BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi

Tahap implementasi dilakukan setelah perancangan selesai dilakukan kemudian dilakukan pengujian terhadap sistem. Implementasi merupakan tahapan meletakan sistem agar dapat dioperasikan oleh pengguna.

4.1.1 Implementasi Perangkat Keras

Perangkat keras yang dibutuhkan dalam pembuatan simulasi ARTatasurya adalah sebagai berikut:

1) PC atau Laptop dengan *Processor* 2 GHz atau lebih
2) *Random Access Memory* (RAM) 1 GB
3) *Harddisk* dengan space 20 GB
4) VGA 256 MB 32 Byte
5) *Webcam* 30 fps 5 Mb Pixel
6) *Marker*

4.1.2 Implementasi Perangkat Lunak

Dalam pembuatan Perangkat lunak ini digunakan software penunjang yaitu:

1) Microsofw window XP SP2
2) Autodesk 3D Studio Max 9
3) ArToolKit Software Library Versi 2.65

4) Adobe Photoshop CS3

5) OpenGL

4.1.3 Implementasi Aplikasi

Implementasi aplikasi ini merupakan sebuah langkah-langkah penggunaan aplikasi dari seluruh sistem yang telah dibuat. Sebelum menjalankan ARTatasurya kita harus memeriksa objek_data_vrml apakah sudah sesuai dengan objek yang akan ditampilkan atau belum. Object_data_vrml berguna untuk memanggil objek yang akan ditampilkan. Untuk mengedit file ini disarankan menggunakan text editor *WordPad*.

1. Eksekusi file

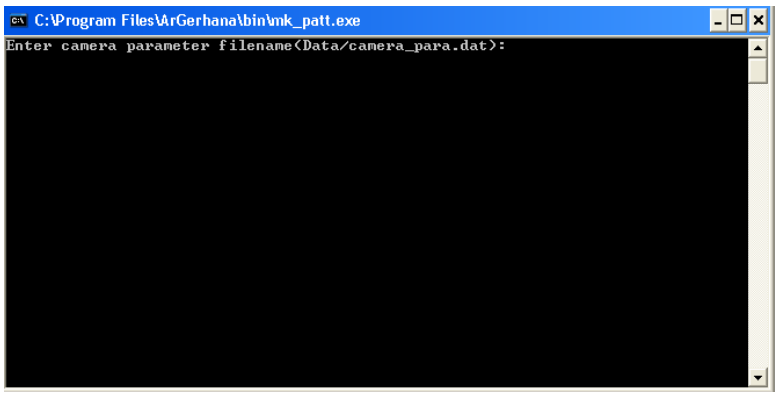
Eksekusi ARTatasurya.exe yang kemudian akan tampil sebuah form seperti yang terlihat pada gambar 4.1. Pada form utama terdapat 4 buah tombol Deteksi *marker*, Jalankan AR, Refresh dan tombol Keuar.



Gambar 4.1 *User Interface* ARTatasurya

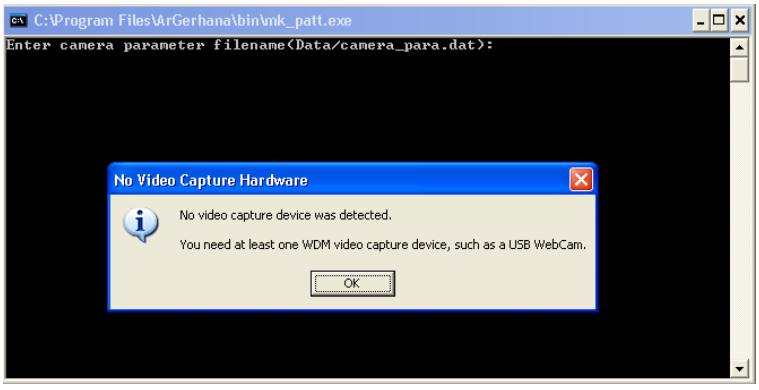
2. Tombol Deteksi *Marker*

Tombol deteksi *marker* ini digunakan untuk mendeteksi *marker* ARTatasurya. Apabila *user* menekan tombol deteksi *marker* maka akan muncul tampilan seperti pada gambar 4.3.



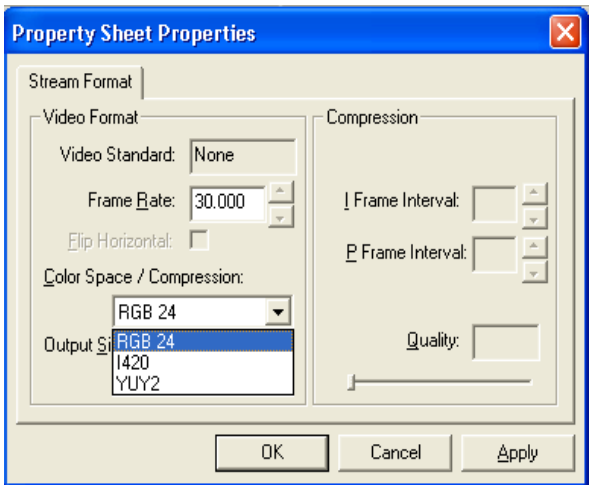
Gambar 4.2 Tampilan parameter pencari *port webcam*.

User harus menekan tombol *enter* pada *keyboard* untuk memeriksa apakah *port webcam* terdeteksi atau tidak. Apabila *webcam* tidak terdeteksi maka akan muncul pesan seperti pada gambar 4.3.



Gambar 4.3 Pesan *webcam* tidak terdeteksi

Apabila PC atau laptop mendeteksi *port webcam*, maka sistem akan otomatis menampilkan *property sheet properties* seperti pada gambar 4.4.

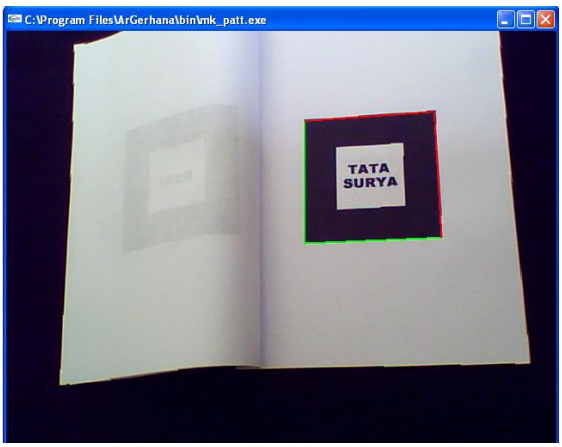


Gambar 4.4 *Property Sheet Properties*

Pada pilihan *Color Space/Compression* pengguna bisa memilih tipe *color space* yang diinginkan. Secara *default* sistem sudah memilih RGB 24.

Deteksi *marker* tata surya

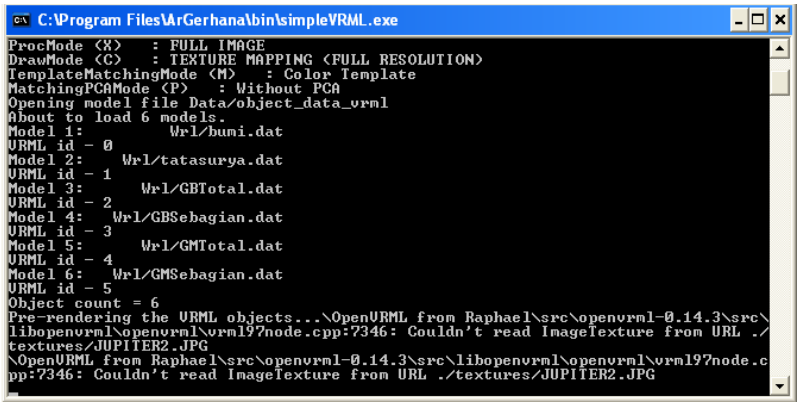
Marker tata surya untuk menempatkan objek tata surya (patt.tatasurya).



Gambar 4.5 deteksi *marker* tata surya

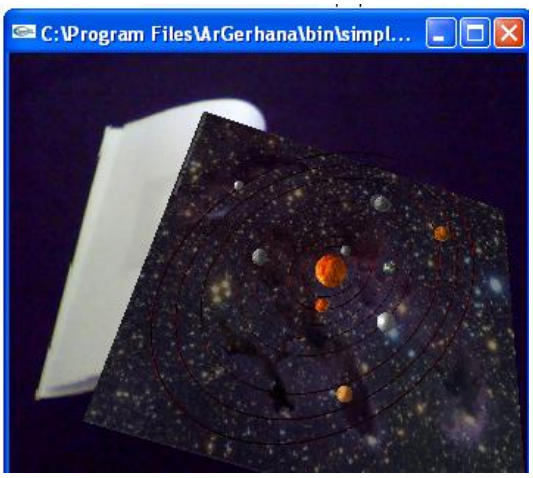
3. *Render* Objek

Render objek adalah tombol yang ada pada *form* utama untuk memunculkan objek virtual. Tampilan utama setelah pengguna menekan tombol Jalankan AR terlihat seperti pada gambar 4.12 yang berisi tentang informasi objek yang akan ditampilkan.

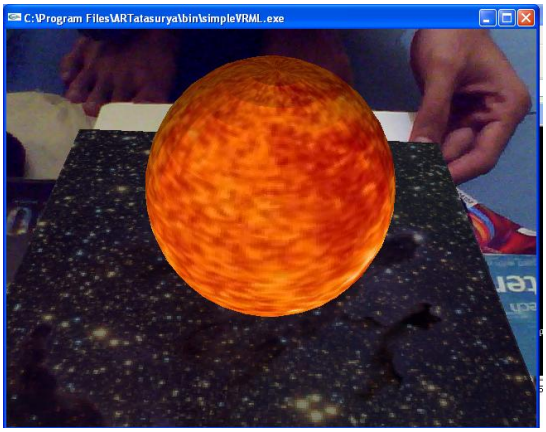


Gambar 4.6 *Camera Parameter*

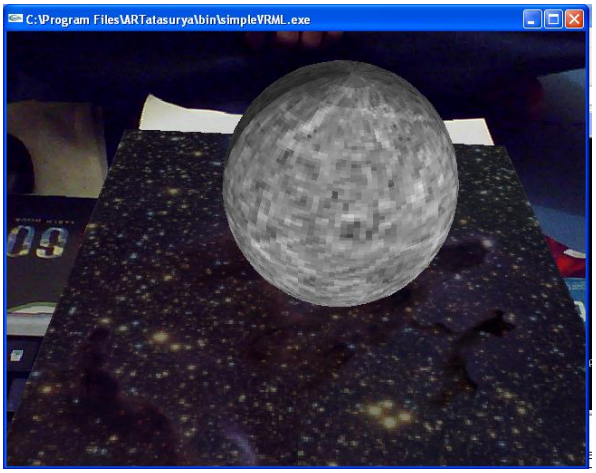
Objek tata surya disini memiliki 9 buah planet dari mulai markurius sampai dengan Pluto. Semua planet akan mengelilingi matahari termasuk juga bumi.



Gambar 4.7 *Tatasurya*



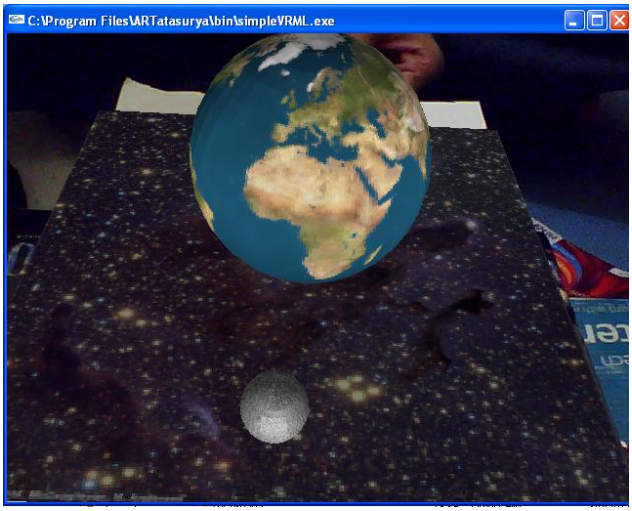
Gambar 4.8 Matahari



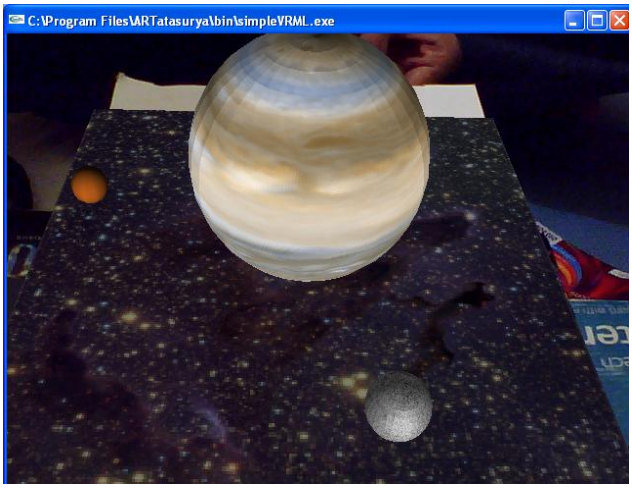
Gambar 4.9 Merkurius



Gambar 4.10 venus



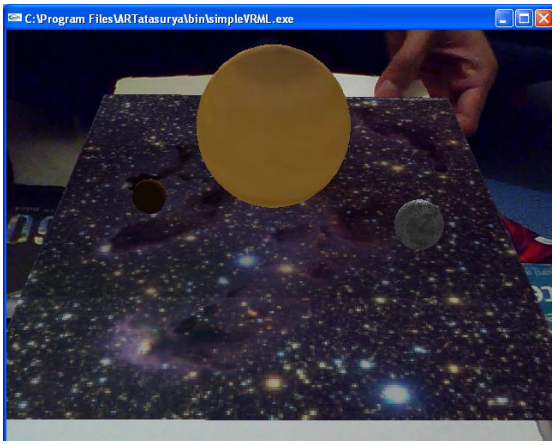
Gambar 4.11 Bumi



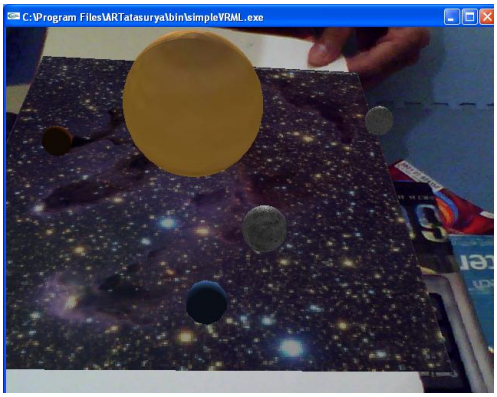
Gambar 4.12 Mars



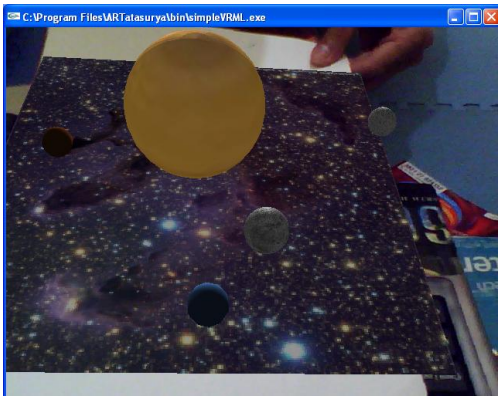
Gambar 4.13 yupiter



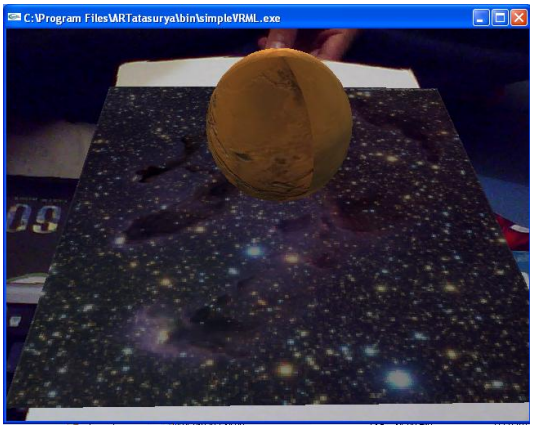
Gambar 4.14 saturnus



Gambar 4.15 uranus



Gambar 4.16 neptunus



Gambar 4.17 pluto

4.2 Pengujian Alpha dan Betha

Pengujian black box berfokus pada persyaratan fungsional perangkat lunak yang dibangun, sedangkan white box berfokus pada persyaratan procedural perngkat lunak yang dibangun.

4.2.1 Rencana Pengujian

Rencana pengujian aplikasi gerhana dilaksanakan dengan tujuan agar sistem dari aplikasi yang dibangun memiliki kemampuan sebagai media peraga proses gerhana. Rencana pengujian dapat dilihat pada tabel 4.1.

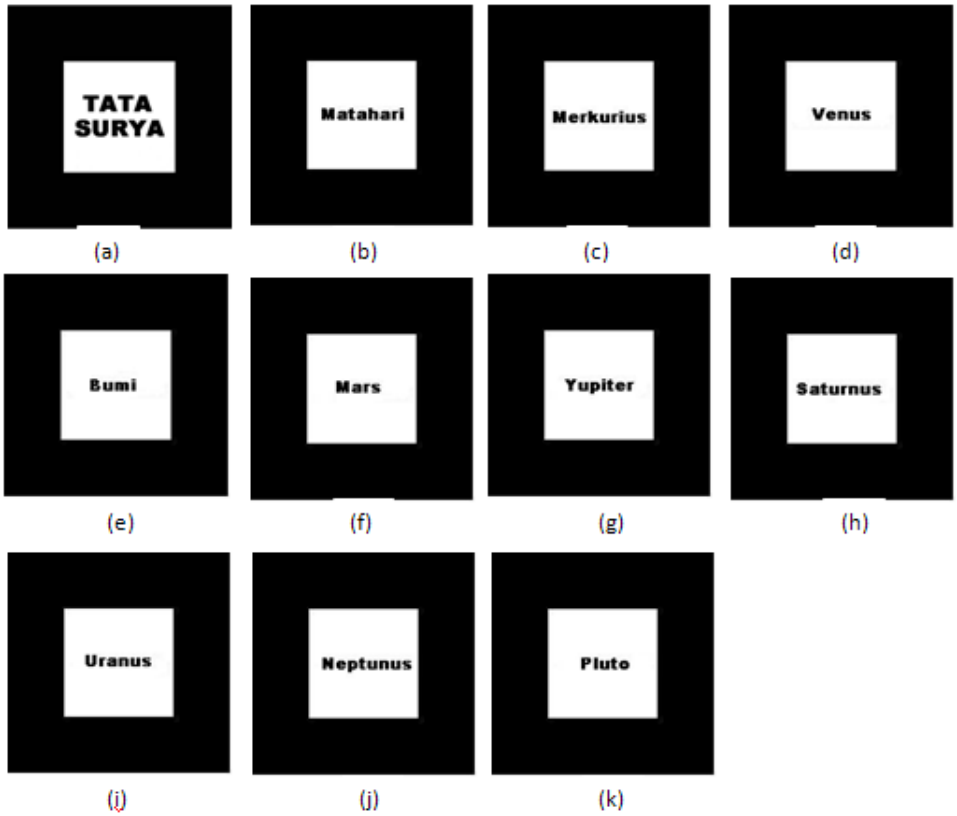
Tabel 4.1 Pengujian aplikasi ARTatasurya

Item uji	**Detail pengujian**	**Jenis pengujian**
Form utama	Menampilkan *form* awal.	*Black Box*
	Menampilkan tombol tambah *marker*, jalankan AR, edit data vrml, Refresh, keluar	*Black Box*
Deteksi *Marker*	Menampilkan parameter kamera	*Black Box*
	Menampilkan *property*	*Black Box*

	sheet	
	Mendeteksi *marker*	*Black box*
Render objek	Menampilkan *property sheet*	*Black Box*
	Menampilkan info objek	*Black Box*
	Menampilkan objek 3D	*Black Box*
Refresh	*Enable* tombol lain.	*Black Box*
Tombol Keluar	Keluar dari program	*Black Box*

4.2.2 Pengujian *Alpha*

Melakukan pengujian terhadap aplikasi yang akan dijalankan mulai dari proses pendeteksian *marker* sampai pada tahap *rendering* objek. Dalam pengujian ini, enam buah *marker* akan dideteksi terlebih dahulu oleh *webcam.* Pendeteksian *marker* dilakukan dengan cara mengarahkan *marker* tepat didepan *webcam* sehingga seluruh permukaan *marker* dapat terlihat oleh *webcam*. *Marker* yang akan di deteksi harus di print terlebih dahulu dengan menggunakan bahan kertas yang tidak memantulkan cahaya. *Marker* sering kali menampilkan model yang bukan modelnya, ini dikarenakan oleh pola dari *marker* yang memiliki kemiripan *background* sehingga menimbulkan kesalahan ketika proses pendeteksian. Pengujian dilakukan dengan menggunakan *webcam Micropack*. Jarak sangat mempengaruhi dalam pelacakan optik ketika *marker* digerakan menjauhi *webcam,* jarak terjauh memiliki *pixel* yang lebih sedikit sehingga tidak cukup detail untuk dapat mengidentifikasi pola pada *marker*. Semakin besar ukuran *marker* maka semakin jauh jarak pendeteksian *marker*. Dalam pengujian ini kedalaman warna menggunakan RGB-24 bit dan *frame rate* 30 fps.



Gambar 4.18 *Marker* ARTatasurya

4.2.3 Kasus dan Hasil Pengujian

1. **Pengujian deteksi *marker***

Tabel 4.2 Tabel pengujian deteksi *marker*

Data masukan	Yang diharapkan	Pengamatan	kesimpulan
Marker a	Mendeteksi *marker* patt.tatasurya	*Marker* a ditandai dan disimpan	Diterima
Marker b	Mendeteksi *marker* tata patt.matahari	*Marker* b ditandai dan disimpan	Diterima
Marker c	Mendeteksi *marker* patt.merkurius	*Marker* c ditandai dan disimpan	Diterima
Marker d	Mendeteksi *marker*	*Marker* d	Diterima

	patt.venus	ditandai dan disimpan	
Marker e	Mendeteksi *marker* patt.bumi	*Marker* e ditandai dan disimpan	Diterima
Marker f	Mendeteksi *marker* patt.mars	*Marker* f ditandai dan disimpan	Diterima
Marker g	Mendeteksi *marker* patt.yupiter	*Marker* g ditandai dan disimpan	Diterima
Marker h	Mendeteksi *marker* patt.saturnus	*Marker* h ditandai dan disimpan	Diterima
Marker i	Mendeteksi *marker* patt.uranus	*Marker* i ditandai dan disimpan	Diterima
Marker j	Mendeteksi *marker* patt.neptunus	*Marker* j ditandai dan disimpan	Diterima
Marker k	Mendeteksi *marker* patt.pluto	*Marker* k ditandai dan disimpan	Diterima

2. Pengujian render objek

Tabel 4.3 pengujian rendering objek

Data masukan	**Yang diharapkan**	**Pengamatan**	**kesimpulan**
Marker a	Menampilkan objek tata surya	Menampilkan planet yang ada dalam tata surya termasuk matahari	Diterima
Marker b	Menampilkan objek matahari	Objek matahari	Diterima
Marker c	Menampikan objek planet merkurius	Rotasi planet merkurius.	Diterima
Marker d	Menampilkan objek planet venus	Rotasi planet venus.	Diterima
Marker e	Menampilkan objek planet bumi	Rotasi planet bumi dan bulan.	Diterima
Marker f	Menampilkan objek planet mars	Planet mars beserta satelitnya	Diterima

Marker **g**	Menampilkan objek planet yupiter	Rotasi planet yupiter beserta satelitnya.	Diterima
Marker h	Menampilkan objek planet saturnus	Rotasi planet saturnus beserta satelitnya.	Diterima
Marker i	Menampilkan objek planet Uranus	Rotasi planet Uranus beserta satelitnya.	Diterima
Marker j	Menampilkan objek planet neptunus	Rotasi planet neptunus beserta satelitnya.	Diterima
Marker k	Menampilkan objek planet pluto	Rotasi planet pluto	Diterima

3. Pengujian Jarak

Tabel 4.4 Spesifikasi kamera uji.

Data msukan	**Resolusi**	**Fps**	**Warna**	**Jarak Terpendek**	**Hasil**
Marker uji	320x240	30	RGB-24 bit	25 cm	OK
	320x240	30	RGB-24 bit	45 cm	Patah-patah
	320x240	30	RGB-24 bit	55 cm	Patah-patah
	352x288	30	RGB-24 bit	25 cm	OK
	352x288	30	RGB-24 bit	45 cm	Patah-patah
	352x288	30	RGB-24 bit	55 cm	Patah- patah
	640x480	30	RGB-24 bit	25 cm	OK
	640x480	30	RGB-24 bit	35 cm	OK
	640x480	30	RGB-24 bit	55 cm	Patah-patah

4.2.4 Kesimpulan Pengujian Alpha

Webcam yang digunakan sebaiknya mempunyai nilai *distorsi* yang rendah agar objek dapat ditampiklan dengan baik dan dapat memperkecil kesalahan perhitungan walaupun dengan *webcam* yang berbeda. ARTatasurya ini akan berjalan dengan baik apabila menggunakan *resolusi* 320 x 240 dengan *frame rate* 30 fps. *Marker* yang digunakan sebaiknya memiliki *background* yang tidak sama

persis, karena apabila memiliki kemiripan *background* maka satu *marker* akan menampilkan beberapa objek.

4.2.5 Pengujian *Betha*

Pengujian *betha* merupakan pengujian yang dilakukan untuk mendapatkan kesimpulan dari aplikasi yang telah dibuat dengan cara memberikan kuesioner kepada *user* secara langsung. Pertanyaan yang diajukan kepada pengguna terdiri dari 10 pertanyaan. Berdasarkan data hasil kuesioner maka dicari prosentase masing-masing jawaban dengan menggunakan rumus kuesioner:

Y=P/Q*100

Keterangan:

Y= nilai prosentase

P = Banyaknya jawaban responden dari setiap soal

Q = Jumlah responden

1. Apakah *Marker* dapat terdeteksi dengan baik?

No	Keterangan	Frekuensi jawaban	Jumlah sampel	Presentase
A	Sangat setuju	2	10	20%
B	Setuju	6	10	60%
C	Ragu-ragu	2	10	20%
D	Tidak setuju	0	10	0

2. Apakah dalam mendeteksi *marker* membutuhkan waktu yang lama?

No	Keterangan	Frekuensi jawaban	Jumlah sampel	Presentase
A	Sangat setuju	1	10	10%
B	Setuju	9	10	90%
C	Ragu-ragu	0	10	0
D	Tidak setuju	0	10	0

3. Apakah tampilan antar muka menarik?

No	Keterangan	Frekuensi jawaban	Jumlah sampel	Presentase
A	Sangat setuju	0	10	0
B	Setuju	3	10	30%
C	Ragu-ragu	5	10	50%
D	Tidak setuju	2	10	20%

4. Apakah objek dapat ditampilkan dengan baik*?*

No	Keterangan	Frekuensi jawaban	Jumlah sampel	Presentase
A	Sangat setuju	1	10	10%
B	Setuju	6	10	60%
C	Ragu-ragu	3	10	30%
D	Tidak setuju	0	10	0

5. Apakah user mendapatkan pengalaman baru tentang teknologi AR?

No	Keterangan	Frekuensi jawaban	Jumlah sampel	Presentase
A	Sangat setuju	5	10	50%
B	Setuju	5	10	50%
C	Ragu-ragu	0	10	0
D	Tidak setuju	0	10	0

4.2.6 Kesimpulan Pengujian Betha

Berdasarkan hasil kuesioner (pengujian *betha*) yang dilakukan maka dapat disimpulkan bahwa aplikasi ARTatasurya dalam mendeteksi dan menampilkan objek telah sesuai dengan prosedur yang diharapkan dan pengguna dapat memahami apa yang disampaikan melalui pemodelan objek 3 dimensi. Pengguna merasa tertarik dengan aplikasi ARTatasurya, karena kebanyakan pengguna belum mengetahui teknologi AR.