VU University Amsterdam
Department of computer science

Master of Science Thesis

MimicMe

Modeling essential features for creating recognizable avatars

by

Tom Bestebreurtje

Supervisor: Anton Eliens

Amsterdam, 2010


MimicMe
Mimic
avatar
face
users
used
online
model
also
avatars
customization
body
head
software
render
user
use
virtual
MimicMe's
different
fashion
clothing
website
texture
Virtual
hair
worlds
system
time
new
way
like
human
Figure
Mimic's
features
request
recognition
rendering
create
size
items
shape
using
Stylezone
options
textures
CHAPTER
results
parts
means
separate
development
server
information
number
range
introduced
games
modeling
site
database
uses
web
done
facial
make
audience
room
two
Fig
data
models
Graphics
many
application
form
looks
blend
research
created
applications
fit
internship
realistic
faces
look
jaw
community
type
still
much
map
real
Stardoll
Life
current
Looklet
allow
systems
less
appearance
parameters
set
shop
MIMICME
Users
want
sizes
course
process
images
mouth
width
style
FaceGen
chapter
recognizable
important
first
able
eyes
page
view
re
Faceworx
texturing
since
high
mesh
height
color
customizability
changed
apparel
every
skin
Second
tting
avatar's
age
longer
makes
length
customizable
requirements
rendered
possible
better
Mimics
lot
deform
basic
focus
user's
male
alter
see
now
main
made
world
sliders
level
altered
content
eyebrows
image
added
Avatar
values
section
available
nose
files
design
vectors
architecture
shapes
value
Summary
interface
IMVU
good
various
work
point
around
renderer
Twinity
allows
even
change
female
well
one
photo
Model
stored
Creating

Contents

Abstract

In this thesis I report on my internship at Mimic Media. It documents the research done on the topic of creating recognizable avatars for the MimicMe web application. A recognizable avatar is defined as an avatar made to look like an individual that is recognizable by people who know this individual's appearance.

MimicMe is a web application that uses an avatar that can be modeled after the user's own appearance for use in online fitting of clothes. In the rapidly growing market for online clothes shopping MimicMe aims to provide its users with an opportunity to virtually try out clothing items on a look-alike avatar, called a Mimic.

An important goal of MimicMe is to provide the user with an accurate virtual representation of him- or herself in order for the user to be able to identify with his or her created representation. What effect an accurate representation has on the user is one of the topics being researched. To maintain a level of user-friendliness consistent with its target audience a sub-goal of MimicMe is to have the process of customizing an avatar as simple as possible, while maintaining a balance between the amount of required user input and the level of recognizability achieved with these inputs.

The use of avatars has become the de facto standard for communication and representation in virtual worlds. To explore the ways in which an avatar's appearance can be altered a wide range of virtual worlds will be examined on their aspects of avatar customizability.

The human shape, especially its face, has been the subject of many studies on our ability to recognize persons. Results from studies in the fields of Biological Vision, Computer Vision and Image Processing are used here in an attempt to find which features of the human form are essential to model in MimicMe.

After examining MimicMe and working on its development a comparison is made between MimicMe and other applications concerned with avatar customization and fashion (sub)culture. Possible solutions and improvements for avatar creation will be discussed.

In conclusion, some of the studied principles have been implemented in MimicMe, while others might be implemented in the future. The possibilities will allow MimicMe to ultimately achieve its desired level of avatar customizability.

Keywords: Avatar, Online fashion, Fitting, Facial recognition, Virtual world, Anthropometry, Measurements

Chapter 1

Introduction

While online shopping started with an online bookstore in 1992 (two years before Amazon.com's creation), today one can buy almost any product via an online retailer or *web shop*. Online clothing sales in particular have increased substantially[1] in the last 5 years after initially being somewhat left behind in the online market by books, CDs/DVDs and electronics.

This can be attributed to the fact that shopping for apparel items is an activity that generally requires a prospective buyer to be able to touch and feel the products and then take clothing items to a fitting room to see if they look good when worn on her person. This level of feedback is hard to achieve in a web shop, where the consumer has no possible way to try on items before ordering them.

To make up for the lack of an actual fitting room, online apparel consumers are encouraged to shop at online retailers because of very consumer-friendly return policies among other reasons [7]. As the market for online apparel shopping continues to grow, the return rates of clothing items in particular remain high.

Since virtual worlds like Second Life became popular around 2006, the use of avatars on the Internet has also increased dramatically. Various social networking websites use avatars to allow users to engage in virtual 3D chats and customize their looks, while the use of avatars in games has also continued to become more complex.

Retailers and other companies attempt to include various forms of a virtual fitting room into online shopping experiences to make consumers more certain about their purchase. These include digital assistance in taking one's measurements and thereby allowing consumers to select the right size products in a web shop, thus lowering the chances of the product being returned because of a wrong size. Shops also provide more and more images, even fabric close-ups[2], of their products in order to recreate

[1] www.admanager.nl/online/nieuws/9366/CBS_online_shoppen_steeds_populairder/
[2] www.welikefashion.com

the touch-and-feel environment of a regular apparel store. The implementations of various forms of an online virtual fitting room are also being explored in many cases[3].

MimicMe[4] is such a virtual fitting room. Through the creation of a Mimic, a lookalike avatar, MimicMe provides consumers of web shops with virtual representations of themselves for use in an online fitting room. The goal is to provide users with an accurate virtual representation of themselves and the clothing they are trying on. The aim is to allow consumers to make a more informed decision about their purchase before the item is ordered and tried on upon delivery. This means that less ill-fitting, non-matching or unflattering items are returned. Lower return rates greatly help online retailers' businesses and MimicMe aims to create a solution for web shops of any size.

Since, at this time, MimicMe caters exclusively to a female audience it seems fitting to use female pronouns when referring to the user. Also, since the application can potentially be used not only at MimicMe's main website, but potentially also at a web shop that has included Mimicme on their website, the terms user and consumer are used interchangeably.

1.1 Research question

The virtual fitting room of MimicMe is centered on the Mimic, which serves as the user's personal mannequin. A Mimic's measurements, facial features and hair can be altered to the user's taste. Most often this results in the user changing the features to most closely resemble him- or herself [11]. This is the intended purpose of the Mimic, as it aims to let the user see herself in her Mimic, thus acting like a virtual mirror like in conventional in-store fitting rooms. The more accurate the representation, the more engaged the user becomes with her shopping experience, which increases the effectiveness and the value of the product, Mimicme.

The Mimic is used for modeling MimicMe's virtual clothes and representing the user. My role as intern at MimicMe is to assist in development and maintenance of the Mimic and the website interface around it. My main task is managing the Mimic's level of recognizability. The goal of my research is to determine what makes an avatar, like the Mimic, recognizable. Recognition of an avatar in this context means that the user or her friends can recognize the Mimic's owner in the Mimic's appearance. An important factor in creating a recognizable avatar for MimicMe is the ease with which the user should be able to create a Mimic with a sufficient likeness to herself. The issue of balancing the complexity of modifying an avatar and the number of actions a user ideally likes to perform in total is important, because ultimately, it's the users that need to be satisfied with the level of realism.

[3] http://gizmodo.com/391508/virtual-fitting-room-turns-you-into-an-online-paper-doll
[4] www.mimicme.com

Since MimicMe does not yet have a large loyal user base, measuring the level of recognition through questionnaires is difficult. However, scientific literature on facial recognition, anthropometry and digitalization of the human form through 3D scanning do provide empirical results that indicate the workings of the human visual system, and thus what areas of the body and/or face to focus on to achieve recognizability. By examining various modern virtual worlds that deal with fashion and lifestyle through the use of avatars, I attempt to find parallels between the techniques discussed in the literature and those that are being applied in virtual worlds and see if this knowledge can be used to further improve the Mimic.

MimicMe is a startup enterprise, which researches and implements new features at a very high rate. The range of features introduced for the MimicMe is ever-changing, which has caused that its appearance and features have already evolved further than my research describes. I will mainly focus on the research done at MimicMe during my internship. A look at MimicMe in its current state is presented in appendix A.

1.2 Thesis Structure

This thesis centers around the questions of how to customize an avatar in order to make it recognizable and determining an efficient, user-friendly way of achieving this recognizability. Finally, this research has to be placed in context of the MimicMe product, bearing in mind the limits of what can be achieved at a small startup enterprise in terms of available financial means and manpower.

Chapter 1 contains the introduction and research question. In chapter 2 I describe Mimic Media's business context, its mission statement and technological architecture as well as the process which is central to MimicMe: the depiction of the Mimic. I also take a look at some of MimicMe's competitors, since they provide MimicMe with valuable information on the topic of avatars and fashion communities.

In chapter 3 I study the way avatars have been used in the past and how they are used in modern video games, virtual worlds and fashion applications. I compare the avatar systems of a number of applications with regard to the general style used and the existing options for customization. Many ideas for future development of MimicMe can be taken from these existing applications.

MimicMe's design requirements are formalized in chapter 4. What do MimicMe's users want? What does MimicMe have to offer its users from a technological point of view? Using the information on avatar customization from chapter 3, in addition to ideas for customization that are developed in-house, we explore possibilities for further enhancement of the Mimic's features using third-party software packages for human face modeling.

Chapter 5 lists the results of all work done at MimicMe on both the Mimic's functionalities and the MimicMe website. A more detailed list of my activities at MimicMe is presented in appendix B. Chapter 6 concludes with an overview of Mim-

icMe's current potential. A short anecdotal look at what MimicMe has become after ending my internship is presented in appendix A. MimicMe has already changed significantly since my internship ended. This is a good opportunity to see the effects our research has had on the Mimic, and whether or not MimicMe has changed course since then.

Chapter 2

MimicMe

In this chapter I give an overview of the startup enterprise Mimic Media, its goals and the technology we used while working on Mimicme.

2.1 Origins

The growing market for online apparel shopping, combined with the relative ease of setting up a web shop, has caused many small fashion outlets to appear that only sell online. Smaller web shops generally target consumers of a certain demographic, as opposed to the larger online apparel sellers, established manufacturers that have an online store and large order-by-mail companies.

At time of writing this thesis, MimicMe is the only product of Mimic Media, founded in the Netherlands on December the 28th of 2007, by Xavier Baars, Tom Burger, and Corné Overbeeke. Mimic Media started out as a V.O.F., but later changed to a B.V. in December of 2008. The name MimicMe in this document is more or less equivalent to the main enterprise, Mimic Media, and will be used interchangeably. MimicMe's initial conception began with Baars's enterprise he had set up during his studies, which aimed to enhance online product presentations by allowing 360-degree views. In 2006 he noticed the rapid growth in the online apparel market. Clothing was, and is often still presented online with very few images, in contrast with the touch-and-feel environment of conventional clothes shopping. From this basic presentation consumers did not get a good idea of how the clothing would look when worn, resulting in over 30% of the items being returned. With the current standard in web technology and game design, Baars thought it would be possible to create a virtual fitting room. Baars and Overbeeke knew each other previously and decided to start up the enterprise together. After some brainstorming they further developed the idea that would become MimicMe. After an intensive search for an experienced programmer they came into contact with Burger.

They named their enterprise Mimic Media, and their virtual fitting room was named MimicMe. They began developing a prototype for their application, which gradually evolved into the current version of MimicMe. Funds were secured by finding an investor, so for the next two years they could buy the necessary hardware and software. Baars handles Mimic Media's marketing and external contacts. He researches the state of the art of the online fashion industry and designs the MimicMe website's layout. Overbeeke taught himself how to use 3D modeling software with which he models the Mimic and the various apparel items that are used in MimicMe. Burger developed the MimicMe software and website. After some time they hired an additional 3D modeling expert to help with streamlining the modeling process. Besides the three founders and one lead 3D artist, Mimic Media regularly hires interns to assist with web development, modeling and analyzing fashion trends. Yours truly worked on research and development with Burger, focusing mainly on the customizability of the Mimic.

2.2 The Mimic

The Mimic is the user's avatar and the centerpiece of Mimicme. The main website, www.mimicme.com functions as a showcase for MimicMe, and it is here that new features are first introduced. However, a main goal is to have multiple separate MimicMe applications implemented on multiple web shops, each instance tailored to each shop's wishes. In principle, each user has one Mimic linked to her account, but it is possible to modify the system to have several Mimics available to manage per user. While the Mimic is essentially used as a static mannequin, it's virtual nature allows its appearance to be altered so that the user can adjust the Mimic's physical appearance to her wishes. At any time, the user can change their Mimic's appearance, whether to update the Mimic's hairstyle and/or color to match the user's hair after a recent haircut, or just to play with the different settings.

Adjusting the Mimic's appearance to the user's wishes is one of the two core functionalities of MimicMe. This section of MimicMe is named *Mimic* because it deals exclusively with the Mimic's measurements and does not allow for clothing items to be used other than some basic underwear.

Next to customizing the Mimic's appearance, using the Mimic as a mannequin to try on clothing is the most important aspect of MimicMe. This section is called *shop*, and places the Mimic next to a selection of clothing. It is the complete picture of the user's Mimic and the selected clothing worn by the Mimic that forms the core experience of MimicMe. If the user has created an outfit that she wants to remember or share, this can be done by posting a snapshot to the MimicMe gallery or by sharing it via Twitter [1] or the like. This feature and the possible social networking applications are briefly mentioned in chapter four.

[1] www.twitter.com

Figure 2.1: Various Mimic images

(a) Amber (b) Eric (c) Black skin (d) Optional view of the lower body

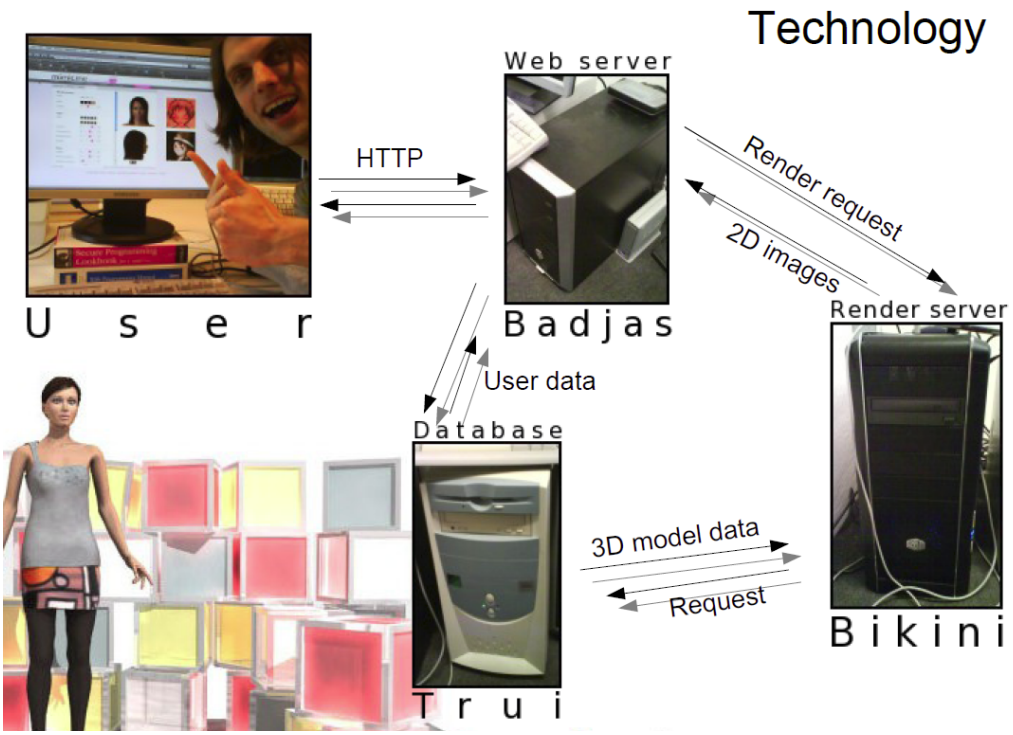
While the MimicMe interface is constantly undergoing changes, at its most basic, the underlying concept remains the same and can be divided into these sections *Mimic* for the body, and *shop* for the clothing.

2.3 MimicMe systems and architecture

As stated, MimicMe's aims to cause a decrease in returned items for web shops. Of course a virtual fitting room also provides the users with entertainment and inspiration, which are also beneficial for a web shop. It is important for MimicMe to be a viable solution for web shops of any size. An important way to achieve this is by making the application as portable as possible.

The hardware architecture used at MimicMe has changed over time. The main system used to consist of three machines: A web server, the *badjas* hosted MimicMe's website (Fig. 2.2). The *trui* served as a MySQL server as well as the file system in which all 3D object files were stored. Finally, the render server *bikini* rendered all the images of the Mimics used on the website. When the MimicMe systems moved to a dedicated hosting provider, these servers' functionalities were integrated into a single 19" rack mount, the *jurk*. If MimicMe's user base grows in the future, so that a single server can no longer carry the load, the architecture might be organized as separate machines once again.

Figure 2.2: MimicMe's architecture when we still used separate machines



2.3.1 Web server

MimicMe's website uses a combination of PHP and AJAX to display its pages. This includes the Mimic, which is rendered as a 2D image. The implications this has for the user are mentioned below. In contrast to many new websites that fit into the Web 2.0 stream of sites, MimicMe was not built on any specific framework, but it has been loosely based on the Symphony framework. That framework, in turn, has been based on the Model-View-Controller design pattern (MVC). With MVC, domain logic that is internal to the application is isolated from the input and presentation layers, allowing for separate maintenance of both. Eventually, some features from different JavaScript frameworks were also included in order to use certain widgets and effects. See chapter 5 for an overview of the developments of the website.

To be implemented on a different website, MimicMe simply requires an iframe element on the page or web shop in question that links to that store's page on MimicMe's web server.

2.3.2 Database

All data pertaining to the MimicMe website is stored in a MySQL database. A transition was made from a system where every user had her Mimic settings included in her user data to a system that is more modular. This separated the user from

her Mimic settings and simply assigned ID's to each Mimic in the database, so that theoretically user's could have multiple Mimics stored at te same time. This could prove useful for situations in which a customer would want added functionalities for users of MimicMe on their site. The resulting system is referred to as the Web-API. It was designed to add more structure to the application and with a future in mind where multiple instances of the MimicMe application could access the MimicMe database more easily.

The database contains all user data, Mimic data, and Mimic images. User data consists of typical things such as user names encrypted passwords the users' measurements and the users' uploaded icons/avatars. Each user also has a Mimic ID associated with it. As stated, this could be expanded in the future to allow for multiple Mimics per user account. The Mimic Database contains the parameters for each of the Mimics, each with a unique ID. The images database contains images of Mimics from the shop that have been saved by users, as well as all other Mimic images that have been rendered, for caching purposes.

Not stored in a database, but plainly in the file system are all the 3D files that make up the Mimic, plus every individual item that can be worn by the Mimic or otherwise rendered. Over the course of my internship, the database has undergone many changes, but the essence remains the same.

2.3.3 Renderer

MimicMe's render server is the most complex component of the system. Written entirely from scratch in C, its main purpose is to render the Mimic when requested to do so by the web server, which usually acts on behalf of the user. The Mimic consists of several parts (body, head, hair, etc.), which are subject to change as the render server, and thus the process by which the Mimic is rendered, is altered. The parts are meshes in Waveform OBJ format, along with image files for textures. Clothing items are organized in a similar fashion. All parts are rendered according to the parameters given in the request. The so-called render request comes from the web server, and after the renderer has completed rendering the image it returns the file location of that image to the web server.

One of the most notable features is that the web server therefore provides the user with 2D image files of the user's Mimic. Even though the application uses 3D graphics on the server side, the user sees 2D images displayed in its browser. The single most important output is the Mimic that is displayed on the user's screen. It is used when changing the Mimic's body proportions, size, and hair color, and to show the Mimic wearing its various outfits. Portability is achieved by rendering the Mimic as a simple JPEG image. The implications of rendering all 3D data server-side will be discussed in chapter 4. The technical side of the rendering process is explained in the next section.

2.4 A technical analysis of the Mimic

This section describes in more detail the steps that are made to render the Mimic before storing the image and sending the location to the web server.

For every alteration to the Mimic the web server issues a new render request to the render server. While essentially the same, we can still distinguish two cases: Either the Mimic is changed because the body has been altered or an item of clothing is removed or added; or the Mimic is viewed from a different angle, which also requires a new render of the same Mimic, but from the new angle.

The Mimic can always be viewed from a limited amount of angles. Originally, this was 360 degrees divided in six angle of 60 degrees each. One alternative to rendering a new Mimic with every turn is to render a complete set of angles for each change that is made to the Mimic. This way, the other angles are already present in the cache, in anticipation of the user wanting to view the change to the Mimic from a different angle. The Mimic can also be viewed from a set of possible camera angles, allowing for close-ups of the face, upper body, or legs. Possible expansions of this functionality are mentioned in chapters 4 and 5.

2.4.1 Communication of the request

The rendering of any Mimic is done by submitting a render request to the render server over a designated socket. A render request is a string which contains all information needed to render a specific image of a Mimic. This includes the viewing angle, the Mimic's body and facial parameters and every garment that the Mimic is wearing. The renderer will return the file name of the completed render once it is complete. If there were any errors, the renderer will return `#error`, or simply close the socket. The website then interprets this error and displays a default image instead, notifying the user of a problem. Possible errors in rendering are errors in parsing of the request, missing files, or a bug in the renderer's programming that resulted in a failure.

2.4.2 Files that make up the Mimic

The Mimic is made up of various parts that the renderer uses to create each Mimic. These parts are stored in the file system in Waveform OBJ format [2]. The Mimic is divided into body, feet, head, and hair. As stated in the renderer's overview, the Mimic's division into parts is subject to change as new features are introduced. All OBJ files for use at MimicMe are exported from Autodesk Maya 3D [3]. The OBJ files contain all geometrical information that is needed for the object: Vertices in (x,y,z), Texture coordinates in (u,v), Normals in (x,y,z), and the organization of vertices,

[2] http://en.wikipedia.org/wiki/Obj

[3] http://usa.autodesk.com/adsk/servlet/pc/index?siteID=123112&id=13577897

texture coordinates, and normals into faces. The texture for each object is created using Autodesk Mudbox [4].

It is important in areas where parts have to be joined precisely, like the neck and ankles, that the data from the two OBJ files matches exactly. This is done once, when each model is taken into use, by running a script that identifies the vertices located on the seam from both OBJ files and rounds them to matching values to remove any irregularities in the geometries and shading when the two objects are placed together. This is necessary to counter slight variations in the floating point values of the vertices that are exported from Maya 3D. While the geometrical differences are not really noticeable, the differences in shading would be visible as a ring around the neck.

2.4.3 The render request

The parameters of the render request are frequently changed, to accommodate new options for the Mimic. When the parameter organization is changed, this has to be reflected in multiple locations: The database must be altered for the new parameter to be stored, the website and its logic must be altered to access the parameter, and lastly, the parameter must be inserted in the render request. An altered render request means that the renderer's parser must be altered to read the newly formatted render request and to perform the action(s) associated with the new option.

For instance, when the option was added for the Mimic's feet to be positioned either flat on the ground or in a "raised heels" position, changes had to be made in the clothing database to be able to distinguish flat shoes from shoes with heels. The website was then reprogrammed to extract this extra information, and the render request was altered to include an extra parameter specifying whether the Mimic should have level or raised heels to match the chosen shoes. Finally, this new render request had to be parsed correctly in the renderer. The process of rendering the Mimic then had to be changed by splitting the Mimic's body into a *feet* and a *body* part. A function then had to be added to the renderer that, depending on the type of feet selected, loaded the correct models for the feet and also positioned all the other parts at the correct height. This functionality was later expanded to accommodate four different heights of heels (including the basic flat position).

[4] http://usa.autodesk.com/adsk/servlet/pc/index?id=13565063&siteID=123112

A typical render request in its entirety looks like this:

```
render #1 #0 #1.74090909091 #0.866497933884 #0.668150031786
#0.491802546348 #0.336877373241 #0.504521147302 #0 #0.79 #1
#8 #b #body #/net/database/amber-1-april/blend/body/diffuse1
#m #head #23 #0 #-0.666666666667 #0.25 #-1.33333333333
#-0.5 #0 #0.5 #0.05 #0.4 #0.65 #0.32 #0.25 #0.15
#-0.433333333333 #0 #0 #0 #0 #0 #0 #0.269 #0.613 #0.500337774669
#/net/database/amber-1-april/deforms/head/diffuse-2-2 #n #eyes
#/net/database/amber-1-april/static/eyes/diffuse3 #b #feet1 #none
#n #hair23 #/net/database/amber-1-april/static/hair23/diffuse-6-128x128
#t #277 #b11 #278 #/net/database/amber-1-april/blend/278/diffuse
#u #eb-2-3-2 #FILENAME #0 #60 #300 #
```

This type of render request was used at one point during my internship. I will divide it into sections to illustrate the operations the renderer performs in order to use it for rendering the Mimic.

`render #1 #0`
Each render request starts with *render*. The # symbol is used throughout the request to separate the parameters. I have added the spaces to improve readability. *The Model ID*. This means that depending on the parameter, a different model is read. For most of the time, MimicMe has worked with one model, named "Amber." At one point we introduced a new male model, named "Eric." The male model was later removed again, in order to focus more on the development of MimicMe's core female target audience. It helps to see a model as a differently shaped Mimic; A child could also be a new type of Mimic, once again separated into a "boy" model and a "girl" model. It is important to note that a model on the file system is a directory, and within that directory everything for that model is stored. This means all body parts, possibly with their own deformation vectors (see further below) and textures, but also all garments for that Model are stored within this directory. Since different models can contain a different set of parts, the renderer's parser must have some hard-coded way of determining which model type is being read and follow the right procedures for rendering that model. Ideally however, there should be a certain uniformity, allowing the renderer to be able to process any model, as long as it is structured in a predetermined way. This was not the case during the time the male model was present on the website.

Clipping value. This is used to clip certain parts of the Mimic away because, depending on the Mimic's measurements and the type of garment worn, visible clipping errors could be seen. This was most often visible in areas of great geometrical variation, like the breasts and armpits. Depending on the clothing items selected,

the web site sets the Clipping value to indicate which set of clippable parts should be clipped out.

The various techniques created and employed by Burger to fit relatively roughly modeled clothing tightly to the Mimic's skin surface without any visible anomalies are impressive, but fall outside the scope of this thesis which deals with the customizability of the Mimic.

`#1.74090909091 #0.866497933884 #0.668150031786 #0.491802546348 #0.336877373241 #0.504521147302 #0`

Seven floating point numbers that control so-called blending values. Blending is the term used in MimicMe for a function that alters body proportions. The eight blends used here are: height, leg inseam length, hip size, waist size, under bust size, bust size, shoulder width. Blend values are used by the renderer to change the shape of the body by taking vertices at and around a specific height like the waist, and expanding or contracting that ring. Similar vertex alterations are done to adjust the relative leg length and the overall Mimic height. These blend are presented to the user as a way to fine tune their appearance. A lot of the variance in the blend values is actually controlled by one parameter: "weight". This parameter could have more aptly been named "build", since it alters all blend values, making the entire Mimic larger or smaller. After processing the weight parameter, the individual blend parameters are applied, which typically multiply the new blend value by a factor 0.85 to 1.15. To allow for a better approximation of real world sizes, the original model's measurements, which were included with the model, were taken and these measurements could be synchronized with the blend values so that the floating point numbers used can be read as metric units. This Mimic, for instance, is approximately 1.74 meters tall. Calculating the weight for the Mimic using volumetric approximations was also thought of, but would be too hard to determine accurately, so this was left for the time being. Adjusting the height was one of the main reasons to separate the Mimic's head from the body, so that making a Mimic taller would not also make its head taller; a deformation that would of course be unrealistic and therefore undesirable. The last entry for shoulder width was added to make the male model's shoulder width adjustable. The corresponding blend function was never implemented, so it remains a dummy blend for now and could be used when MimicMe decides to start using a male model again.

`#0.79 #1 #8`

Skin shade. As can be seen below, further into the render request, the Mimic uses a set of four basic textures to simulate skin types. I have researched human skin types using Von Luschan's chromatic scale [5] so that the user can fine tune her skin tone to be lighter or darker (Fig. 2.1c).

[5] http://en.wikipedia.org/wiki/Von_Luschan's_chromatic_scale

Camera. A set of camera positions can be chosen for viewing the Mimic. When altering the Mimic's facial features a close-up of the face is used. When selecting a hair style and color the camera centers on the top half of the Mimic. Another camera angle centering on the Mimic from the waist down is also used (Fig. 2.1d). Possibilities and trade-off of multiple camera angles and degrees of freedom for the user are discussed in chapters 4 and 5.

The number of parts that follow. Right now, only the body's blend parameters have been read, but have not yet been processed and applied to the Mimic's body mesh. In theory, all following parts, including the head, could allow for choice between different shapes, as all parameters refer to a certain directory within the model's main directory. The head and body meshes have been carefully aligned for the model "Amber," so any variant meshes should be made to match as well. Each type of part used is indicated with a letter as the first element, typically followed by a name which corresponds to a subdirectory where the mesh and one or more textures fo that mesh are located. Textures are stored as .png or .jpg files.

`b` - Blended mesh
`n` - Non-blended, static mesh
`m` - The head, including all parameters for the Mimic's facial characteristics
`t` - Texture-only garment
`u` - Makeup texture

```
#b #body #/net/database/amber/blend/body/diffuse1
```

The mesh is altered by the renderer using the blend values read from the beginning of the render request to control the girth of the Mimic's meshes in predetermined locations such as hips, breasts, etc. `Diffuse1` refers to one of the four skin types that are available. The types are: Caucasian, olive, brown, and black. Many values in the database are used to indicate in which directory a certain mesh or texture is stored. In this case, the database's `mimic_skin_type` entry has the value 1 which is then used in the path that points to the right texture.

```
#m #head #23 #0 #-0.666666666667 #0.25 #-1.33333333333
#-0.5 #0 #0.5 #0.05 #0.4 #0.65 #0.32 #0.25 #0.15
#-0.433333333333 #0 #0 #0 #0 #0 #0 #0.269 #0.613 #0.500337774669
#/net/database/amber/deforms/head/diffuse-2-2
```

The head uses the mesh stored in the directory of the same name. The renderer is programmed to use the same texture as it does for the body. The next number indicates the number of deform values that follow for the face. A deform is a vector, created in Maya 3D, that acts on a group of vertices. For each of the 23 deform values mentioned in this version of the render request, there is a deform vector stored in the same directory as the head. Deform vectors alter facial characteristics, like the width of the eyes, length of the nose, fullness of the lips, etc. The last deform value

controls the overall "fatness" of the face, depending on the measurements of the body. It takes the blend values, except for those dealing with length, and calculates the appropriate value for the deform vector so that the head's area around the neck has a size that matches the size of the body mesh's neck area. It also creates a fuller chin and jaw, in accordance with heavier body types.

Depending on the current design of the website, and thus the number of parameters accessible to the user, some deforms might remain unused. This can be because its value was set to a specific value, that is not meant to be changed by the user, or because the deform in question is no longer used and discarded by the renderer. The deform for the height of the cheekbones, for example, is such an unused deform that was designed and functional, but was left out of the Mimic's customizability design and thus not accessible via the website and left at its default value, which was determined to be a good height for cheekbones.

The 22 deforms used in this version of the render request are:

`eye_size` (Size of the eyes)
`eye_width` (Distance between the eyes)
`eye_height` (Vertical position)
`eye_slant` (Tilted inwards or outwards)
`nose_length` (Depth of the nose)
`nose_tip` (Points the tip of the nose up or down)
`nose_width` (Nose width at the nostrils)
`nose_nostrils` (Position of the nostrils: up or down)
`nose_bridge` (Nose width at the bridge)
`mouth_height` (Vertical position)
`mouth_width` (Width of the mouth)
`mouth_corners` (Mouth corners up or down)
`head_forehead_width` (Width of the forehead)
`head_cheekbones_width` (Width of the cheekbones)
`head_cheekbones_height` (Height of the cheekbones)
`head_jaw_corners_width` (Width around the jaw corners)
`head_jaw_width` (Width of the jaw)
`head_chin_length` (Length of the chin)
`head_chin_width` (Width of the chin)
`head_head_length` (The length of the head, affecting the height of the forehead and length of the jaw)
`mouth_upper_lip` (Upper lip fullness)
`mouth_lower_lip` (Lower lip fullness)

A very important aspect creating a Mimic in a user-friendly way is to have less

parameters for the user to alter. Finding ways to reduce the number of parameters and making the system more intuitive as a whole is one of the goals of my research.

The last part of the head section of the render request is a placeholder that is no longer used. The render server would crash however, if this head texture is not included. This is a typical example of the constant evolution of the render request.

`#n #eyes #/net/database/amber/static/eyes/diffuse3`
Because models often come with the eyes as a separate mesh, to allow for animation of eye movements, they are loaded separately from the head. The texture controls the color of the iris.

`#b #feet1 #none`
The feet mesh. This is an example where there is more than one option for a mesh for the same body, namely the feet in flat or in raised heels position. The renderer is programmed to compensate for the extra height by translating the rest of the Mimic upwards. No texture is needed, since the feet use the body texture.

`#n #hair23 #/net/database/amber/static/hair23/diffuse-6-128x128`
Like the eyes, the hair is a static shape. Each hairstyle is a different mesh with its own range of textures. while we strive for the same range of colors for each style, it is common for the older hair styles to have less colors to choose from. The choices the user can make are hard coded into the website. In addition, each texture comes in a 128x128 and a 256x256 pixel format. The larger one is only used for close-up camera angles because aliasing effects distort the hair's pattern from larger distances. A problem arose when we made the width of the head, especially the forehead, customizable. The Mimic's temples would clip through the hair mesh, which was particularly visible with the shorter hair styles. To counter this, we programmed the renderer to take the head width into account when rendering the hair and make the hair mesh wider so it would fit again.

`#t #277`
A textured garment, most commonly seen in the form of a legging. At the time we introduced this type of garment, Burger's automatic cloth fitting system was already quite good at making garments fit the shape of the Mimic. However creating a tight fit as with a legging still produced clipping error, and using a texture was a logical fix, because it was also less expensive in terms of rendering.

`#b11 #278 #/net/database/amber/blend/278/diffuse`
A typical garment, containing a number and a texture, so that the renderer can locate it in the file system. In the case of garments, the 'b' can be followed by two numbers that control render-time adjustments of the garments with respect to

clipping and body vertices being moved when they are not supposed to. As stated before, these additions to the render request fall that deal with garment adjustments fall outside of the scope of this thesis.

`#u #eb-2-3-2`
The `u` indicates makeUp. Right now this is the only type of makeup. To alter the facial texturing of the Mimic, we use a specially formed UV [6] map of the face to which various textures can be applied. In this case, the only makeup used is not really makeup, but are the Mimic's eyebrows. Each type of eyebrow is a texture with transparency that is added to the face. Of course this principle can be extended to adding blush, makeup around the eyes, and other things that could be textured. With 5 shapes of eyebrows, 4 colors and 3 thicknesses, this was a lot of manual photoshop labor, but the results were satisfactory. If the coloring, or even some shaping of the textures can be automated, then this way of adding texture can be extremely useful for further personalization of the Mimic.

`#FILENAME-HASH #0 #60 #300 #`
Finally, all information in the render request is hashed into a *filename* for that unique Mimic. The last three numbers indicate that the Mimic should be rendered with a rotation of 0 degrees -which means a frontal view-, and then again from an angle increased by 60 degrees, up to 300 degrees. This way, the result is six rendered images of the same Mimic from 6 angles. The angle can be read from the filename, because it is not hashed into it, creating filenames ending in `-30.jpg` to indicate a 30-degree rotation.

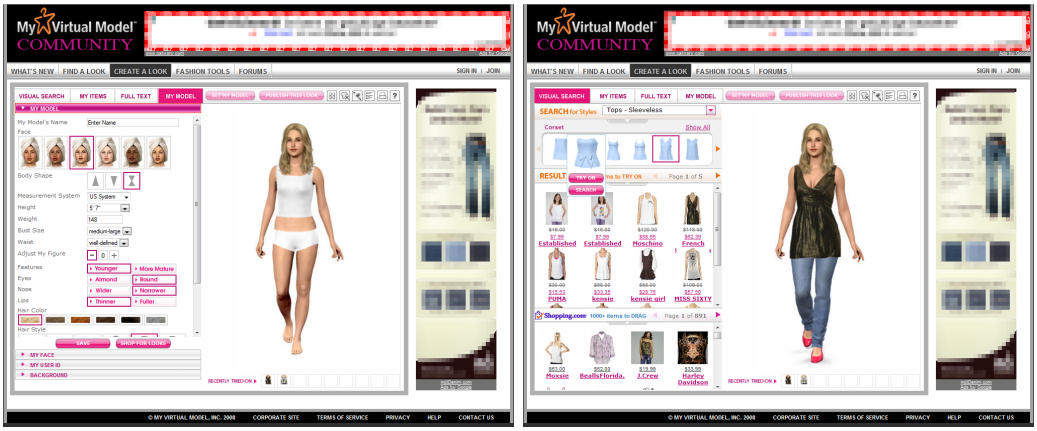
Having fully described the render request, we can see that every new functionality that is conceivable in the area of meshes, blends, deforms or texturing can be added to the request. It is worth repeating that this is a snapshot of the current state of the render request. The render request serves as a framework within which we can describe other future alterations to the Mimic's adaptability.

2.4.4 Amber and Eric

While we were still working on improving the initial female Mimic, named Amber, we introduced a male model, named Eric (Fig. 2.1b), to the website. Eric used the same methods of adaptability as amber. It quickly became clear however, that a male model needed different deformation vectors for the male face. In theory, if enough basic deforms are available, both the female and male faces could be shaped using the same vectors. Not the exact same vector .OBJ files of course, because each vector is modeled for the specific male or female mesh, but the same vector, like in how far the eyes are apart, or how wide the chin is. With the male model we needed to focus more attention on the jaw line in particular. This resulted in a lot

[6] http://en.wikipedia.org/wiki/UV_map

Figure 2.3: My Virtual Model



(a) Customizing your avatar (b) Browsing the shop

of hard coded functions in the renderer, because the two models now used different sets of deforms. The render requests for Amber and Eric now differed in the number of parameters.

2.5 Business context

MimicMe is a relatively new enterprise in the online fashion market, but even in this young branch of Internet marketing there are already several competitors with similar products.

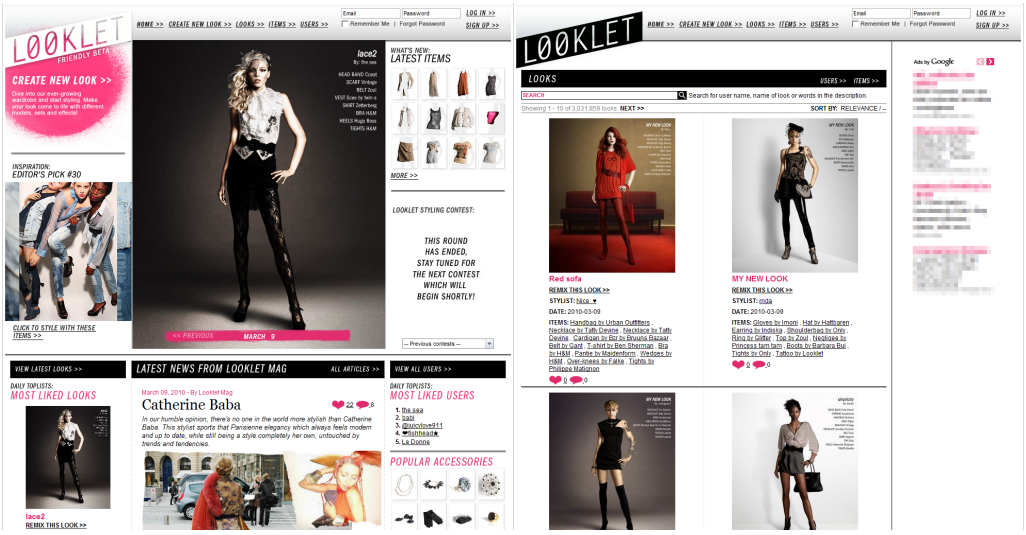
My Virtual Model[7] (fig. 2.3) is one of the closest competitors MimicMe has. A survivor of the first dot-com bubble of the late nineties, Canada-based My Virtual Model has a longer track record [8] but also suffers from its size, while MimicMe rapidly prototypes and adapts the Mimic to its user's wishes. It was nice that, as my internship neared its end, we all realized that MimicMe was about to surpass My Virtual Model in terms of graphical quality.

However, other new enterprises also embrace one of the golden rules[8] of startups: "launch fast." A very popular virtual dressing room that had its closed-beta launch around the start of my internship is Looklet[9] (fig. 2.4), a virtual dress-up site that uses cleverly blended photographic material to create its *looks*, that are more like magazine photo's, but without the use of 3D graphics, are not as interactive. The functionalities of My Virtual Model and Looklet will be studied further in chapter 3.

Another site that is all about fashion -except that there is no avatar to put

[7] www.mvm.com
[8] www.paulgraham.com/13sentences.html
[9] http://looklet.com

Figure 2.4: Looklet

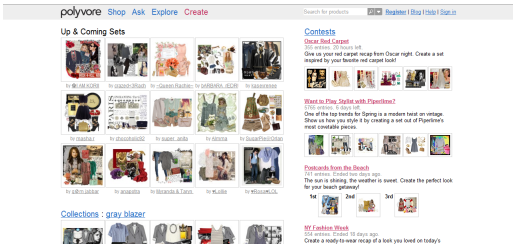


(a) Front page (b) Different looks

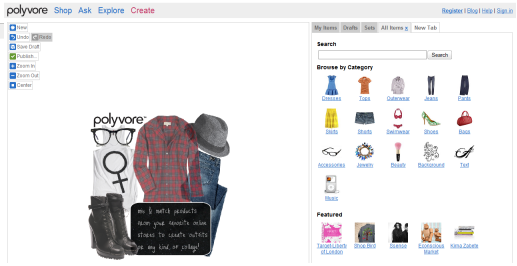
clothes on- is Polyvore[10], a fashion scrapbook community where users can piece looks together from various images of apparel items. (fig. 2.5) These sites offer valuable insight in the dynamics of web communities, which is what MimicMe also aspires to become.

[10] www.polyvore.com

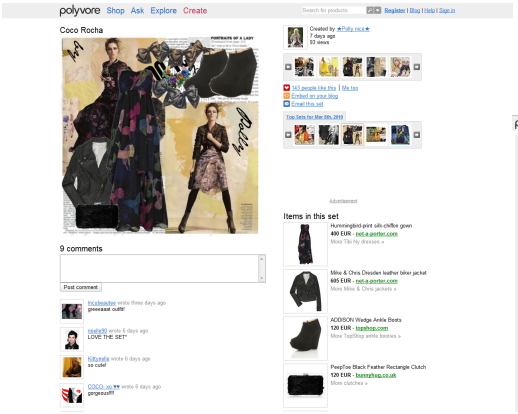
Figure 2.5: Polyvore



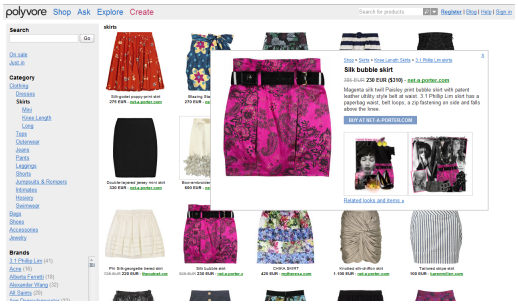
(a) Exploring the many different sets



(b) Putting together items to create a set



(c) Viewing the details and comments of a set



(d) Examining a single item often links to a web shop. Example sets using this item are also shown

Chapter 3

Fashionable avatars

From the Free Merriam-Webster Dictionary:

av.a.tar

Pronunciation: \'a-vè-,tär\
Function: noun
Etymology: Sanskrit avatāraḥ descent, from avatarati he descends, from ava- away + tarati he crosses over more at ukase, through
Date: 1784
1 : the incarnation of a Hindu deity (as Vishnu)
2a : an incarnation in human form 2b: an embodiment (as of a concept or philosophy) often in a person
3 : a variant phase or version of a continuing basic entity
4 : an electronic image that represents and is manipulated by a computer user (as in a computer game)

While the word avatar is originally more spiritual in meaning, it is used on the internet in less spiritual activities; surfing the Web, playing games, etc. That we use avatars to represent ourselves in a world of our own creation makes us the gods of the Internet, incarnated as avatars. I will briefly describe how the use of avatars online began, followed by some examples of modern state of the art avatar technology. I will compare avatar systems of various virtual worlds that can be related to MimicMe, to see how they deal with user-avatar likeness.

3.1 A short history of avatar customization

The application Worlds Chat[1] that launched in 1995 is recognized as the first online 3D chat application. While the use of 2D images on the Internet on bulletin boards is older still, the start of 3D avatars is worth mentioning. Avatars in most computer games remained devoid of options for customization, probably because those games were single player games. This was probably due to the fact that most 3D games were First Person Shooter (FPS) games, where your avatar was only seen on-screen as the typical disembodied hand holding a wide range of weaponry. Visible in-game protagonists, on the other hand, were not customizable, since a lot of work usually went into the design of the character.

In the late nineties, the pc gaming industry was slowly shifting towards online multiplayer games. Everquest[2] (1999) (figure 3.1) was the first commercially successful Massive Multiplayer Online Roleplaying Game (MMORPG). It allowed players to alter their avatar's sex, hair style and color, and even weight. Such customizability had not been implemented in a game before. The same year as the release of Everquest, in the genre of the FPS, two big titles were released almost simultaneously; Quake 3 Arena[3]and Unreal Tournament[4]. Both games had shifted from a single player FPS game to a more multiplayer-oriented style of gameplay. Even in the much older Quake II[5] (1997), which only had a very simple multiplayer mode, users could customize the texture map to create their own style of space marine, or basically anything in the shape of a space marine. In games that relied heavily on players interacting with other players, it was logical that means to distinguish oneself from other players needed to be introduced. Even in FPS games you could now decide the look of your avatar, even if your creation was only viewed by the other players. It has been proven that applications where the avatar can be customized are more immersive. Users' online social interactions even change when using an avatar that is made to look like themselves [11]

As time progressed, more and more games introduced ways to customize avatars, but the games that included these features were almost exclusively Role Playing Games (RPGs) or FPS games with an online multiplayer mode. The reason for this is twofold; RPGs are all about the player creating a character. A wiry old wizard and a powerful fighter should not look alike, and allowing the player to customize the avatar further increases the immersion into the game. FPS games on the other hand, have customizable avatars, because it allows players to change their outward appearance in the online arena. So where RPG players become more immersed by viewing their avatars, FPS players tend to use avatar customization as a way to gain

[1] http://en.wikipedia.org/wiki/Worlds_chat
[2] http://everquest.station.sony.com/
[3] http://www.idsoftware.com/games/quake/quake3-arena/
[4] http://en.wikipedia.org/wiki/Unreal_Tournament
[5] http://www.idsoftware.com/games/quake/quake2/

individuality among the other players.

Of course, when games do not exactly fit into one category and, for instance, a 3rd-person camera position is used, the options for customization also vary. A good indicator is that the more is seen of the character by the player the more beneficial customization becomes. In single player games this is greatly influenced by how well the protagonist has been defined by the game's developer.

Virtual worlds combine the use of avatars in games and in chat rooms. Virtual worlds evolved from 3D chat applications and are, perhaps even more than games, the perfect place to create an online identity. Virtual worlds are online-only applications that users log on to using special client software or directly from the browser. They sometimes have the characteristics of a game in the way the avatar is controlled, but without a plot or characters other than the human users that control their avatars. But ultimately, most virtual worlds are still very much comparable to 3D chat rooms.

Today, avatars are probably best known from the successful MMORPG World of Warcraft[6] (2004) (figure 3.1) and the much hyped Second Life[7] (figure 3.2). New MMORPGs and virtual worlds are constantly introduced, older applications close their virtual doors, and now the fashion industry too is beginning to use avatars in their online interactions with customers. Avatars from different applications have different degrees of customization, inline with the general purpose of the application. World of Warcraft focuses less on facial and hair customization, but bases the look of the avatar mostly on the selected combination of race and class (eg. Tauren druid, Dwarven paladin, Undead mage, etc.) and the items the avatar uses that have been earned in-game. Second Life offers complete freedom to its users, allowing them to alter every aspect of their avatar and even make other items themselves. It is only logical that fashion-oriented avatars focus on looks, focusing on the face, body and clothing. Because many fashion avatars emule magazine-style photos they're often not animated, but animation is not to be confused with being able to change the avatar's pose.

MimicMe and other more static fashion avatar systems could be regarded as quite different from the general type of virtual world where your avatar is free to wander about in a 3D setting. These fashion avatars do show similarities to those used in virtual worlds because of their ability to be customized. It may be a big step from a high-resolution rendered 3D avatar to a dynamically rendered animated one, but it is a step that can be taken nonetheless. Also, fashion applications often have a community behind them as well, so whether users interact via chat or by commenting on looks or avatars makes little difference. Virtual worlds are about interaction between the users and giving users the freedom to express themselves through their avatar. One of the most important effects an avatar has on its user is that it serves the user as a way to view themselves online. Many users model

[6] www.worldofwarcraft.com
[7] www.secondlife.com

Figure 3.1: Two MMORPGs from the past and the present

(a) Everquest (1999) (b) World of Warcraft (2004)

their avatars after themselves to create a connection between them [11]. In fashion applications this is logical, because looking at the avatar should ultimately become like looking a yourself in a virtual mirror, which is why being able to customize an avatar to be a recognizable lookalike is so important.

3.2 Comparing avatar systems

To determine a course of action for improving customizability of the Mimic I look at several virtual worlds that together offer a wide range of options for avatar customization. Many of these worlds have a form of virtual currency, traded with real currency, that allows users to buy certain exclusive options for their avatar. For many sites this is a part of their business model which even allows users to profit from selling their own creations [5].

These economic aspects will not be discussed, since MimicMe has no plan as of yet to charge users for anything on the site. I will focus on the interface used to customize the avatar, as well as the range of customization offered by each system. The fact that most virtual worlds offer surroundings in which the avatar can move and that can be customized as well will also not be discussed in detail, since the focus lies on the avatar. Also, most worlds offer a choice between a male and female avatar. I will focus on female avatars, because even though MimicMe has had a functional male Mimic for a while, MimicMe centers around the female Mimic for now.

As part of my research, I have visited the following virtual worlds. These applications represent the current state of the art in avatar technology for online applications, and I look at the ways in which these avatars are created, displayed and the possible ways to customize them. The applications are ordered by launch date: *My Virtual*

Model, Second Life, There, IMVU, Stardoll, Stylezone, Virtual Me, Twinity, Frenzoo and Looklet.

I have gathered information on these virtual worlds on aspects relevant to MimicMe. Describing these aspects should make it clear where MimicMe stands in relation to these other worlds. Each of the following sub sections describes a virtual world in the following way:

Name
Launch date Not counting closed alpha/beta stages
Website
Description A short description of the application and its intended functionality.
Audience size The number of registered accounts for the application in Q2 2009 (figure 3.16)
Average audience age (figure 3.16)
Platform Which platforms are supported and what, if anything, is known about the software architecture behind it.
Graphics and style The look of the application in general. The use of two- or three-dimensional graphics. The general style used to display the avatars.
User-generated content The way(s) in which users can add their own personal touch to their avatars. Are they allowed to design, store and even sell their own creations?
Avatar customization The range of customization of the avatar offered by the application. Extra attention is paid to features that are unique to the application and/or are implemented with great effectivity. Customization of the avatar will be the focus of visiting these virtual worlds, so I study the interfaces used in the customization process of the avatar itself.
Summarized positive and negative aspects of the application The good aspects are well worth looking into to improve the existing features of the Mimic. Negative aspects that can be identified in other virtual world can hopefully be avoided in MimicMe, or at least be assigned a lower priority.

3.2.1 My Virtual Model (MVM)

Launched 1998
website www.mvm.com
Description An absolute pioneer in the online fashion industry. Launched in 1998 by retailer Land's End[8]. Users create an avatar to their likeness and try on various apparel items. My Virtual Model recently went bankrupt (November 2009). The community is still open for the time being.
Audience size unknown
Average audience age unknown

[8] www.landsend.com

Platform Browser-based with Flash.
Graphics and Style 2D renders of 3D models with a high polygon count. MVM was always very much like MimicMe in terms of graphical style. The details of the hair styles and shading techniques were always an inspiration to MimicMe.
User-generated content MVM does not have any user-created content. Almost all items on the site are for sale from various retailers. Some items are shown that cannot be tried on, but can still be purchased online.
Avatar customization While the level of avatar customization from older versions of MVM are not known, it has been observed that MVM's level of interactivity has decreased over time. Less and less angles could be chosen to view the avatar from, and options to adjust the avatar's sizes and facial features also disappeared, making it a very basic avatar. Recently, rotation of the avatar has completely disappeared. Instead, users can now upload their own photo to be used as the avatar's head instead of the original customizable head. What's left of the customization at MVM is done through some selectors that allow users to choose from options for their avatar's figure, eye size, lip thickness and skin color.
Summary of the good and the bad While not much is known about MVM's origins, it is safe to assume that a lot of their initial success in North America has been due to the fact that they were backed from the start by a large clothing retailer[9]. This meant that the users instantly have something real to connect their online experience with. The decrease in features could be caused by aiming to reach an even broader audience. Less features means that users start shopping sooner. It also means less development is needed to add new clothing.

3.2.2 Second Life

Launched June 2003
Website www.secondlife.com
Description The best known virtual world, partially due to extensive media coverage late 2006. Second Life is a typical virtual world that allows its users to traverse a vast 3D landscape populated by all the other users' avatars. Second Life has one of the most advanced in-world content creation tools, as well as a huge economy.
Audience size 19M
Average audience age 33
Platform The Second Life client software is available for Windows, Mac OS X and most Linux distributions. It is open source, allowing users to create their own versions of the client.
Graphics and Style 3D animated avatars with a low polygon count, rendered in real time. The basic Second Life avatars are very nondescript and all look somewhat sportive and healthy.
User-generated content This is where Second Life really shines. Users can create

[9] www.landsend.com

Figure 3.2: Second Life

(a) Altering your avatar's body (b) Altering your avatar's face (c) Exploring Second Life

anything they want; from a car to a pair of pants to an alternate appearance for the avatar, way beyond the avatar's default appearance.

Avatar customization If the user does not want to go all out and create an elaborate custom avatar, there is still a plethora of options for the default avatar, mostly organized in a large collection of sliders in various categories.

Summary of the good and the bad Second life avatars are customizable in every way. Unique among other avatar systems is the possibility to adjust the hair, which consists of several meshes with transparent textures, so that the user can alter the length and size of each part of the hair separately. The ability to create everything yourself, from earrings to a complete –and often much better looking– skin texture easily makes the Second Life avatar the most customizable of all.

Unfortunately, the vast amount of sliders and the time required to delve into modeling your own avatar in detail are also Second Life's main drawback. This is perhaps most clearly reflected in the average age of Second Life's users and is a key point in user-friendly avatar design.

3.2.3 There

Launched October 2003 (closed March 2010)

Website www.there.com

Description There was a virtual world with a distinctively friendly feel to it. The 3D world was organized around a group of islands. A content creation system was in place, as well as a small economy. There eventually closed its website and 3D world[10].

Audience size 2.5M

Average audience age 22

Platform Client software for Windows, although an alpha version for Mac OS X was being created.

Graphics and Style 3D animated avatars with a low polygon count, rendered in

[10] http://www.there.com/info/announcement

Figure 3.3: Different emotions in There

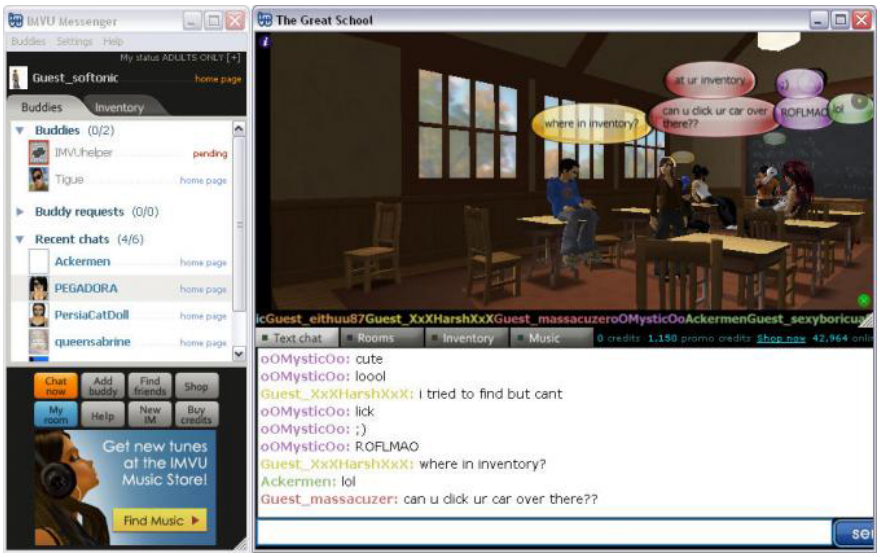
real time. There avatars have a cell-shaded cartoon feel to them.
User-generated content Not known
Avatar customization Avatar customization is basic, on par with the average virtual world. There's avatars can show an impressive range of emotions that give them an edge over other avatar systems. (Fig. 3.3)

Summary of the good and the bad The overall friendly tropical holiday island feeling makes There a fun world to be in. The well-implemented emotional responses make interaction improve the experience of interacting with other. However, this means that the cartoony style of There must suit the user, because while appearances can be changed, the style remains the same.

3.2.4 IMVU

Launched Q2 2004
Website www.imvu.com
Description A 3D instant messaging (IM) client. Avatars can be dressed by the user, and have their own rooms which can be decorated with furniture and accessories.
Audience size 35M and growing fast, likely due to an aggressive online campaign with alluring advertisements with slogans like "dress-up for grown-ups." (Fig. 3.4)

Figure 3.4: IMVU



(a) An IMVU chat room

(b) One of IMVU's alluring banners

Average audience age 19
Platform Website and Windows-only client software.
Graphics and Style 3D avatars with a low polygon count, rendered in real time. Avatars are not animated, but instead can be positioned by clicking on chairs and the like. The IMVU avatars are best described as sexualized caricatures, with big heads, big eyes, and big –uhm, you know... ears.
User-generated content IMVU has its own economy and marketplace, where users that have invested enough time or real currency can create content using various 3rd party programs, like Autodesk 3D Studio Max or Blender[11].
Avatar customization Only the basic facial features and skin tone are adjustable.
Summary of the good and the bad Sex sells, and IMVU's accessible and sexy avatars draw lots of new users to the community. As an IM client, IMVU is very sophisticated and caters to its users every need with customizable, re-sellable items and a well-implemented user interface.

The downside is that, as with There, everyone still looks more or less the same and that individuality can only really be achieved through one's outfit and accessories; not through a unique face.

[11] http://www.blender.org/

3.2.5 Stardoll

Launched Q1 2005
Website www.stardoll.com
Description An online community that combines fashion and celebrities into an avatar system in the style of paper dress-up dolls.
Audience size 34M
Average audience age 14
Platform Browser-based with Flash.
Graphics and Style Stardoll has a unique style that distinguishes itself from other virtual worlds and fashion applications because of the use of 2D graphics. The avatar is only visible from the front and its features resemble photographic material only with the number of colors reduced to simplify the appearance. The result is a very human-looking cell-shaded 'doll.' Stardoll has drawn attention from celebrities such as Avril Lavigne who are represented on the site[12] (Fig. 3.6), and whose styles can be used as inspiration for young fashion designers-to-be[13].
User-generated content Users can create their own versions of clothing by arranging textures and patterns and by choosing colors.
Avatar customization Avatars' bodies are customizable around the hips and bust, both in three sizes. The face is where Stardoll's combination of 2D graphics and restyled photographic material give good results. A wide range of eyes, noses, hairstyles, eyebrows and mouths, with different colors where applicable, allow for a lot of possibilities. Whether or not everyone can create an avatar that closely resembles herself remains to be seen though.
Summary of the good and the bad For its intended purpose, Stardoll's avatars are excellent and allow for a wide range of customization. With the unique 2D style comes the drawback that the avatar cannot be viewed from any other angle.

3.2.6 Stylezone

Launched August 2006
Website www.stylezone.com
Description A seemingly abandoned online fashion community with an impressive piece of software at its core. Stylezone is made by Browzwear[14], founded in 1999. Browzwear appears to be a mayor player in the online fashion industry on a business-to-business level and offers fashion design software for modeling clothing in 3D.
Audience size unknown
Average audience age unknown
Platform Client software for Windows.
Graphics and Style 3D avatars with a very high polygon count, rendered in real

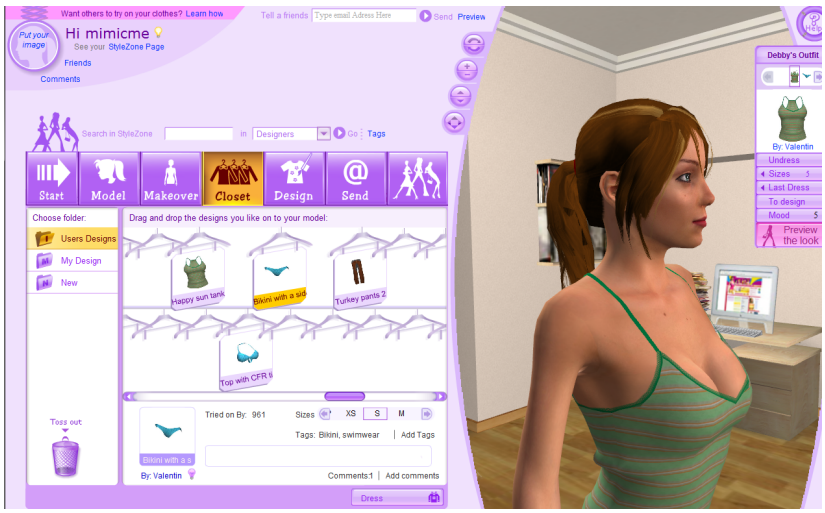
[12] http://www.dmwmedia.com/news/2007/04/23/avril-lavigne-posts-first-artist-page-on-teen-web-community
[13] http://www.dmwmedia.com/news/2007/04/18/millennials-conference-stardoll-to-start-selling-virtual-cl
[14] www.browzwear.com

Figure 3.5: Creating an avatar in Stardoll

(a) (b)

Figure 3.6: Avril Lavigne on Stardoll

Figure 3.7: Stylezone



time. While not animated, the facial expression of the avatar can be posed. When trying on items, the body can also be posed seven different ways. Stylezone aims for a realistic look, with extremely well-detailed skin and clothing textures. Stylezone uses an advanced cloth modeling system to render clothes on the avatar.

User-generated content Stylezone's users are encouraged to design their own clothing using the Stylezone client. These items are then shared on the Stylezone site.

Avatar customization Both the avatar's face and body are extremely customizable. Each slider has a descriptive animation above it which highlights the part of the body that it changes. There aren't many hairstyles to choose from, but there is a wide range of colors to choose from, along with separately colored highlights.

Several options exist to modify the avatar's posture, as well as lots of options to define the breasts. Several sliders exist that alter the shape of the breasts to simulate the effects of various types of bras (sports, push-up, pointed).

Summary of the good and the bad Next to the many options for adjusting the avatar's face and body, Stylezone also uses cloth modeling to render fitting clothing on the avatar. Various different size tables are used for pants, skirts, shirts, etc. Selecting jeans that are one or two sizes larger, for instance not, only makes them longer, but also makes the denim show more creases around the knees and even causes the waist to hang slightly lower on the hips, due to its increased size.

Apart from being able to freely rotate the avatar, and the fact that it is rendered in real-time, Stylezone is very similar to MimicMe in its presentation. Stylezone does suffer from excessive use of sliders to model the avatar. Also, the minimum and maximum boundaries for the sliders are influenced by other sliders in a way that is opaque to the user. however, the impressive results outweigh the negative aspects in Stylezone's case.

Figure 3.8: Stylezone's real-time rendered fitting of clothing

(a) The red lines move around until while computing the fit

(b) The result

3.2.7 Virtual Me

Launched April 2007
Website discontinued
Description A 3D avatar studio intended for an online version of the Big Brother television program. It was created together by game distributor Electronic Arts[15] and Endemol[16], creator of the big brother television format, but was never used. Users can create and dress an avatar. Regrettably, the software has all but disappeared entirely from the Internet, except for an obscure Polish promotional video on youtube[17].
Audience size N/A
Average audience age N/A
Platform Client software for windows. No online features whatsoever.
Graphics and Style 3D animated avatars with a high polygon count. Though the graphics are of lesser quality than Stylezone, Virtual Me has a similar realistic de-

[15] www.ea.com
[16] http://kotaku.com/252484/ea-and-endemol-announce-virtual-me
[17] http://www.youtube.com/watch?v=9BNKZRXFpoQ

Figure 3.9: In Stylezone, the differences between different sizes of the same garment are actually visible

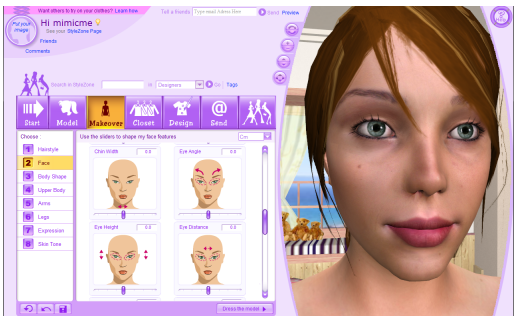
(a) Top: Large

(b) Top: Small

(c) Jeans: size 34

(d) Jeans: size 30

Figure 3.10: Customizing the face in Stylezone



(a) Basic features for face customization

(b) Combining various facial traits to create an expression

Figure 3.11: Stylezone's *body type* slider

(a) Normal body mass

(b) Heavy body type

piction of the human form. Because it was originally intended for live interaction, Virtual Me avatars have some animations that can be used, albeit with a slightly goofy expression on its face.

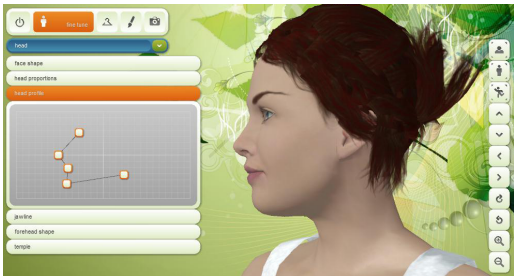
User-generated content Virtual Me has no means for users to create content, and does not use an Internet connection for anything.

Avatar customization Virtual Me's avatar customization is extensive and makes use of some very original concepts. At the most basic there are options, represented by one-dimensional sliders to alter height or width of various features like the nose, mouth and eyes. A two-dimensional grid structure is used to define more complicated or composite features. The height and girth have been combined in such a grid, where the user can place a single point. A much more inventive use of the grid is when multiple points, connected through a single line, have to be arranged to form the avatar's profile or jaw line. In shaping the profile, points represent things as the brow, tip of the nose and chin among others. Users are also given a choice from various typical nose, eye and mouth shapes to use for their avatar.

Summary of the good and the bad Virtual Me's options for avatar customiza-

Figure 3.12: Virtual Me's inventive composite interfaces for altering the avatar's face shape

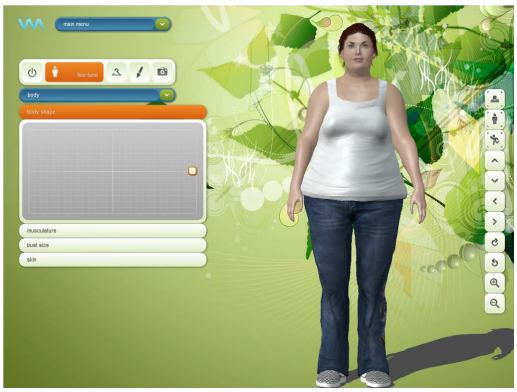
(a) Jawline



(b) Facial profile

Figure 3.13: Virtual Me's *body shape* grid; the y-axis controls the avatar's height

(a) Slim posture



(b) Wide posture

tion are slightly less detailed than those of Stylezone, but the unique use of two-dimensional objects to help define the avatar's features make Virtual Me worth studying. If four to six sliders can be condensed into a single grid, the user can save a lot of time in creating an avatar. The grid should indicate more clearly which point affects which attribute though, and having the line and points form a shape that is recognizable as a facial profile would help here.

Similar to Stylezone, it is not clear what the effect on the sliders and grids is, when the user picks a standard shape for the eyes, nose or mouth. Whether or not this kind of opacity helps the user or not is a point worth discussing.

3.2.8 Twinity

Launched Q4 2008
Website `www.twinity.com`
Description Twinity creates virtual versions of real cities. First Berlin, then London,

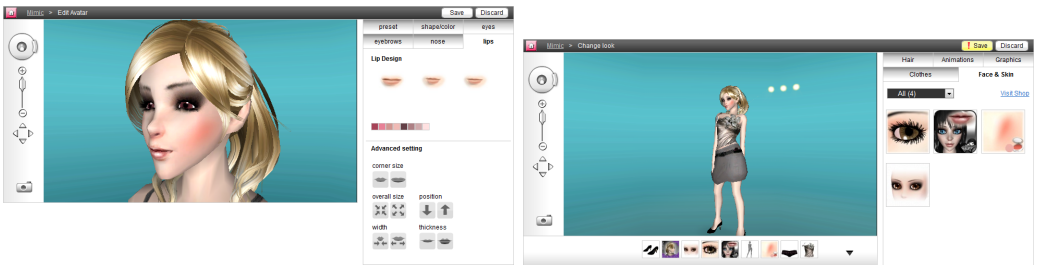
and now Singapore as well have been recreated virtually. Much like in Second Life, users can buy or create their own clothing and furnish their own virtual apartment.
Audience size <1M
Average audience age 36
Platform Client software for Windows only.
Graphics and Style 3D animated avatars with a low polygon count, rendered in real time. Twinity's avatars have a very basic feel to them, which fits the realistic style of Twinity's cities.
User-generated content Users can create their own clothing and furnishing, but because everything apart from clothing stays within the confines of one's own virtual home, the streets in Twinity looks a lot more organized than in Second Life.
Avatar customization Avatar customization in Twinity is fairy basic, with one exception: Twinity makes use of sophisticated software that users can user to create their face in Twinity from a photograph. The software in question is a third party package called FaceGen[18] which will be examined for possible use in combination with MimicMe in chapter 4. Users can upload a maximum of three photos (front, profile and partially turned), and mark key points like the tip of the nose and corners of the eyes. Twinity uses this information to let FaceGen generate a fully textured mesh of the face.
Summary of the good and the bad The use of FaceGen seems like an excellent way to make a low-poly avatar resemble its user. It could be argued that it is too much software for such simple avatar, but the result is effective nonetheless.

3.2.9 Frenzoo

Launched Q4 2008
Website www.frenzoo.com
Description A relatively new avatar studio on the market. Built along the same principles as IMVU.
Audience size <1M
Average audience age 16
Platform Browser-based. Frenzoo uses the Unity Web Player browser plugin[19], which runs on Mac OS X and Windows.
Graphics and Style 3D animated avatars with a medium polygon count, rendered in real time. The avatars cannot be controlled, but can be posed or made to execute animations. The graphical style of Frenzoo was initially a lot like stylized cartoony faces from Japanese anime, but has since moved more towards the style that can be seen in IMVU. In contrast to IMVU, Frenzoo focuses more on being a fantasy-fashion community.
User-generated content Users can create their own clothing using patterns and

[18] www.facegen.com
[19] http://knol.google.com/k/review-of-unity-3d-web-platform-why-it-was-selected-for-frenzoo-com

Figure 3.14: Frenzoo



(a) Altering the face (b) Miscellaneous body parts

images. In addition, users with the necessary skills can also import models from Autodesk 3D Studio Max[20].

Avatar customization Much like in IMVU, the avatar's body proportions aren't adjustable. The face area has all the standard customizability. Comparable to Second Life, face attributes, such as makeup and eyelashes can be added as custom items.

Summary of the good and the bad Frenzoo is trying to catch up to IMVU, so it shares many of the latter's pros and cons.

3.2.10 Looklet

Launched Q3 2009

Website www.looklet.com

Description An online fashion application that rapidly became popular due to its excellent design and high-quality photographic clothing and models.

Audience size Unknown

Average audience age Unknown

Platform Browser and Flash.

Graphics and Style Looklet's images look like they come straight out of a fashion glossy. The combined use of professional models, a single striking pose and fashionable clothing make every result look like a professional photograph.

User-generated content Users cannot create their own clothing, but can share their looks on the website.

Avatar customization The only thing that can be changed about looklet's avatar is the model's face, which also affects the skin color. Most models have more than one facial expression to choose from.

Summary of the good and the bad It is with good reason that Looklet has shot straight to the top of fashion communities. The entire site oozes style and every item the model wears instantly looks great.

[20] www.autodesk.com

But as an avatar, Looklet's model is very limited, because users can only select models who already have their own faces. Looklet achieves the idea of a virtual mirror only partially though the use of photographs, while most users may have trouble identifying themselves with the professional model on the other side. The trade-off between looks and customizability is central to the way users experience a product.

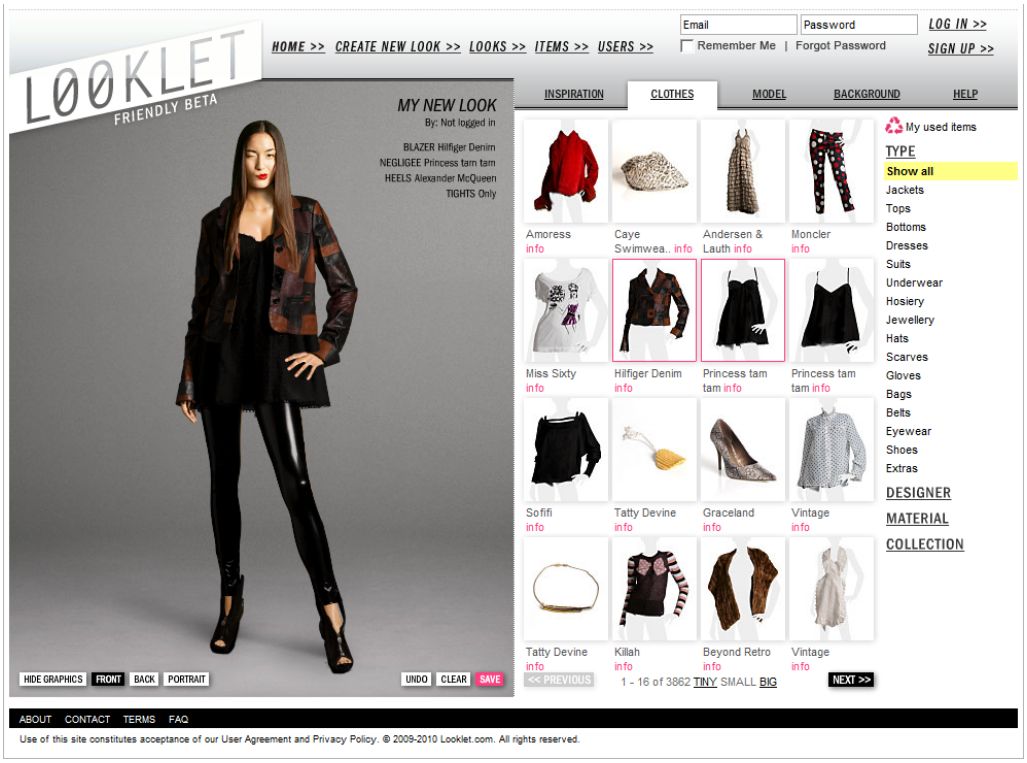
3.3 Summary

It came as a surprise that MimicMe's goal of letting users recognize themselves in their avatar is more unique than was originally believed, because many applications use a more stylized type of avatar. What becomes clear is that a highly customizable avatar always means that a lot of settings are involved, often in the form of a long list of sliders for all the features. While an application like Virtual Me attempts to group these sliders into a cleverly designed grid, the input is still different from a drag-and-drop application like Stardoll. It is likely that an application's chosen visual style is a major influence on the type of audience it attracts.

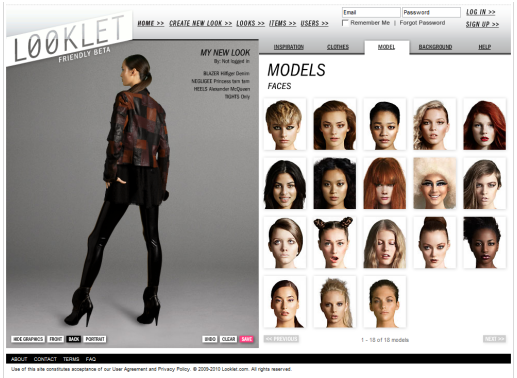
KZero[21] is a consulting company specializing in virtual worlds which originally started out as a blog. KZero's mapping of virtual worlds by age and sector have been very useful (Fig. 3.16, Fig. 3.17). Many more virtual worlds are shown on KZero's charts that also target very young audiences. It is understandable that many of these virtual worlds don't have any notable avatar system in place or 3D surroundings that can be explored, and therefore fall outside the scope of this thesis. Of course, there are many more virtual worlds. The ones selected were most relevant to my research because of their focus on a combination of fashion, realistic avatars and communities. In the next chapter, the information gathered here is used to evaluate MimicMe and find ways in which to improve and expand the functionalities of the Mimic, using strong points and perceived pitfalls of the applications that have been discussed as a guide.

[21] www.kzero.co.uk

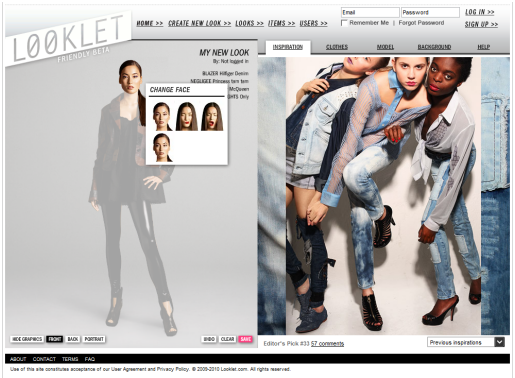
Figure 3.15: Creating looks in Looklet



(a) Front view / Clothing



(b) Back view / The available models



(c) Choosing a model's expression / Weekly inspiring mood board

Figure 3.16: KZero: The number of registered accounts per virtual world in Q2 2009. The angular position indicates the average age of the users, as determined by KZero. The radius shows the time each virtual world has been in existence.

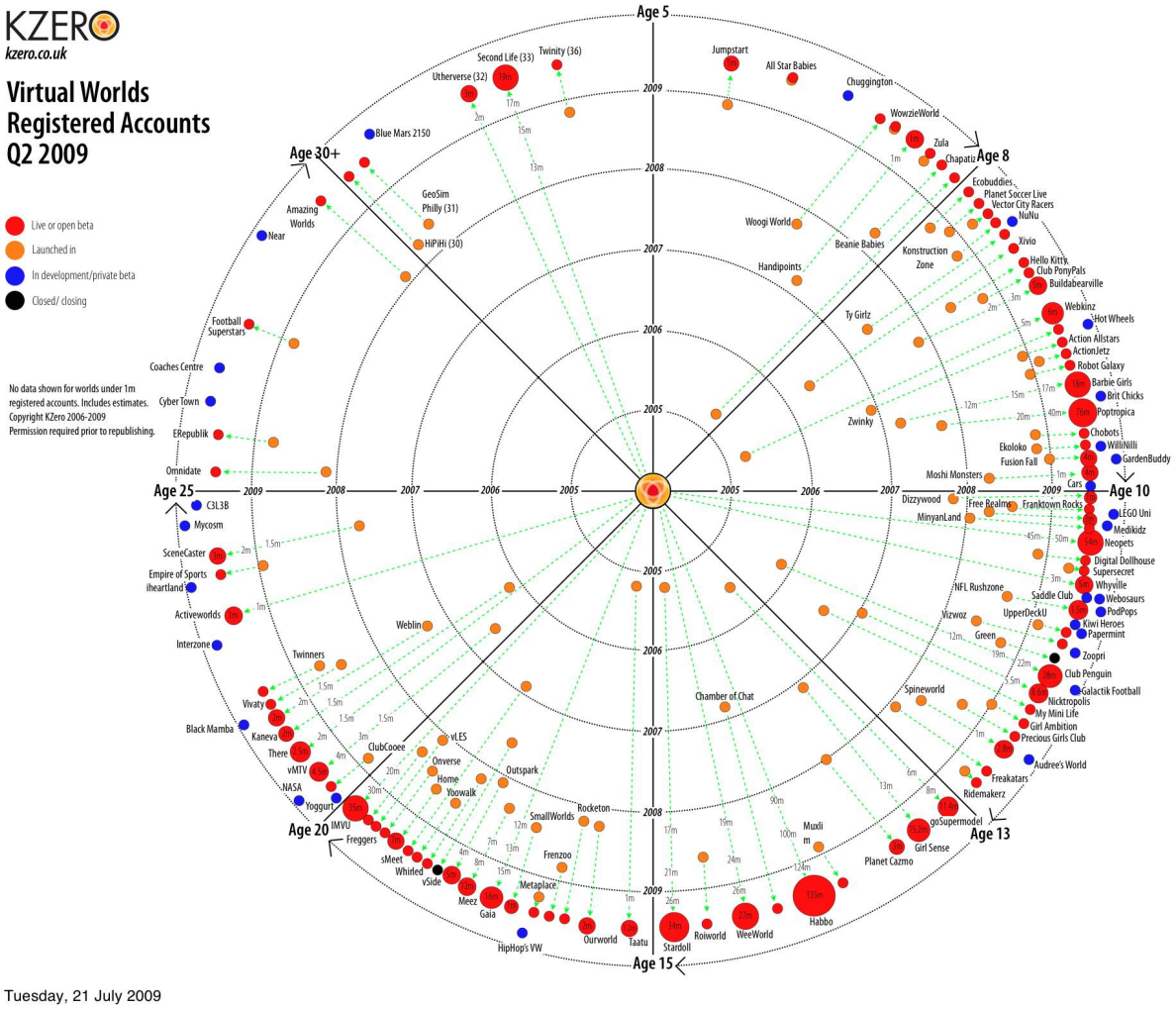
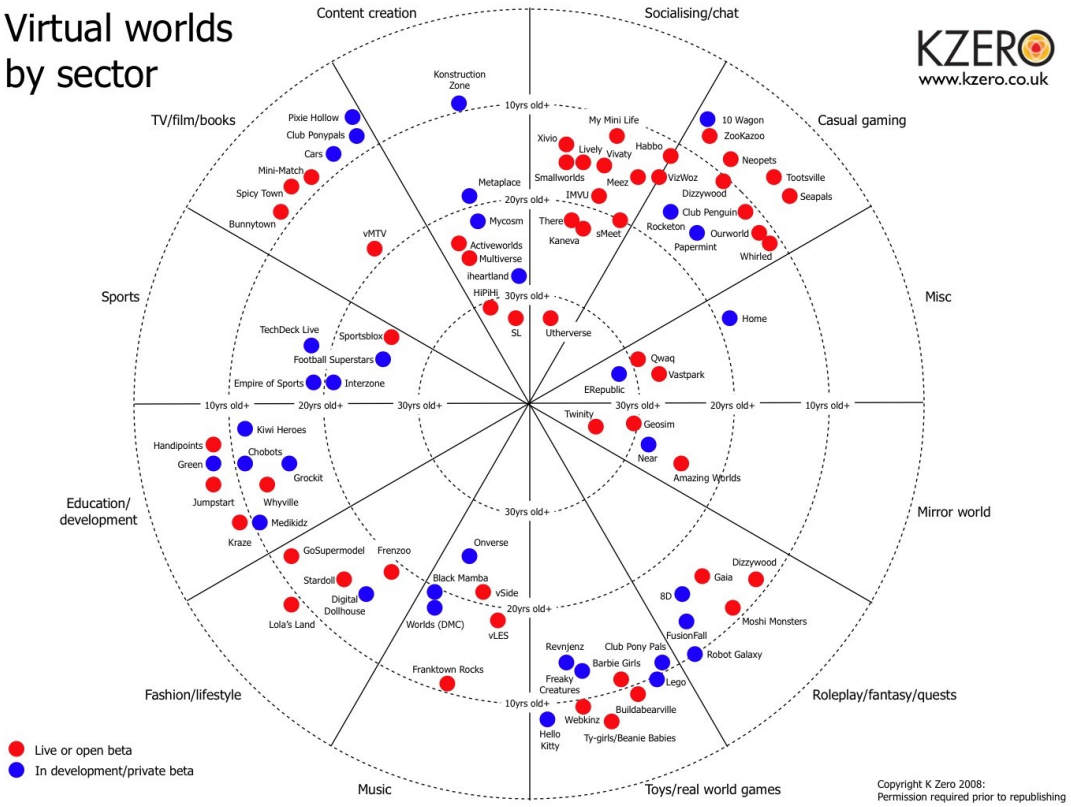


Figure 3.17: KZero: Virtual worlds grouped by sector/subject in late 2008. The radius is used to show the average age of the target demographic.



Chapter 4

Meeting MimicMe's design requirements

This chapter formalizes the requirements of MimicMe. I use the information from chapter 3, discuss relevant literature and explore the possibility of implementing a process which allows users to uses submitted photographs the Mimic's face in order to plan a course of action for further development of the Mimic.

4.1 Requirements

The requirements in this section have been divided into two categories. User requirements describe what functionalities MimicMe's users want to see in the Mimic, while technical requirements deal with what MimicMe's systems have to offer.

4.1.1 User requirements

The user requirements primarily revolve around the trade-off between having a complex interface and a high level of realism, and having an simple interface and an avatar with less options for customization. This can be illustrated by looking at Second Life's avatar, where a multitude of options is available to the user that can be used to alter the smallest details so that as many different faces as possible can be made. An avatar like Frenzoo, on the other hand, has significantly less options available to alter the face, and none whatsoever to alter the body, resulting in avatars that might look different, but are clearly all based on the same basic model. It can be observed, that the avatar systems with the least amount of options for customization are also the ones that have the most stylized avatars. This can also be attributed to the assumption that applications design their avatars to match the wishes of their target audience; younger, or more casual audiences appreciate a stylized avatar more

than virtual world fans that want more freedom and options for their avatar. Key to all virtual environments is interaction. Being able to see the Mimic from six different angles increases the level of interaction compared to Stardoll for instance, which only giver users a frontal view of their avatar. Virtual worlds where users can control their avatar like a game character (There, Twinity, Second Life) offer even more interaction. More fashion-oriented applications like Stylezone allow a choice between a number of poses for the avatar.

MimicMe aims for realism and mainly targets adult women, so it seems right that MimicMe takes the realistic approach to avatars. Users need a recognizable avatar, because the concept of a virtual mirror is very important to MimicMe. This avatar should be adequately recognizable to the user herself, but also to her friends who see images of the user's Mimic. Finding out what makes humans recognize each other is the subject of various scientific papers that will be used in section 4.2.

Because MimicMe is all about clothing, it must allow users to easily switch clothing items and rapidly try on as many combinations of clothing as they like. Combining outfits is the most important feature of MimicMe's shop section. Users may have several garments at home that they want to combine with virtual garments in the shop, so having a wide range of items available could allow the user to find the item she has at home in MimicMe's shop section and allow her to combine it with various other items in order to find suitable additions to her current (physical) outfit.

Lastly, a feature that has been requested the most in MimicMe's user surveys is the use of real measurements for both the Mimic and the clothing. Ideally, users want to enter their sizes and have MimicMe calculate and render a Mimic of which the body matches the user's as close as possible. Users want to know if a pair of pants in their size fits the way they expect.

4.1.2 Technical Requirements

One of the main goals for MimicMe as a product is that its users should not have to download separate plugins to their computer and that it should not be cpu-intensive for its users to render Mimic. This makes the application more accessible for mobile browsers that might not even support Flash and for users with low-end systems. Each time a user changes her Mimic, MimicMe performs a series of complex 3D operations. The main point is that the user does not need to worry about this.

As mentioned in chapter 2, to meet these goals, MimicMe has been designed so that every browser can display the application. All images of Mimics are rendered server-side and transmitted to the user as .png files, though a transition is being made to the use of .png files. If the render server's load becomes too high, more servers could be used to balance the load. However, a recent test showed that The server can easily render over 100 Mimics per second, provided that all 3D data is present in the server's working memory, so speed is not yet a problem.

Another important user requirement poses a greater challenge: displaying accurate sizes. This problem has two sides. Firstly, users want to be able to enter their sizes and see an accurate representation of themselves. Secondly, users want to be able to choose different sizes of clothing and see how that particular size fits their Mimic. The fact that fashion manufacturers each use their own proprietary size tables, combined with the skewed size numbers for women's clothing due to so-called vanity sizing –where clothing labels have been showing smaller sizes over time for the same measured size clothing, to make buyers feel thinner– make it very hard for MimicMe to implement a variable sizing system for both Mimic sizes and clothing sizes.

From a technical perspective, to render clothing in various sizes that fit accurately, the way the Mimic is rendered would have to be changed drastically. Currently, when the Mimic's body properties change, clothing on the Mimic is scaled to match. More accurately, this scaling takes place after the clothing is applied to the Mimic, thus applying to both clothing and Mimic at the same time. If a Mimic gets longer legs or narrower hips, the fit of a pair of jeans would not be affected. Realistic fitting of clothing on an avatar that can vary greatly in size is effectively done by Stylezone (Fig. 3.9, section 3.2.6). This way of rendering clothing on the fly requires significantly more processing power and has to be applied on specifically designed and patterned clothing items, which is exactly what Stylezone's creator Browzwear sells.

4.2 Developing the Mimic

While keeping MimicMe's requirements in mind, the Mimic needed to be developed further. I have used ideas from the virtual worlds visited in chapter 3, along with information taken from literature from the fields of anthropometry and machine vision. Several techniques used in the studied avatar systems could be added to MimicMe as well, as long as they do not conflict with MimicMe's requirements. Lastly, as a different approach to the use of only deformation vectors, I evaluate several third party software packages that deal with modeling human faces from photographs onto 3D representations.

4.2.1 Literature on facial recognition and anthropometry

Two fields of study that provide much information relevant to the creation of recognizable avatars are anthropometry and machine vision. Anthropometry, which measures variance in human shapes is used in the fashion industry for sizing, and to describe human shapes in general. Machine vision, which deals with human shapes, specifically human faces, takes valuable insights from what makes the human visual system work and apply that knowledge to face recognition [6]. An excellent summary of key points in machine vision is given by Pawan Sinha in [10]. This knowledge can

then also be used to determine what features have the greatest effect on an avatar's recognizability.

Some key points pertaining to the recognition of a face are:

- The human visual system is capable of recognizing faces even from severely degraded images. The tolerance for such degradation increases with familiarity to the person.
- The eyebrows play the biggest role in recognizing a face.
- The face plays a much larger role in recognition than the body and posture.
- Scaling a face vertically or horizontally does not impair recognition.
- Pigmentation of the face plays a role in recognition along with shape, and can help if the face shape is not sufficiently detailed.

The Civilian American and European Surface Anthropometry Resource Project (CAESAR Project)[1] created 3D full-body scans from 2.400 North American and 2.000 European civilians. These scans contain raw vertex data that has been used extensively in scientific research. Principal Component Analysis (PCA) is used on the extremely detailed 3D data in [2] and [1], which reduces the number of variances by grouping them into eigenvectors, each of which combine a group of correlated variances of the entire set, and therefore the entire range of human shapes. These eigenvectors can be recognized as typical human variances and named appropriately, like height, weight, posture, nose length, mouth height, etc.

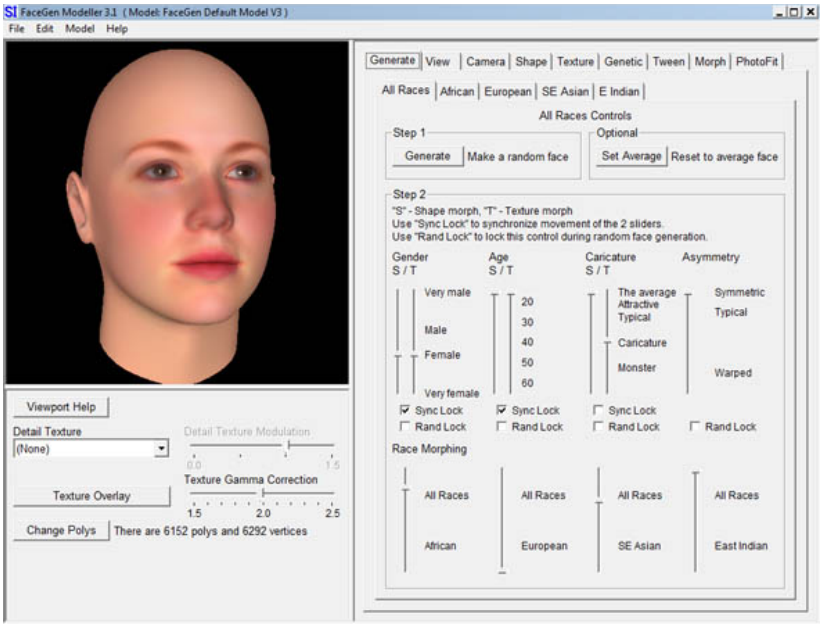
We visited TNO's Human Factors department, located in Soesterberg, where this kind of research on human shapes is being used in the design of equipment for the department of defense. We learned that by using only 50 eigenvectors, 95% of the human shapes in the set can be created. Approximately half of these eigenvectors are located in the head. Unfortunately, this research is proprietary and costly to obtain. Such a complete set, with animations and names for each of the eigenvectors could be considered the holy grail of parameterized human shapes. Blanz and Vetter [4] use a similar approach on a data set from Cyberware[2], focusing entirely on the face. Using PCA, they parameterize the human face to create a morphable model, which can be overlaid over photographic material to alter faces.

4.2.2 Facial texturing for MimicMe

Mainly because of the excellent results that can be observed at Twinity, MimicMe wanted to explore the possibilities of allowing a user to upload a photo of her own face to be used as the face for her Mimic. Users would upload photos taken from the front,

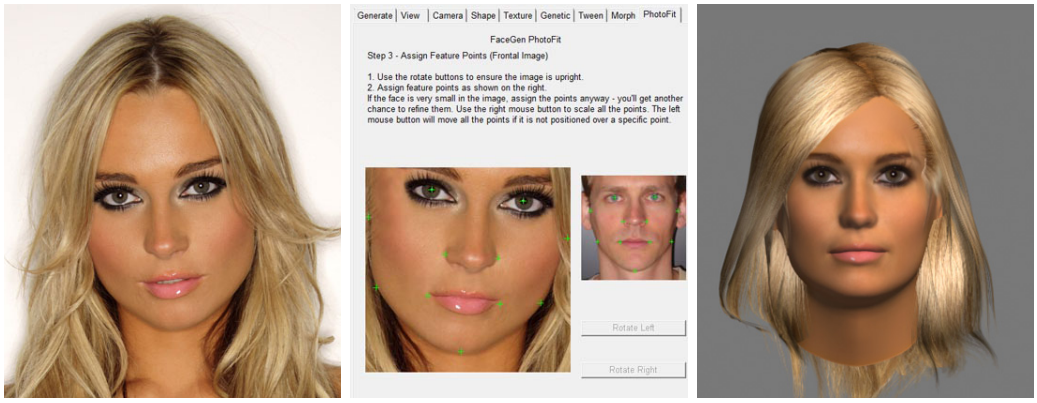
[1] store.sae.org/caesar/
[2] www.cyberware.com/

Figure 4.1: The FaceGen modeler



and possibly even the side of their head. After modifying them these photos would then be used as a texture on the Mimic instead of the Mimic's default texture. This presents a very different approach to recognizability of the Mimic. Where MimicMe now relied solely on altering the face shape using deformation vectors, we would now examine alteration of the face texture, because facial recognition relies on both surface reflections and 3D shapes [9]. Ultimately, three-dimensional shape and two-dimensional reflection should be joined to achieve the best result, but we had to start by focusing on face textures alone. This goes beyond the mere color variations that are possible with MimicMe's standard textures and their adjustments for lightening or darkening and includes eye color and eyebrows for instance. Instead of developing this software in-house at MimicMe, there's also the option of using a third party software package which allows users to upload photographs of themselves to be used as face textures. We decided to explore this option before we started development of such a feature ourselves. The most important requirement here is that the software can be integrated into the current MimicMe systems and that the process should be automatic. Possibly, the users will have to indicate landmarks on their submitted photographs, like the corners of the eyes and mouth. Of course there is also the cost-benefit ratio that plays a role.

Figure 4.2: Creating a face from a photo in FaceGen



(a) A photo of a face

(b) Placing the required landmarks on the photo

(c) The resulting face. The hair is not modeled in FaceGen, but imported

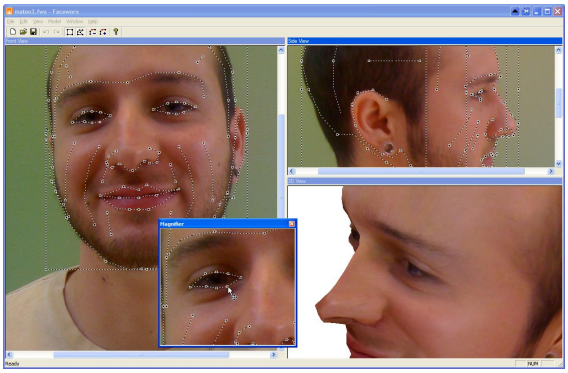
FaceGen

Facegen[3], by Singular Inversions[4], is well-known in the game industry. Several games utilize FaceGen to allow the player to customize their character's faces. Most recently, FaceGen was used in Fallout 3[5], by Bethesda Softworks[6]. Facegen is based on Blanz and Vetter's work on morphable face models [4]. The FaceGen SDK contains an API which could allow users to upload photo's to the FaceGen application running on MimicMe's servers. The results obtained with a test version of just the FaceGen Modeler are impressive, as can be seen on the FaceGen website[7] and in a user's experiment, pictured in figure 4.2.

One drawback is that FaceGen only focuses on the face. The ears are part of the default head mesh and are also not textured. This isn't a big problem since The Mimic almost always has hair covering the side of it head. As mentioned before, Twinity uses FaceGen in much the same way. In Twinity, the process of calculating the FaceGen face configuration, after uploading photos and selecting the appropriate points takes around 15 minutes. It's not known whether or not the relatively much higher polygon count of the Mimic means that rendering faces for the Mimic takes that much longer, but I suspect it doesn't, since all calculation are done to extract the parameters from the photo images, not to render them. Unfortunately, the FaceGen SDK, which we need if we want to integrate FaceGen into MimicMe, costs 14.000 USD; a very high price to pay for a startup company. The high price and the amount

[3] www.facegen.com
[4] http://www.facegen.com/about.htm
[5] fallout.bethsoft.com
[6] http://www.bethsoft.com/
[7] http://www.facegen.com/photofit_samples.htm

Figure 4.3: Creating a face from a photo in Faceworx



(a) Adding lines that somehow make the photo better fit the generated 3D mesh. Faceworx is lacking official documentation on how to best achieve this.

(b) The result

of work needed to integrate it into the MimicMe software make FaceGen unsuitable at this time for use in MimicMe.

Faceworx

Faceworx is a piece of software developed by Looxis[8] for use in a process that converts photo portraits into 3D faces which are then etched into glass in 3D using a laser. The software itself is easy and intuitive to work with at a basic level but has no API whatsoever, so all work must be done manually. Also, the quality of the 3D face greatly depends on the amount of detail added in Faceworx, as can be seen in figure 4.3. Looxis has donated the Faceworx program code to the MakeHuman project[9], an open source program aimed at creating 3D humans from different ethnicities, which also uses parameterization of the human form in their project [3]. While Faceworx does a better job than FaceGen on the parts of the head beyond the face, its main drawback is still that it lacks an API, making integration into a separate system like MimicMe impossible.

4.2.3 Summary of evaluated face texturing software

While using existing software for face texturing could help MimicMe achieve recognizability, this depends on the features offered by such a package. FaceGen and Faceworx each have their strengths and weaknesses.

[8] http://www.looxis.com/
[9] http://www.makehuman.org

FaceGen

+ extensive set of around 130 parameters for face modeling, including complex settings fr age, race and gender
+ possible to integrate into an existing system, like the Mimic
– there's a lot of work to integrate such a complex system
– only generates the face. The rest of the head remains the same
– expensive. Licenses cost 300 USD to publish generates images, 1.000 USD to use exported 3D models in commercial software, and 14.000 USD for the SDK.

Faceworx

+ both the front and sides of the head are modeled, provided that a photo taken from the side is used
+ more unusual head shapes are possible that with FaceGen
– no API. Faceworx does export to Waveform OBJ however.
– too simplistic to accurately map key points in textures like the eyes to the mesh

Evaluation

After evaluating FaceGen and Faceworx, we decided that between FaceGen's high costs and Faceworx's limited functionality there was no feasible option to use any of the two packages with MimicMe. The next course of action was to create our own tool that allows us to use custom facial texturing in MimicMe. Because we were going to develop it ourselves, this meant that we could start small and expand functionality from there on. We would start by finding a way to prepare a portrait photo in such a way that that photo more or less aligns with the facial texture of the Mimic and then create an interface that allows photo's to be aligned this way via the MimicMe website.

Chapter 5

Results

This chapter describes the development done during my internship. Some of the plans for concepts used with the Mimic had already been set in motion when I started working at MimicMe, like the use of deformation vectors for customizing the face, while other concepts, like blend values that alter both body shape and clothes accordingly had been used for a long time before. While my primary focus was development of the Mimic, I have also developed the MimicMe website and made an early prototype for uploading a photo as the Mimic's face.

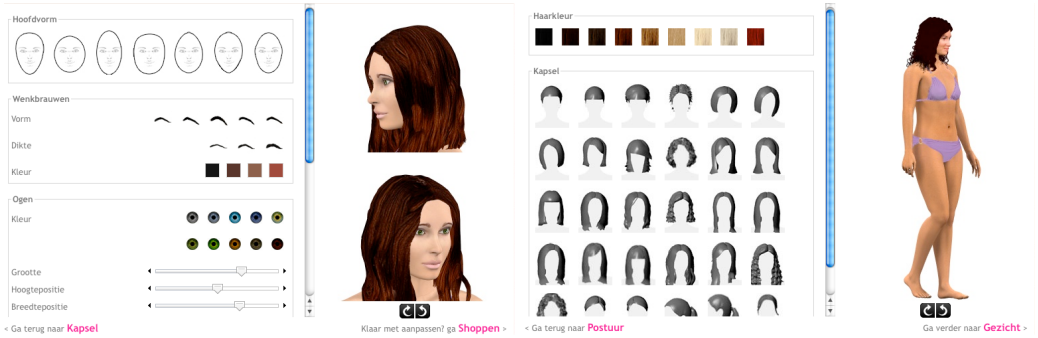
When I started my internship, I first familiarized myself with the systems used by making minor adjustments to MimicMe's PHP code, like browser detection. After some study of the site's design and the Model-View-Controller design pattern, I had a better picture of MimicMe's structure. Working on alterations in MimicMe's code, be it PHP, JavaScript or C, as used in the render server, proved challenging because of the lack of commentary in the code; a side-effect of the rapidly changing nature of MimicMe.

5.1 Development of the Mimic

Work on the MimicMe began with implementing the facial deform vectors which had just been created. These deforms were the second form of customization available after the blend values, that were used on the Mimic's body. Many new deforms were added over time, in order to expand the range of customization in the face. I introduced a more uniform way of naming the deform vectors in order to allow for easy naming of new deforms.

After MimicMe acquired a digital male model, named Eric, we implemented this model for use next to the existing female model, Amber. Because the model's head looked much too hulky for a fashion model, we had to alter the default values of the newly created deform vectors. This was an excellent way to test the new, more

Figure 5.1: Customizing the Mimic's head and hair



(a) Altering the Mimic's head shape and facial features (not possible when the user's customcustom face texture)
(b) Choosing a hair style and color (shown with face texture is enabled)

modular architecture of MimicMe's back-end. We successfully tested Eric's separate set of deform vectors, many of which were centered around the jaw and chin. Although these vectors were initially deemed essential to the male model, we then implemented a similar set of vectors for Amber, because while essential to the male model, the female face also turned out to benefit from customizability in this area. Further development of the male model was postponed in order to focus more on the female model, since MimicMe's intended audience was still mostly female. I did create a complete set of sizes, that could be used to fit tailored suits to the male Mimic in the future, but these were never implemented. As the render quality of Amber improved, we decided to remove the somewhat underdeveloped Eric from the site because he looked out of place, and no new clothes were being modeled for the model.

For the Mimic's hair, we created a standard set of colors. Eventually, every hairstyle came in the same set of colors, which meant less confusion for the users. New hairstyles were continuously added, but individual hair styles remained static.

After reading articles about choosing a fitting hairstyle for different types of faces, we created deform vectors for the entire head, allowing users to widen or narrow the forehead, cheek and jaw areas, as well as lengthen or shorten the head. With these deforms I created seven typical head shapes, as illustrated in figure 5.1). As suggested in [10], changing not the face, but the head and hair around it proved another efficient way of achieving recognizability. In similar fashion, functionality for customizing the eyebrows was also added.

After the people doing the 3D modeling came up with new and improved textures (Fig. 5.2), they created four basic textures: pale/Caucasian, Asian, brown and dark brown. After some research using Von Luschan's chromatic scale, as mentioned in section 2.4, I added an additional parameter to the Mimic that made the render server darken or lighten the chosen skin texture further while rendering the image.

Figure 5.2: Images of the Mimic before and after redesigning the pose and skin textures

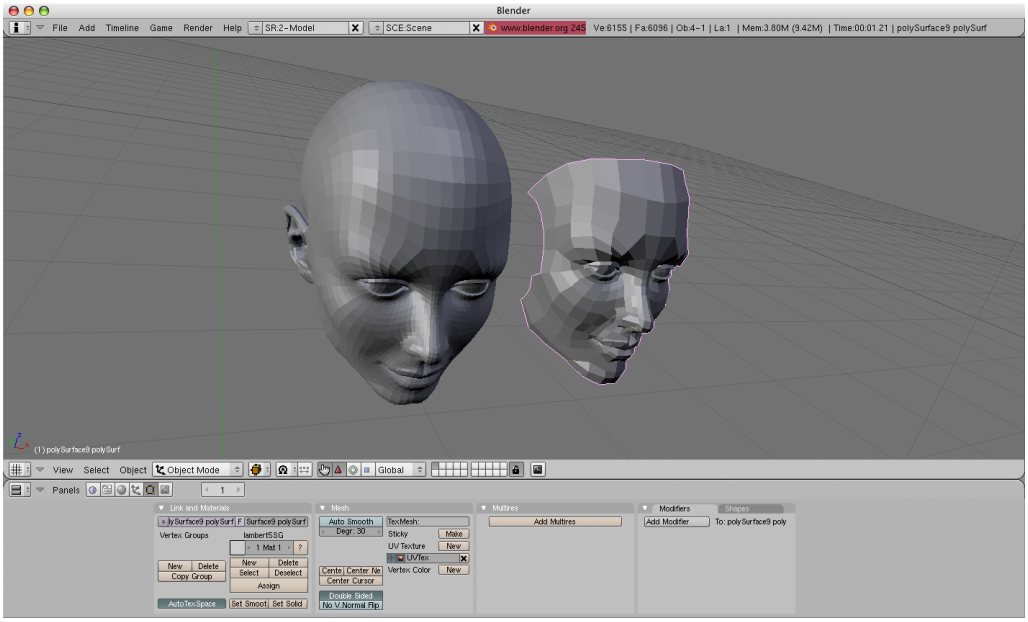
(a) old (b) old (c) new (d) new

The maximum and minimum values for each individual skin type were set manually to be limited to realistic shades. Now, users had extensive control over the Mimic's skin tone. While the Asian skin looked much too stereotypically yellow by default and had to be reworked, the brown and black textures looked very realistic, as can be seen in (Fig. 2.1c).

We attempted to introduce a form of accurate sizing of the Mimic by measuring the Amber's dimensions. While easily measured parameters, like the circumference of each blend value and the Mimic's height are easily translated, the weight parameter, as discussed in chapter 2 remains an estimation.

Because we decided not to use third party software for customized facial texturing, we created our own prototype for this. Creating the face mapping tool took several steps. We created a separate UV map which overlays the Mimic's face, so that we could apply a texture of the face only, instead of having the body and face in one texture, as with the Mimic's default textures. Creating this UV map took some work in Maya for the 3D modelers and some scripting, to rearrange the new UV map's faces to match those of the Mimic. (Fig. 5.3) This new UV map now allowed us to use a new face on the Mimic. This new face had no deform vectors created for it, so using it meant having a custom texture on a neutral face. New textures created from portrait photos had to be scaled to match the new UV map. To do this, we created a trace outline of the original Mimic's face when used with the new UV map. Uploaded photographs now had to be scaled to fit in this outline, so that they fit more or less correctly on the UV map. Scaling faces does not severely impair recognition, as demonstrated in [10]. We just had to make sure that the eyes, mouth and jaw

Figure 5.3: Creating the UV map for the alternative face texture



line in particular were matched. Since not all faces have the same properties, scaling could not always produce exact matches. To visualize the scaling, so that users could do this, I used UvumiTools Crop[1], together with scaling using the Unix `convert` commando to create an interface which allowed users to fit their photo to the outline by simply dragging the edges.

A user now had a way to upload a picture of her face, fit that picture to the required size, and see her face on the Mimic in 3D. While simple, this functionality meant a very different way of picturing the Mimic than before. I will list some future possibilities for this feature in section 5.4.

5.2 Development of MimicMe website

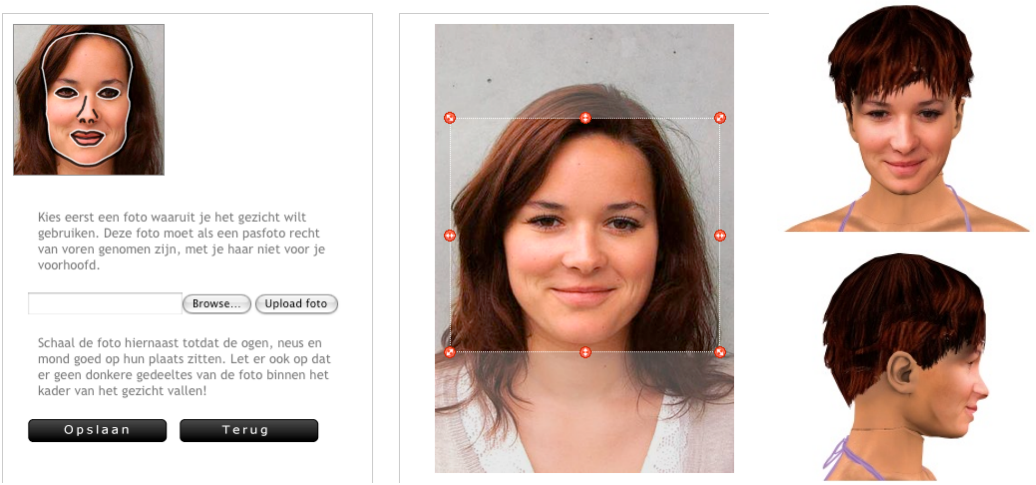
One of the first tasks to be completed for the MimicMe website was to update the design in order to accommodate the new input elements that controlled the Mimic's newly added functionalities. At this point, MimicMe was still in an open Beta stage. Around March, we started building up some hype[2,3] by announcing an official launch, including the new site design, on the first of April. This launch date was chosen because Mimic Media was one of the nominated startups for an award at The Next

[1] http://uvumitools.com/crop.html

[2] www.idealize.nl/2009/04/sociale-3d-paskamer-als-nieuw-advertentiemedium/

[3] www.emerce.nl/nieuws.jsp?id=2902425&utm_campaign=rss&utm_medium=rss&utm_source=rss

Figure 5.4: Uploading and scaling a custom texture



(a) Making the photo image fit the required outline

(b) The texture applied to the Mimic

Web 2009 conference[4], where MimicMe would be presented. The media attention created an influx of new users that allowed us to explore MimicMe's capacities as a fashion community.

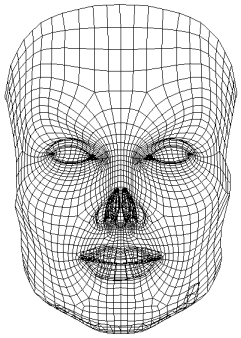
After including the Prototype[5] JavaScript framework and later on the MooTools[6] and script.aculo.us[7] frameworks, the use of more advanced visual features became possible. I replaced MimicMe's original input mechanism, a horizontal line of blocks with one block selected at any time, and introduced a slider which allowed for a more detailed range of values for each deform. Also, through the use of duplicate overlaying images of the Mimic, I was able to make the newly rendered image of the Mimic after making a change fade in, while the old image faded out. This effect, which is also used in Looklet, gives the application a more finished feel, and also allows users to better see the changes they make to their Mimic.

Mimic Media made a deal with Sanoma[8] to implement MimicMe on Shopgirl, their online fashion community which has since been renamed to Fashionfile[9]. This meant another site design, but also a good way to test MimicMe's new JavaScript functionalities. As the first external implementation this also allowed us to test MimicMe's web API, mentioned in chapter 2. As an established community, Shopgirl was a great way to get more users to experience MimicMe.

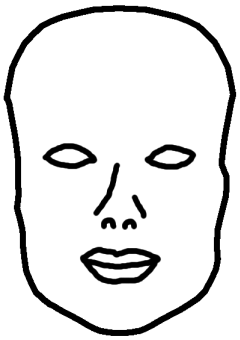
[4] http://2009.thenextweb.com
[5] http://www.prototypejs.org/
[6] http://mootools.net/
[7] http://script.aculo.us/
[8] www.sanoma-uitgevers.nl
[9] http://www.fashionfile.nl/

Figure 5.5: Creating the texture from a photo image

(a) The selected photo image

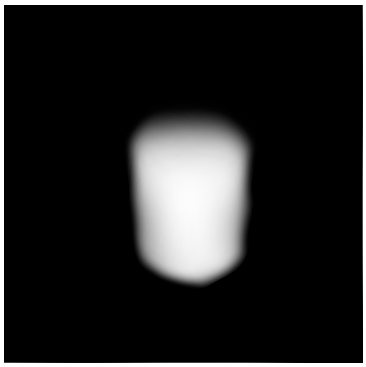


(b) The head shape the image must be fitted to



(c) Outline of the shape for use in the interface

(d) The area selected with the crop tool on the website



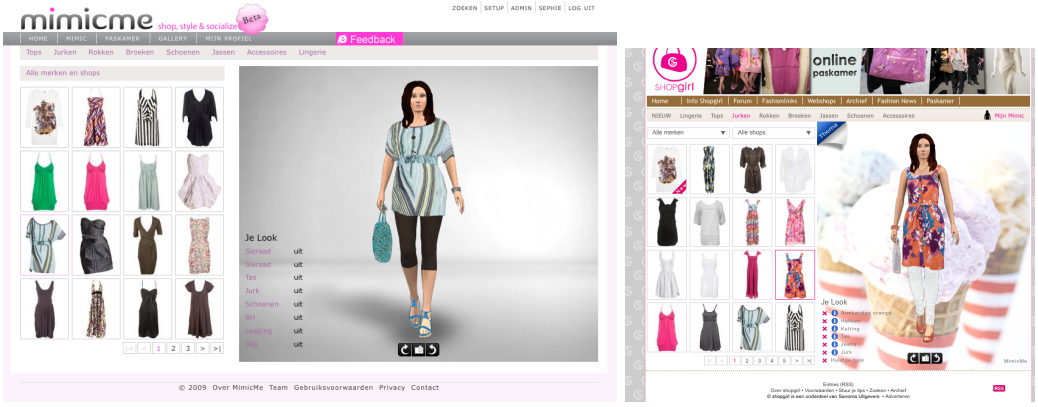
(e) The mask that is applied to soften the edges

(f) The resulting texture, with transparency, ready to be overlaid over the Mimic's face

Figure 5.6: MimicMe as a fashion community

(a) Main page (b) MimicMe profile page

Figure 5.7: The MimicMe shop page



(a) The MimicMe shop page (b) MimicMe implemented on `www.shopgirl.nl`

With most development now being done to improve the instance of MimicMe running at Shopgirl, our own community at MimicMe.com was now less active. After testing on the local server, most new developments were now directly deployed to Shopgirl's MimicMe instance, leaving MimicMe.com at somewhat of a standstill. Our limited development resources were better utilized there where MimicMe got the most exposure.

5.3 Results summary

The development done on both the Mimic and the website has taken MimicMe to a new phase in its development. The overall graphical quality of the Mimic and the Mimic's customizability have improved. With the research done on virtual worlds and face modeling software, a prototype in-house tool has been developed that could

allow for greater recognizability of the Mimic. As for MimicMe as a product, the launch and presentation at The Next Web 2009 and the implementation on fashion community Shopgirl were a great step forward and allowed MimicMe to reach a larger audience.

5.4 Future areas of development

While my internship ended with the completion of the face texturing prototype, many more options are still left to explore. From the Mimic's current state, several areas can still be developed by taking ideas from other avatar systems.

- Hairstyles are still static, but could be made more customizable by adding deformation vectors to the meshes. This would allow users more control over the hair length, which has been a much requested feature.

- The blend value for the bust does not yet provide sufficient quality in rendering breasts. A way to select different cup sizes should be introduced. Also, without using blending the amount of cleavage could be simulated using lighting techniques on the texture. Stylezone, which achieves excellent results using 3D only, remains an important source of inspiration for this.

- Grouping related deform vectors in a grid, much the same way as in Virtual Me (Fig. 3.12) could greatly improve customization of some parts, because the user then has a better overview. Possible uses of grids over separate sliders are in the nose, shape of the eyes, face profile, jaw line, mouth, posture and face shape.

- Cheekbones are an important part of the face. By adding deforms for the cheekbones, next to the 7 standard head shapes, more unique faces could be created. Cheekbones should be positionable in all three dimensions.

- Stylezone allows many different facial expressions, by simulating individual gestures like blink, frown, squint, sneer and smile. People's neutral faces seldom have the eyes wide open. The addition of a deform which allows the eyes to be less than completely open means that a more natural look becomes possible.

- The work on custom UV maps could be expanded to allow users to personalize their Mimic by recreating tattoos of the user. Adding an image of the tattoo, with transparency transfers a very personal property of the user to her digital representation as the Mimic. This could also be used the other way around, to test out future tattoo designs and locations. This concept has recently been launched on TatMash[10].

[10] www.tatmash.com

- The newly developed face texturing prototype can be improved by combining 2D reflection modeling using custom textures with 3D surface modeling using deform vectors. This way, the principle that recognition of a face is based both on 2D reflections and 3D surfaces [9] would be applied in a simple but complete avatar system.

Chapter 6

Evaluation & Conclusion

Over the course of my internship we have made a lot of progress on both the level of customization in the Mimic and the design and user interface of the website. While the Mimic's face was somewhat static in the beginning, the Mimic is now potentially more recognizable through the use of better deform vectors, more realistic lighting, more and better hair styles, different eyebrows and the possibility for the user to use her own face as that of the Mimic.

There are many different ways to approach an avatar system, all depending on the goals that have been set for the system and the wishes of its users. In this chapter I discuss what the effect has been of the development done at MimicMe on the recognizability of the Mimic to its users, and discuss whether or not MimicMe is on the right path to meeting its goals.

6.1 Answering the research question

The research question in section 1.1 asks what makes an avatar recognizable by its user. The theoretical answer to this question can be found in the literature. Sinha's excellent overview of machine vision [10] shows us what the human visual system is capable of and which shape and color ques matter to humans in recognition of faces. O'Toole, Vetter and Blanz demonstrate in [9] that shape and pigmentation ques are equally important in facial recognition and expressing the human form with eigenvectors in [2] show us the possibility of recreating any human in digital form.

However, it quickly becomes clear that for a startup company, where time, money and manpower are limited, a simpler approach is needed, where only the avatar's essential features are modeled. Essential features are those that contribute the most to recognition and are also the easiest to implement. Modeling eyebrows is essential [10] and can be achieved, while modeling the leaning posture is also an important

feature [2], but very much irrelevant to the MimicMe application at this moment due to the fact that the Mimic's posture is fixed.

Developing an avatar system creates two dilemmas. One between the level of complexity in an avatar's features and the realism of the avatar. The other between the avatar's graphical quality and its development costs.

Complex creation & realistic avatar vs. Simple creation & stylized avatar

There is the trade-off between having a simple user interface with less options for avatar customization and having a complex interface which allows detailed customization of the avatar. This trade-off is greatly influenced by a system's intended audience. While a system like Second Life is aimed at an audience that appreciates complex technology, MimicMe's users should be able to create their Mimic with relative ease and in very few steps. One thing that can be observed is that the systems with a simple user interface often have a more stylized avatar, like IMVU and Stardoll. MimicMe has a moderately complex user interface and a realistic looking avatar, so it can be argued that MimicMe is on the right track.

A pitfall here is that a realistic avatar with little options for customization essentially becomes a mannequin. Looklet is a prime example of this, and it comes as no surprise that of all the virtual worlds discussed in this thesis, Looklet has the *least options* for customization and the *most realistic* avatar.

While MimicMe's current user interface consists solely of sliders and selectable styles of hairstyles, eyebrows, etc., Grouping related customization options together into an effective user interface could allow MimicMe to further simplify its interface, while maintaining its level of realism. If a system can only have few options for customization, it is important that the few options that are used are the most essential to the avatar's recognizability. Using an efficient creation process in combination with a priority for essential features would be the best way for a small company to approach realistic avatars.

Graphical excellence & high development costs vs. Low-end graphics & low development costs

Another trade-off that is more applicable to computer systems in general is between quality and development cost. Simply put, MimicMe has little development resources, so creating a complex, but highly effective system like FaceGen is near-impossible. While MimicMe's rendering process certainly isn't the most simple, it is still subject to various limitations, especially in the body area, where blending values are used instead of deform vectors. To change this way of rendering the Mimic would mean changing the fundamentals of MimicMe's systems; something we do not have the resources for at this moment.

6.2 About the internship

Doing an internship at a small company was a great opportunity for me. With three founders and one full-time employee, the other interns and myself felt that everything we worked on mattered to the company.

While my role was to further develop customization of the Mimic. I also spent time on the implementation of the MimicMe website. While not directly related to the Mimic, it was still a necessary and learning process. For the skin color, eyebrows and face modeling tools I received a lot of help from my two colleagues, designated as the modeling *department*, while I myself was seated at the research & development table. A small company can have big dreams.

As to MimicMe's vision on the avatar, this wasn't always clear. While the Mimic is all about recognition between the users and their Mimics, this aspect was sometimes sacrificed for the sake of simplicity or time constraints. Understandably, when resources are limited, compromises have to be made.

6.3 MimicMe in the market

When examining MimicMe in relation to other avatar systems, it becomes apparent that MimicMe is unique because of its architecture, where rendering of the Mimic is done server-side. MimicMe delivers images of high-polygon models that can be viewed by any browser without the need for any separate plugin. Because this sort of hybrid way of rendering is not completely 3D it poses some limitations on the amount of interactivity provided; Models can only be viewed from six different angles. However, camera control can be easily expanded to allow for any viewing angle to be chosen in theory.

Whether or not MimicMe's architecture is very scalable remains to be seen, since the systems have not been yet under such a load that the limit has ever been reached. The concept of having multiple instances of MimicMe running on multiple machines works well with MimicMe's plan to sell to multiple online retailers.

Relative newcomer Looklet is being praised for its high level of realism and that they did not choose to implement software to allows users to alter the model. However, these choices also take away users' options to customize their model. Creating an avatar that looks realistic is achieved this way, but creating an avatar that looks like yourself has been made impossible. MimicMe wants a user to be able to recognize herself in her Mimic, which is why we spent so much effort on the Mimic's customizability.

The pressure to have glamorous-looking models on a fashion website is large, as dictated by the clothing industry. It is having a hard time making a successful transition to an online business model where the internet is used as more than just an online storefront, but also a new way of presenting apparel items in ways that have not yet been done before. It is here that virtual fitting room applications each try in

their own way to involve the customer in a new online buying experience in the hope that large retailers become convinced of the added value of such an application to their online effort. MimicMe has not yet landed such a deal with one of the big fish in the pond. The recent recession has hit the fashion industry hard, which has not made things easier for Mimic Media. But development and improvement on MimicMe will continue until its goal is met: satisfied users who use their lookalike Mimics to shop for clothing on many different web shops.

Bibliography

[1] B. Allen, B. Curless, and Z. Popović, "The space of human body shapes: reconstruction and parameterization from range scans," *ACM Transactions on Graphics (TOG)*, vol. 22, no. 3, pp. 587–594, 2003.

The authors describe how to create 3D parameterized human figures from range scans from the CAESAR data set, a collection of 3D scans of thousands of people from the United States and Europe. The CAESAR subjects also wore 74 markers at various anthropometric landmarks. After filling holes in the range scans, typically located around the ears, armpits and inner thighs, principal component analysis (PCA) is used to characterize the space of human body variation. Mapping these variations to intuitively recognizable facial features such as eye size, mouth width and sex has been done in [Blanz and Vetter, 1999]. Similar explorations centered on the human body are done in [Azouz et al. 2005].

[cited at p. 48]

[2] Z. Azouz, C. Shu, R. Lepage, and M. Rioux, "Extracting main modes of human body shape variation from 3D anthropometric data," in *3-D Digital Imaging and Modeling, 2005. 3DIM 2005. Fifth International Conference on*, 2005, pp. 335–342.

The authors use PCA on 3D scans from the CAESAR data set to describe the human body's main modes of variation. An important first step is to normalize the models so that they are in complete correspondence to each other. Using PCA on the normalized set, a number of eigenvectors is extracted from the data. Using animation on the individual eigenvectors, starting from the one resulting in the highest percentage of shape variation, the modes of variation are recognised. The 5 most prominent variations are mentioned. Most variation occurs in the subject height. The models are height-normalized to allow for better isolation of other modes. Other variations include weight, leaning posture, hip-abdomen ratio and head position. An estimated 64 eigenvectors are enough to account for 95 percent of total variation.

[cited at p. 48, 63, 64]

[3] M. Bastioni, S. Re, and S. Misra, "Ideas and methods for modeling 3D human figures: the principal algorithms used by MakeHuman and their implementation in a new approach to parametric modeling," in *Proceedings of the 1st Bangalore annual Compute conference*. ACM, 2008, p. 10.

The authors describe algorithms used in MakeHuman to organize a thousand parameters into a system that is more easily usable by users that approach the human form from an anthropomorphic point of view. For the new version of MakeHuman they created fuzzy widgets to better read vague input on, for instance, how muscular, tall, or feminine models should be. Based on user requirements and feedback from the MakeHuman community, 8 fuzzy categories are created to describe the basic human form, organized into 4 2-dimensional widgets: Gender & Age, Muscularity & Weight, Breasts (conical & spherical) and Proportion (Limb length & various general body forms). These widgets work on the large number of parameters that shape the MakeHuman model. For instance, there is a great difference in fat distribution between a 30-year old person and a 70-year old person. The widgets handle these differences transparent to the user. Once these settings are set, the user can move on to customizing other targets such as the head and face.

[cited at p. 51]

[4] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," in *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*. ACM Press/Addison-Wesley Publishing Co., 1999, pp. 187–194.

The authors introduce a framework for modeling textured 3D human faces using a morphable model which allows faces to be generated from a set of parameters or be generated from one or more photographs. The morphable model itself is obtained by transforming the shape and texture of a large set of example faces into vector spaces. Through principal component analysis, different coefficients for easily recognizable face variations are isolated, such as width of the mouth. Complex variation coefficients, such as femininity, are obtained by adding or subtracting vectors that are hand-picked from a set of example faces. An algorithm is described that automatically matches the morphable model to one or more submitted 2D images or a submitted 3D scan.The framework can be used to create 3D faces from photographs that can subsequently be altered with regard to their facial features, facial expression and illumination.

[cited at p. 48, 50]

[5] E. Castronova, "Virtual worlds: A first-hand account of market and society on the cyberian frontier," 2001. [cited at p. 26]

[6] J. Huang, V. Blanz, and B. Heisele, "Face recognition using component-based SVM classification and morphable models," *Pattern Recognition with Support Vector Machines*, pp. 531–540, 2002.

The authors present an approach to pose and illumination invariant face recognition that combines two recent advances in the computer vision field: component-based recognition and 3D morphable models. In a first step a 3D morphable model is used to generate 3D face models from only two input images from each person in the training database. Fourteen components were extracted from the original face images based on information from the corresponding model. By rendering the 3D models under varying pose and illumination conditions a vast number of synthetic face images is created which are used to train a component-based face recognition system.

[cited at p. 47]

[7] E. Kim and Y. Kim, "Predicting online purchase intentions for clothing products," *European Journal of Marketing*, vol. 38, no. 7, pp. 883–897, 2004.

The authors study consumer intentions to buy apparel items online. A mailing survey was conducted with 303 adults who had a computer at home and had access to the Internet in the USA. The perceived attributes of online shopping consisted of four factors: transaction/cost; incentive programs; site design; and interactivity. Of these factors, transaction/cost and incentive programs, along with demographic variables (gender, income and number of children),were important predictors in determining the intention to purchase clothing, jewelry, or accessories via the Internet. Incentive program also mediated the relationship between education level and online purchase intention.

[cited at p. 3]

[8] J. Nantel, "My Virtual Model: Virtual reality comes into fashion," *Journal of Interactive Marketing*, vol. 18, no. 3, pp. 73–86, 2004. [cited at p. 20]

[9] A. O'toole, T. Vetter, and V. Blanz, "Three-dimensional shape and two-dimensional surface reflectance contributions to face recognition: An application of three-dimensional morphing," *Vision research*, vol. 39, no. 18, pp. 3145–3155, 1999.

The authors measure the contributions of three-dimensional shape and two-dimensional surface reflectance to human recognition of faces across viewpoint. Laser scans of human heads are first divided into their two- and three-dimensional components. Next, they create shape-normalized faces by morphing the two-dimensional surface reflectance maps of each face onto the average three-dimensional head shape and reflectance-normalized faces by morphing the average two-dimensional surface reflectance map onto each three-dimensional head shape. Observers learned frontal images of the original, shape-normalized or reflectance-normalized faces, and were asked to recognize the faces from viewpoint changes of 0, 30 and 60. Both the three-dimensional shape and two-dimensional surface reflectance information contributed substantially to human recognition performance, thus constraining theories of face representation to include both types of information.

[cited at p. 49, 61, 63]

[10] P. Sinha, B. Balas, Y. Ostrovsky, and R. Russell, "Face recognition by humans: Nineteen results all computer vision researchers should know about," *Proceedings of the IEEE*, vol. 94, no. 11, pp. 1948–1962, 2006.

A previous work by these authors published online was subtitled "20 results all computer vision researchers should know about." This is a revision of that article from 2005. It presents findings from experimental studies regarding face recognition by humans. Nineteen experiments are described in which strengths and weaknesses of human perception in face recognition can be identified. It aims to give an overview of the current understanding about the way in which humans recognize faces. Through reverse-engineering, these results could help the design of automated systems for human face recognition. The results are categorized according to the themes: Recognition as a function of available spatial resolution, The nature of processing: Piecemeal versus holistic, The nature of cues used: Pigmentation, shape and motion, Developmental progression, and Neural underpinnings of the human visual system.

[cited at p. 47, 54, 55, 63]

[11] A. Vasalou, A. Joinson, and J. Pitt, "Constructing my online self: avatars that increase self-focused attention," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2007, pp. 445–448.

Three studies investigated whether users' strategies for customizing online avatars increase their self-focused attention, also known as private self-awareness. Study 1 showed that a high number of users adapt their avatars to reflect their own appearance. Study 2 demonstrated that users who perceive their avatars to be similar to their own appearance experience as a result heightened private self-awareness. In Study 3, private self-awareness pervaded social interaction taking place over time when users with representative avatars, compared to a control group, reported increased private self-awareness. Drawing from research in interpersonal communication, the authors suggest that avatars which increase their owners' self-focus may have an influence on online behavior in the context of social computing.

[cited at p. 4, 24, 26]

Appendices

Appendix A

Today's MimicMe

Much time has passed since I ended my Internship at MimicMe. Therefore, a short analysis of MimicMe's current status feels appropriate.

While My Virtual Model was MimicMe's closest competitor in terms of graphics and functionality during most of the project, we eventually agreed that we had surpassed MVM on all aspects except income. When MVM declared bankruptcy in November 2009 it turned out that maybe their business model was not that great either.

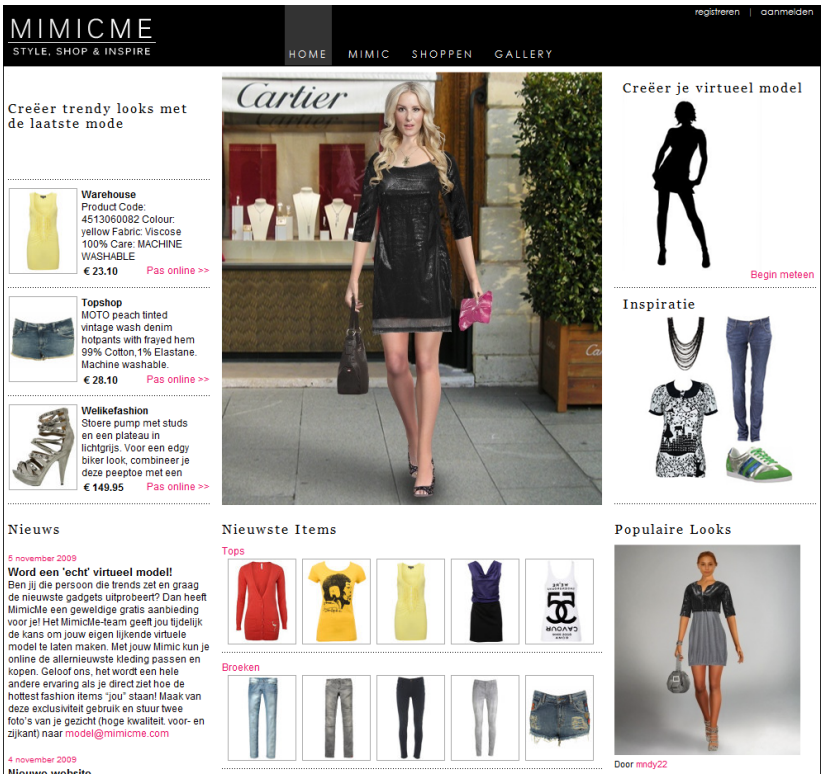
Rising star Looklet is a great example of a fashion community that has it all: a great interface, well-designed website, impressive graphics and a booming community. It is clear that MimicMe is paying close attention to Looklet; MimicMe's front page and shop show similarities to Looklet's designs and even the Mimic has changed dramatically (Fig. A.1). The option for the users to upload their face for use on the Mimic is no longer available, and the software has not been developed further.

The original mission statement of MimicMe included that the users could model the Mimic after their own appearance. Little is left of this idea, now that MimicMe has chosen a style for the Mimic where the user can select from a limited number of faces, all photographs of real people, that are mapped to the six rotational angles of the Mimic. This means that all customizability concerning the head and face has disappeared. We also observed this trend in MVM, where at one point the model could no longer be rotated at all. The Mimic's new faces are actually photo models, so it can be argued that identification with the Mimic's appearance is indeed no longer part of MimicMe's goals, even though this used to be the case.

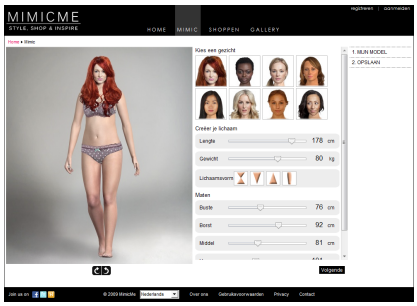
What is left is a very well-detailed Mimic –especially the hair has been restyled and no longer uses 3D meshes– that has, however, drastically reduced its functionality, and original goal, of allowing the user to model her Mimic after herself.

While MimicMe is clearly committed to this change, I feel that it is too bad that MimicMe is now becoming an enemy of its own originality: 3D graphics. While of course still rendered as 2D images, the 3D nature of the Mimic, as opposed to Looklet's static 2D design, allows for a great deal more freedom in rendering the Mimic through zoom, and rotation of the camera. In stead of limiting it, this could allow much more freedom in viewing the Mimic and its clothing.

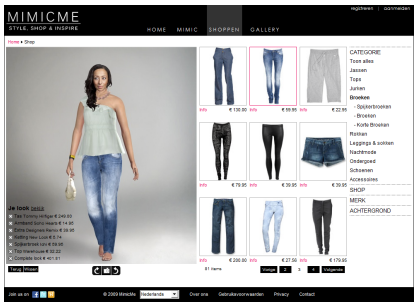
Figure A.1: MimicMe after its redesign in November 2009



(a) The MimicMe website has removed all online community features. What's left is focused entirely on the Mimic and shopping



(b) The Mimic, with new standard faces and simplified body shapes



(c) The shop uses the same style and layout as `www.Looklet.com`

It does not bother me much that large parts of the Mimic that I have spent time working on no longer exist, that is the way it happens in a startup enterprise. I only hope that MimicMe is not trying to target a different market segment; one clearly occupied and fully covered by Looklet. Clothing company H&M has recently added a custom-built version of Looklet to its website[1]. Fashion community Like.com[2] has recently launched Couturious[3], which is clearly a carbon copy of Looklet, much to the latter's dismay.

So there, online fashion is officially booming. The question remains which ideas and companies know best how to woo the giant that is the fashion industry.

[1] http://www.hm.com/nl/fashion/fashionstudio__fashionstudio.nhtml
[2] www.like.com
[3] http://fashionista.com/2010/02/exclusive-couturious-com-launches-tomorrow/

Appendix B

Chronological work log

January 2009

-Introduction to MimicMe's hardware architecture, software architecture and file systems.
-Configuring workstation (Apple PowerBook G4)
-Making a script to automatically insert new hair styles and colors into the file system.
-Implementation of new interface for the *Mimic* page
-Implementation of first set of deform vectors
-Implementation of *Shop* page and experimental feature (*samenshoppen*) that allows users to view a friend's Mimics in real time next to their own, while shopping.
-Using Virtual Me software to compare with MimicMe's current design.
-Installing second workstation (Vista) for with virtual worlds. -Exploring the possibilities of a customizable male avatar (*Eric*)for MimicMe.
-Introduction to the renderer
-Rewriting mimicbox.php for combined usage with multiple avatars.
-Making a planning for the implementation of a customizable male avatar on both the website as well as the renderer.
-Determining the required deform vectors for the male face.
-Getting the bar interface element's JavaScript component to work correctly.

February

-Implementation of male avatar.
-Update of the MimicMe website to show the Male avatar.
-Studying Pawan Sinha's papers on facial recognition[1].
-Visit to TNO's Human Factors department.
-Planning official MimicMe launch on April 1st.
-Reading papers on usage of PCA in modeling humans.
-Adding new hair styles for *Amber*.

[1] http://web.mit.edu/bcs/sinha

March

-Restructuring MimicSizes.php to use a uniform English naming system, and altering the database to reflect this change.
-Changing the way users save their Mimics to use the new database structure.
-Implementing new skin shade option with a shader in the renderer.
-Reading more literature.
-Implementing deform vectors for the head shape.
-Altering the way the renderer renders the hair mesh to make it expand and contract with the head deformations.
-Using Prototype and Script.aculo.us JavaScript frameworks to make changes to the Mimic blend over.
-Preparing for possible use of makeup with the Mimic (includes eyebrows).
-Finalizing the 7 predefined head shapes.
-Implementing the display of real sizes on the *Mimic* page.
-Adding new hair styles and colors.
-Implementing and styling the MimicMe website redesign. New pages for *home*, *outfit*, *item* and *gallery*.

April

-Finishing the new pages for the MimicMe launch.
-Reviewing my work on customization and planning more research of other virtual worlds.
-Adding a hair-textured skin part to the head to make the new curly hair styles look better. Skin shows underneath, which makes the Mimic look balding.
-Making minor fixes to the database.
-Studying KZero and selecting virtual worlds for research.
-Visiting virtual worlds and gathering information on their avatar systems.
-Testing the new makeup elements on the *spijkerbroek* test server to allow for different sets of eyebrows.
-Writing Analysis of customizability 1, which has been incorporated into sections 3.2 and 5.4.
-Exploring FaceGen, and mailing them with a request for information on pricing and possibilities.
-Exploring possibilities of including a JS framework that allows for on-line image resizing.
-Begin implementation of MimicMe on Shopgirl fashion community.
-Reading papers by Marc Rioux.
-Selecting papers for use as reference.

May

-Implementation of MimicMe on Shopgirl.
-Making Analysis of customizability 2 and 3, which have been incorporated into sections 4.2.2 and 5.4.
-Analyzing the possibilities for in-house face texturing.

June

-Finalizing the MimicMe implementation on Shopgirl.
-Exploring UvumiTools Crop[2].
-Developing face texturing feature.

[2] http://uvumitools.com/crop.html

List of Figures


MimicMe
Mimic
avatar
face
users
used
user
virtual
use
online
Mimic's
fashion
customization
website
Virtual
time
also
clothing
like
texture
new
different
body
way
MimicMe's
features
software
model
avatars
render
human
worlds
head
system
create
photo
vectors
now
textures
even
room
hair
architecture
section
Stardoll
want
data
modeling
less
set
Stylezone
images
facial
avatar's
introduced
site
customizable
sliders
IMVU
Avatar
made
longer
eyebrows
makes
renderer
CHAPTER
web
shapes
view
Second
still
files
Model
change
parameters
available
database
able
mouth
faces
development
Figure
form
means
better
possible
recognition
various
results
internship
two
eyes
community
fit
skin
male
texturing
blend
changed
image
height
good
nose
rendered
options
make
chapter
server
color
mesh
length
uses
appearance
values
content
deform
research
tting
MIMICME
using
first
shape
applications
altered
size
focus
allow
interface
recognizable
width
well
Summary
done
Users
basic
level
value
Creating
current
audience
realistic
alter
requirements
request
one
female
created
stored
Graphics
Mimics
around
style
Twinity
page
application
re
every
many
point
sizes
customizability
shop
systems
models
type
important
high
course
games
number
real
work
Faceworx
information
see
apparel
allows
design
added
items
map
world
main
look
parts
Looklet
rendering
range
since
age
much
user's
looks
Life
lot
Fig
process
separate
FaceGen
jaw