    }
                    if (objectType >= 5 && objectType
<= 6)
                    {
                        masksSrcbaju3[ 0 ] = new
maskSrcbaju3();
                        addChild( masksSrcbaju3[ 0
]);
                        masksSrcbaju3[ 0 ].visible =
true;
                        masksSrcbaju3[ 0 ].alpha = 1;
                        masksSrcbaju3[ 0 ].width =
160;
                        masksSrcbaju3[ 0 ].scaleX =
masksSrcbaju3[ 0 ].scaleX * 1.5;
                        masksSrcbaju3[ 0 ].scaleY =
masksSrcbaju3[ 0 ].scaleX * 1;
                        masksSrcbaju3[ 0 ].x = 125;
                        masksSrcbaju3[ 0 ].y = 100;
                    }
                    if (objectType >= 7 && objectType
<= 8)
                    {
```

```
                        masksSrcbaju4[ 0 ] = new
maskSrcbaju4();
                        addChild( masksSrcbaju4[ 0
]);
                        masksSrcbaju4[ 0 ].visible =
true;
                        masksSrcbaju4[ 0 ].alpha = 1;
                        masksSrcbaju4[ 0 ].width =
160;
                        masksSrcbaju4[ 0 ].scaleX =
masksSrcbaju4[ 0 ].scaleX * 1.5;
                        masksSrcbaju4[ 0 ].scaleY =
masksSrcbaju4[ 0 ].scaleX * 1;
                        masksSrcbaju4[ 0 ].x = 125;
                        masksSrcbaju4[ 0 ].y = 100;
                    }
                    if (objectType >= 9 && objectType
<= 10)
                    {
                        masksSrcbaju5[ 0 ] = new
maskSrcbaju5();
                        addChild( masksSrcbaju5[ 0
]);
                        masksSrcbaju5[ 0 ].visible =
true;
                        masksSrcbaju5[ 0 ].alpha = 1;
                        masksSrcbaju5[ 0 ].width =
160;
                        masksSrcbaju5[ 0 ].scaleX =
masksSrcbaju5[ 0 ].scaleX * 1.5;
                        masksSrcbaju5[ 0 ].scaleY =
masksSrcbaju5[ 0 ].scaleX * 0.80;
                        masksSrcbaju5[ 0 ].x = 125;
                        masksSrcbaju5[ 0 ].y = 100;
                    }
                    if (objectType >= 11 && objectType
<= 12)
                    {
                        masksSrcbaju6[ 0 ] = new
maskSrcbaju6();
                        addChild( masksSrcbaju6[ 0
]);
                        masksSrcbaju6[ 0 ].visible =
true;
                        masksSrcbaju6[ 0 ].alpha = 1;
                        masksSrcbaju6[ 0 ].width =
230;
```

```
                        masksSrcbaju6[ 0 ].scaleX =
masksSrcbaju6[ 0 ].scaleX * 1.5;
                        masksSrcbaju6[ 0 ].scaleY =
masksSrcbaju6[ 0 ].scaleX * 1;
                        masksSrcbaju6[ 0 ].x = 80;
                        masksSrcbaju6[ 0 ].y = 70;
                    }
                    if (objectType >= 13 && objectType
<= 14)
                    {
                        masksSrcbaju7[ 0 ] = new
maskSrcbaju7();
                        addChild( masksSrcbaju7[ 0
]);
                        masksSrcbaju7[ 0 ].visible =
true;
                        masksSrcbaju7[ 0 ].alpha = 1;
                        masksSrcbaju7[ 0 ].width =
160;
                        masksSrcbaju7[ 0 ].scaleX =
masksSrcbaju7[ 0 ].scaleX * 1.5;
                        masksSrcbaju7[ 0 ].scaleY =
masksSrcbaju7[ 0 ].scaleX * 0.90;
                        masksSrcbaju7[ 0 ].x = 125;
                        masksSrcbaju7[ 0 ].y = 50;
                    }
                    if (objectType >= 15 && objectType
<= 16)
                    {
                        masksSrcbaju8[ 0 ] = new
maskSrcbaju8();
                        addChild( masksSrcbaju8[ 0
]);
                        masksSrcbaju8[ 0 ].visible =
true;
                        masksSrcbaju8[ 0 ].alpha = 1;
                        masksSrcbaju8[ 0 ].width =
160;
                        masksSrcbaju8[ 0 ].scaleX =
masksSrcbaju8[ 0 ].scaleX * 1.7;
                        masksSrcbaju8[ 0 ].scaleY =
masksSrcbaju8[ 0 ].scaleX * 0.90;
                        masksSrcbaju8[ 0 ].x = 125;
                        masksSrcbaju8[ 0 ].y = 50;
                    }
                    if (objectType >= 17 && objectType
<= 18)
```

```
                    {
                         masksSrcbaju9[ 0 ] = new
maskSrcbaju9();
                         addChild( masksSrcbaju9[ 0
]);
                         masksSrcbaju9[ 0 ].visible =
true;
                         masksSrcbaju9[ 0 ].alpha = 1;
                         masksSrcbaju9[ 0 ].width =
160;
                         masksSrcbaju9[ 0 ].scaleX =
masksSrcbaju9[ 0 ].scaleX * 1.7;
                         masksSrcbaju9[ 0 ].scaleY =
masksSrcbaju9[ 0 ].scaleX * 0.90;
                         masksSrcbaju9[ 0 ].x = 125;
                         masksSrcbaju9[ 0 ].y = 50;
                    }
                    if (objectType >= 19 && objectType
<= 20)
                    {
                         masksSrcbaju10[ 0 ] = new
maskSrcbaju10();
                         addChild( masksSrcbaju10[ 0
]);
                         masksSrcbaju10[ 0 ].visible =
true;
                         masksSrcbaju10[ 0 ].alpha =
1;
                         masksSrcbaju10[ 0 ].width =
160;
                         masksSrcbaju10[ 0 ].scaleX =
masksSrcbaju10[ 0 ].scaleX * 1.8;
                         masksSrcbaju10[ 0 ].scaleY =
masksSrcbaju10[ 0 ].scaleX * 1;
                         masksSrcbaju10[ 0 ].x = 120;
                         masksSrcbaju10[ 0 ].y = 80;
                    }

          }
     }
}
```

ISLAM
UNIVERSITAS
INDONESIA