BAB 4
IMPLEMENTASI DAN PENGUJIAN

Pada bab ini akan dilakukan implementasi dan pengujian terhadap aplikasi yang dibangun. Tahapan ini dilakukan setelah analisis dan perancangan selesai dilakukan dan selanjutnya akan diimplementasikan kedalam bahasa pemrograman. Setelah implementasi maka dilakukan pengujian terhadap aplikasi. Aplikasi yang telah dibangun diuji untuk mengetahui apakah sistem tersebut dapat berjalan sesuai dengan tujuannya atau tidak

4.1 Implementasi

Tujuan implementasi adalah untuk menjelaskan tentang manual modul kepada semua *user* yang akan menggunakan sistem. Sehingga *user* tersebut dapat merespon apa yang ditampilkan di sistem dan memberikan masukan kepada pembuat sistem untuk dilakukan perbaikan agar sistem lebih baik lagi.

4.1.1 Implementasi Perangkat Pembangun

Perangkat lunak yang digunakan pada sistem komputer yang digunakan untuk implementasi teknologi *augmented reality* pada aplikasi adalah sebagai berikut :

a. Perangkat Keras

Perangkat keras yang digunakan untuk pembuatan aplikasi dijelaskan pada tabel 4.1.

Tabel 4.1 Perangkat Keras Yang Digunakan

Komputer	Spesifikasi perangkat keras
Prosesor	AMD A4-5300 3.4 Ghz
Layar	Resolusi 1600 x 900
Monitor	LED 20”
Memori	Memori 4 GB
Harddisk	Harddisk 320 GB
Kamera	5 Mega Pixel

b. Perangkat lunak

Perangkat lunak yang digunakan untuk mengimplementasikan sistem dijelaskan pada tabel 4.2.

Tabel 4.2 Perangkat Lunak Yang Digunakan

Komputer	Spesifikasi perangkat lunak
Sistem Operasi	Microsoft Window 7
Bahasa Pemrograman	C#
Compiler	Microsoft Visual Studio 2013
Framework	EmguCV versi 2.9

4.1.2 Implementasi Class

4.1.2.1 Implementasi Class Setting

Berikut adalah implementasi class setting pada aplikasi yang dibangun :

```
class Setting
{
      public  Image<Bgr, byte> getBackground();
      public void setBackground();
      public IntrinsicParameter getIntrinsicParameter();
}
```

4.1.2.2 Implementasi Class Augmented

Berikut adalah implementasi class augmented pada aplikasi yang dibangun:

```
class Augmented
{
      public void 3DEstimation(listFinger:ListofPointF,out angle);
}
```

4.1.2.3 Implementasi Class VideoController

Berikut adalah implementasi class VideoController pada aplikasi yang dibangun :

```
class VideoConntroller
{
      public VideoEmgu(GraphicsDevice device,int camIndex);
```

```
      public void Start();
      public void Dispose();
      public void QueryFrame();
}
```

4.1.2.4 Implementasi Class Camera

Berikut adalah implementasi class camera pada aplikasi yang dibangun :

```
class Camera
{
      private  Image<Bgr,byte> current;
      private  Image<Ycc, byte> fore;
      private  Image<Gray, byte> binary;
      private  Image<Ycc,byte> bg;

      public  Image<Bgr,byte> getFrame();
      public  Image<Gray,byte> getBinaryImage();
      public  Image<Ycc,byte> subtractBackground();
      private  Image<Gray, byte> extractSkin();
      private  Image<Gray,byte> smoothImage();
}
```

4.1.2.5 Implementasi Class MainView

Berikut adalah implementasi class MainView pada aplikasi yang dibangun:

```
public partial class MainMenu : Form
{
      private void onUpdateFrame(object sender, EventArgs e);
      private void onClickChangeBackground(object sender,
EventArgs e);
      private void onClickBrowse(object sender, EventArgs e);
      private void onClickStart(object sender, EventArgs e)
}
```

4.1.2.6 Implementasi Class MainAR

Berikut adalah implementasi class MainAR pada aplikasi yang dibangun :

```
public class Game1 : Microsoft.Xna.Framework.Game
{
```

```
      private Model model;
      private Matrix world;
      private Matrix view;
      private Matrix projection;
      private float angleX;
      private float angleY;
      private float angleZ;

      private void DrawModel(Model model, Matrix world, Matrix
view, Matrix projection);
}
```

4.1.2.7 Implementasi Class Detection

Berikut adalah implementasi class Detection pada aplikasi yang dibagun :

```
Class detection
{
      private  Image<Gray,byte> binary;
      private  Image<Bgr,byte> frame;
      private  Contour<Point> biggestContour;
      private  Seq<Point> hull;
      private  Seq<MCvConvexityDefect> defects;
      private  List<PointF> listFingers = new List<PointF>();
      private  MemStorage storage = new MemStorage();
      private CircleF enclosing;
      private CircleF inscribedCircle;

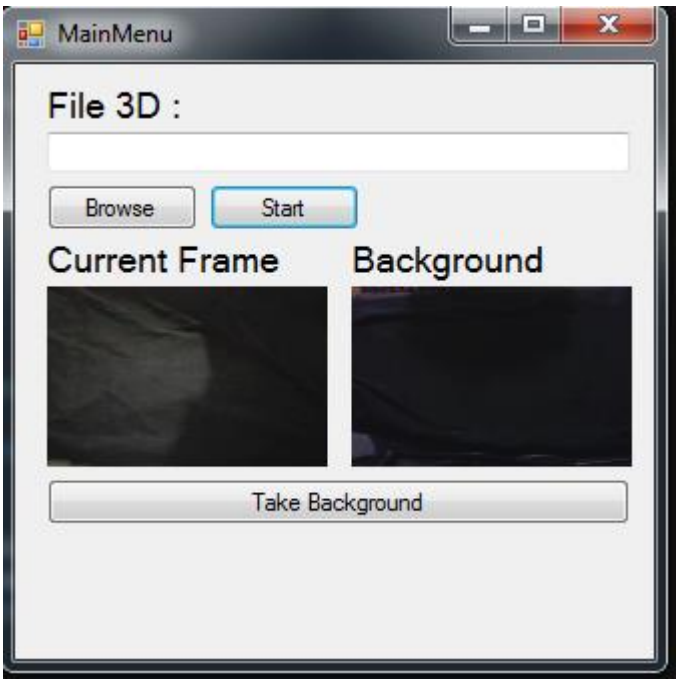
      public  Image<Bgr,byte> handTracking(Image<Bgr,byte>
currentFrame, Image<Gray,byte> binaryImage,out PointF center, out
List<PointF> fingers);
      private  Contour<Point> findBiggestContour();
      private  void findPalmCenter();
      private  void detectFingerLocation();
      private double getAngleA(PointF A, PointF B, PointF C);
      private int getIndex(Point[] contourPoint, Point point);
}
```

4.1.3 Implementasi Antarmuka

Pada tahap ini dilakukan penerapan hasil perancangan antarmuka kedalam sistem yang dibangun dengan menttunakan perangkat lunak yang telah dipaparkan pada sub bab implementasi perangkat lunak. Berikut ini beberapa tampilan antarmuka yang telah diimplementasikan.

4.1.3.1 Implementasi Antarmuka Tampilan Menu Utama

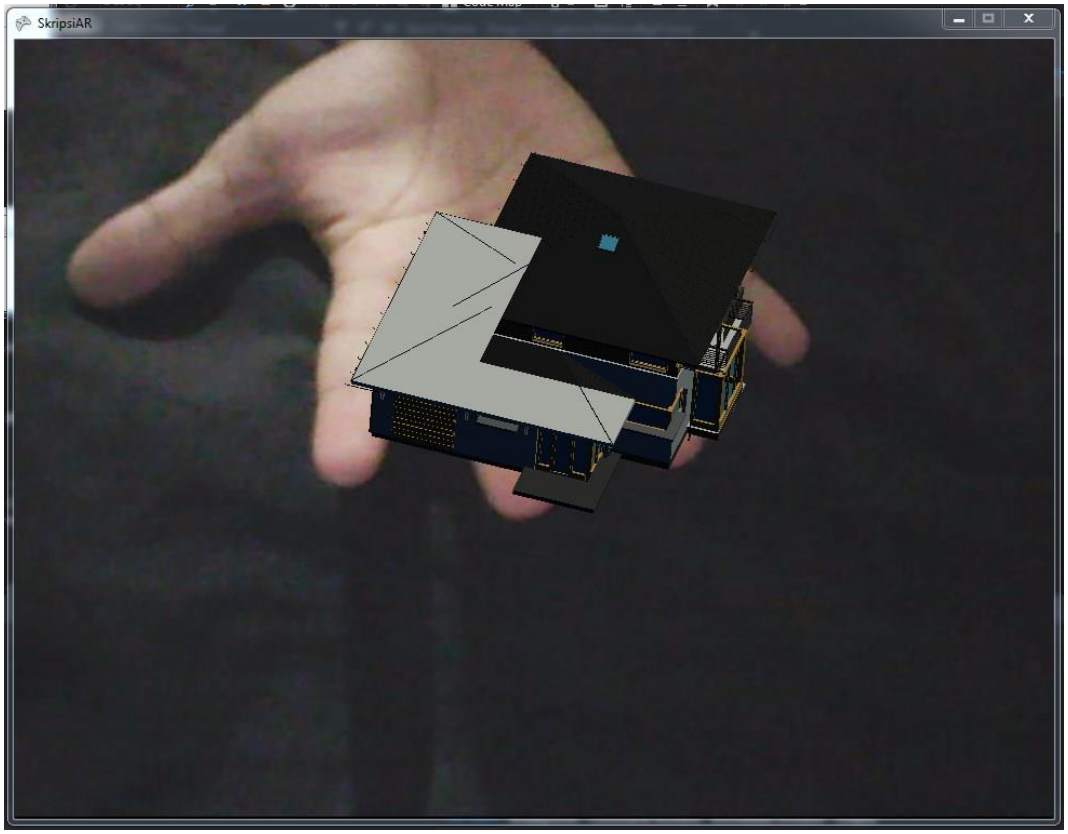
Implementasi antarmuka tampilan menu utama dapat dilihat pada gambar 4.1.



Gambar 4.1 Tampilan Menu Utama

4.1.3.2 Implementasi Antarmuka Tampilan Utama AR

Implementasi antarmuka tampilan utama AR dapat dilihat pada gambar 4.2.



Gambar 4.2 Implementasi Antarmuka Tampilan Utama AR

4.2 Pengujian

Pengujian sistem dilakukan bertujuan untuk menemukan kesalahan atau kekurangan pada perangkat lunak yang telah diuji. Pengujian bermaksud untuk mengetahui perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai dengan tujuan perancangan perangkat lunak tersebut. Dalam penelitian ini pengujian yang dilakukan terhadap sistem yaitu pengujian secara fungsional (*alpha*). Metode yang digunakan dalam pengujian ini adalah pengujian *blackbox* yang berfokus pada persyaratan fungsional dari sistem yang dibangun.

4.2.1 Pegujian *Alpha*

Pengujian *alpha* dilakukan pada sisi pengembangan yang merekam semua kesalahan dan masalah pemakaian. Pengujian *alpha* dilakukan pada sebuah lingkungan yang terkendali.

4.2.2 Rencana Pengujian

Rencana pengujian adalah pengujian terhadap fungsionalitas yang terdapat dalam aplikasi, apakah fungsionitas tersebut sesuai yang diharapkan atau tidak. Rencana pengujian dari sitem yang dibangun dapat dilihat pada tabel 4.3.

Tabel 4.3 Rencana Pengujian Aplikasi

No	Komponen Yang Diuji	Butir Pengujian	Jenis Pengujian
1	Fungsionalitas Mengambil Background	Mengambil background dengan warna yang tidak mirip dengan warna kulit	*Blackbox*
		Mengambil background dengan warna yang mirip dengan warna kulit	*Blackbox*
		Mengambil background dengan tangan terambil oleh kamera	*Blackbox*
2	FungsionalitasMemilih Objek 3D	Memilih file objek 3D Berformat Fbx dari file dialog	*Blackbox*
		Memilih file objek 3D berformat 3ds dari file dialog	*Blaxbox*
		Memilih file objek 3D berformat obj dari file dialog	*Blaxbox*
		Memilih file objek 3D berformat dae dari file dialog	*Blaxbox*
3	Fungsionalitas Hand Tracking	Mengarahkan lima jari ke kamera	*Blackbox*
		Mengarahkan empat jari ke kamera	*Blackbox*
		Mengarahkan tiga jari ke kamera	*Blackbox*
		Mengarahkan dua jari ke kamera	*Blackbox*
		Mengarahkan satu jari ke kamera	*Blackbox*
		Mengarahkan kepalan tangan ke kamera	*Blackbox*
4	Fungsioanilatas Memunculkan Objek 3D	Mengarahkan kelima jari ke kamera	*Blackbox*
		Mengarahkan empat jari ke kamera	*Blackbox*
		Mengarahkan tiga jari ke kamera	*Blackbox*
		Mengarahkan dua jari ke kamera	*Blackbox*
		Mengarahkan satu jari ke kamera	*Blackbox*
		Mengarahkan kepalan tangan ke kamera	*Blackbox*
		Mengarahkan Tangan dengan sudut 45^0	*Blackbox*
		Mengarahkan Tangan dengan sudut 0^0	*Blackbox*

4.2.3 Kasus dan Hasil Pengujian *Blackbox*

Kasus dan hasil pengujian berisi pemaparan dari rencana pengujian yang telah disusun pada skenario pengujian. Pengujian ini dilakukan secara *blackbox* dengan hanya memperhatikan masukan ke dalam sistem dan keluaran dari masukan tersebut. Berdasarkan rencana pengujian, maka dapat dilakukan pengujian *blackbox* pada aplikasi sebagai berikut :

4.2.3.1 Pengujian Fungsionalitas Mengambil Background

Pengujian fungsionalitas mengambil background merupakan pengujian fungsionaltas untuk mengklik mengambil background dari lingkungan yang ditangkap kamera. Pengujian fungsinalitas mengambil background dapat dilhat pada tabel 4.4.

Tabel 4.4 Pengujian Fungsionalitas Mengambil Background

No	Kompnen yang Diuji	Skenario Uji	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Fungsionalitas Mengambil Background	Mengambil background dengan warna yang tidak mirip dengan warna kulit	Proses background subtraction akan berjalan dengan baik sehingga sistem pendeteksian jari akan berjalan dengan baik pula.	Proses background subtraction berjalan dengan baik sehingga proses pendeteksian jari berjalan dengan baik pula.	[√] Diterima [ ] Ditolak
2		Mengambil background dengan warna yang mirip dengan warna kulit	Proses background subtraction tidak akan berjalan dengan baik sehingga proses pendeteksian tidak akan berjalan dengan baik.	Proses background subtraction tidak berjalan dengan baik sehingga proses pendeteksian tidak berjalan dengan baik	[√] Diterima [ ] Ditolak
3		Mengambil background dengan tangan terambil oleh kamera	Proses background subtraction tidak akan berjalan dengan baik sehingga proses pendeteksian teidak akan berjalan dengan baik.	Proses background subtraction tidak berjalan dengan baik sehingga proses pendeteksian tidak berjalan dengan baik.	[√] Diterima [ ] Ditolak

4.2.3.2 Pengujian Fungsional Memilih Objek 3D

Pengujian fungsionalitas memilih objek 3d merupakan pengujian untuk memilih objek 3D yang akan ditampilkan. Pengujian fungsional memilih objek 3D dapat dilihat pada tabel 4.5.

Tabel 4.5 Pengujian Fungsionalitas Memilih Objek 3D

No	Komponen Yang Diuji	Skenario Uji	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Fungsionalitas Memilih Objek 3D	Memilih file objek 3D Berformat Fbx dari file dialog	Objek 3D akan berhasil terpilih dan akan ditampilkan pada tampilan utama AR ketika aplikasi mendeteksi keberadaan tangan	Objek 3D berhasil terpilih dan ditampilkan pada tampilan utama AR ketika aplikasi mendeteksi keberadaan tangan	[√] Diterima [ ] Ditolak
2		Memilih file objek 3D berformat 3ds dari file dialog	Objek 3D tidak berhasil ditampilkan karena format yang tidak mendukung	Objek 3D tidak berhasil ditampilkan karena format yang tidak mendukung	[√] Diterima [ ] Ditolak
3		Memilih file objek 3D berformat obj dari file dialog	Objek 3D tidak berhasil ditampilkan karena format yang tidak mendukung	Objek 3D tidak berhasil ditampilkan karena format yang tidak mendukung	[√] Diterima [ ] Ditolak
4		Memilih file objek 3D berformat dae dari file dialog	Objek 3D tidak berhasil ditampilkan karena format yang tidak mendukung	Objek 3D tidak berhasil ditampilkan karena format yang tidak mendukung	[√] Diterima [ ] Ditolak

4.2.3.3 Pengujian Fungsionalitas *Hand Tracking*

Pengujian Fungsionalitas *Hand Tracking* merupakan pengujian untuk menguji fungsionalitas *hand tracking* yang ada pada aplikasi. pengujian fungsionalitas *hand tracking* dapat dilihat pada tabel 4.6.

Tabel 4.6 Pengujian Fungsionalitas *Hand Tracking*

No	Komponen Yang Diuji	Skenario Uji	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Fungsionalitas *Hand Tracking*	Mengarahkan lima jari ke kamera	Aplikasi akan berhasil mendeteksi keberadaan lima jari	Aplikasi berhasi mendeteksi keberadaan lima jari	[√] Diterima [ ] Ditolak
2		Mengarahkan empat jari ke kamera	Aplikasi akan berhasil mendeteksi keberadaan empat jari	Aplikasi berhasil mendeteksi keberadaan empat jari	[√] Diterima [ ] Ditolak
3		Mengarahkan tiga jari ke kamera	Aplikasi akan berhasil mendeteksi keberadaan tiga jari	Aplikasi berhasil mendeteksi keberadaan tiga jari	√] Diterima [ ] Ditolak
4		Mengarahkan dua jari ke kamera	Aplikasi akan berhasil mendeteksi keberadaan dua jari	Aplikasi berhasil mendeteksi keberadaan dua jari	√] Diterima [ ] Ditolak
5		Mengarahkan satu jari ke kamera	Aplikasi akan berhasil mendeteksi keberadaan satu jari	Aplikasi berhasil mendeteksi keberadaan satu jari	√] Diterima [ ] Ditolak
6		Mengarahkan kepalan tangan ke kamera	Aplikasi tidak kan mendeteksi keberadaan jari tangan	Aplikasi tidak mendeteksi keberadaan jari tangan	√] Diterima [ ] Ditolak

4.2.3.4 Pengujian Fungsionalitas Memunculkan Objek 3D

Pentujian fungsionalitas memunculkan objek 3D merupakan pengujian untuk fungsianlitas memunculkan objek 3D pada aplikasi. Pengujian fungsionalitas memunculkan objek 3D dapat dilihat pada tabel 4.7.

Tabel 4.7 Pengujian Fungsionalitas Memunculkan Objek 3D

No	Komponen Yang Diuji	Skenario Uji	Hasil yang diharapkan	Hasil Pengujian	Kesimpulan
1	Fungsionalitas Memunculkan Objek 3D	Mengarahkan lima jari ke kamera	Objek 3D yang telah dipilih akan dimunculkan berdasarkan lokasi pendeteksian tangan dan juga sudut yang telah diestimasi	Objek 3D yang telah dipilih berhasil dimunculkan berdasarkan lokasi pendeteksian tangan dan juga sudut yang telah diestimasi	[√] Diterima [ ] Ditolak
2		Mengarahkan empat jari ke kamera	Objek 3D yang telah dipilih tidak akan dimunculkan	Objek 3D yang telah dipilih tidak dimunculkan	[√] Diterima [ ] Ditolak
3		Mengarahkan tiga jari ke kamera	Objek 3D yang telah dipilih tidak akan dimunculkan	Objek 3D yang telah dipilih tidak dimunculkan	[√] Diterima [ ] Ditolak
4		Mengarahkan dua jari ke kamera	Objek 3D yang telah dipilih tidak akan dimunculkan	Objek 3D yang telah dipilih tidak dimunculkan	[√] Diterima [ ] Ditolak
5		Mengarhakan satu jari ke kamera	Objek 3D yang telah dipilih tidak akan dimunculkan	Objek 3D yang telah dipilih tidak dimunculkan	[√] Diterima [ ] Ditolak
6		Mengarahkan kepalan tangan ke kamera	Objek 3D yang telah dipilih tidak akan dimunculkan	Objek 3D yang telah dipilih tidak dimunculkan	[√] Diterima [ ] Ditolak
7		Mengarahkan tangan dengan sudut 45^0	Objek 3D yang telah dipilih akan dimunculkan berdasarkan lokasi pendeteksian tangan dan juga sudut yang telah diestimasi	Objek 3D yang telah dipilih berhasil dimunculkan berdasarkan lokasi pendeteksian tangan dan juga sudut yang telah diestimasi	[√] Diterima [ ] Ditolak
8		Mengarahkan tangan dengan sudut 0^0	Objek 3D yang telah dipilih akan dimunculkan berdasarkan lokasi pendeteksian tangan	Objek 3D yang telah dipilih berhasil dimunculkan berdasarkan lokasi pendeteksian tangan	[√] Diterima [ ] Ditolak

			dan juga sudut yang telah diestimasi	dan juga sudut yang telah diestimasi	

4.2.4 Pengujian Akurasi Hand Tracking

Pengujian akurasi hand tracking adalah pengujian terhadap performa dari tingkat akurasi proses *hand tracking*. Pada pengujian ini akan diuji permorma terhadap background dan cahaya dengan cara menguji aplikasi pada background dan intensitas cahaya yang berbeda beda. Dalam pengujian background yang digunakan adalah background biru,hitam,merah,hijau,abu,putih, dan background complex. Untuk setiap warna background pengujian dilakukan dengan cara menutupi area yang tertangkap kamera dengan kain berwarna sedangkan untuk background complex pengujian dilakukan dengan cara mengarahkan kamera kelingkungan yang complex yaitu lingkungan yang dipenuhi dengan barang barang. Sedangkan menguji intensitas cahaya pada pengujian ini dilakukan pengujian untuk setiap warna background pada 3 tingkat intensitas cahaya sebagai berikut :

1) Intentensitas cahaya kurang terang dilakukan dengan menguji setiap warna background pada ruangan yang diterangi lampu pijar 10 watt
2) Intensitas cahaya terang dilakukan dengan menguji setiap warna background pada ruangan yang diterangi lampu neon 20 watt.
3) Intensitas cahaya kurang terang dilakukan dengan menguji setiap warna background pada ruangan yang diterangi lampu neon 20 watt dan juga disinari sinar matahari.

Hasil pengujian akurasi dapat dilihat pada tabl 4.8

Tabel 4.8 Pengujian Akurasi Hand Tracking

No.	Bacground	Jumlah Jari Terdeteksi Pada Intensitas Cahaya			Jumlah	Akurasi
		kurang terang	Terang	Sangat Terang		
1.	Biru	5	5	5	15	100%
2.	Hitam	4	5	0	9	60%

3.	Merah	0	0	0	0	0%
4.	Hijau	4	5	4	14	93.3%
5.	Abu	0	5	4	9	60%
6.	Putih	3	0	4	7	46.7%
7.	Complex	4	5	2	11	73.3%

4.3 Kesimpulan Hasil Pengujian

Dari hasil pengujian alpha semua fungsional sudah dapat dijalankan sehingga semua fungsional dalam aplikasi sudah berhasil terpenuhi, dan berdasarkan pengujian performansi dapat dikatakan bahwa proses *hand tracking* pada aplikasi masih bergantung pada pencahyaan dan warna background dengan warna background dengan hasil terbaik adalah biru yang memiliki tingkat akurasi 100%.