

ANALISA AUGMENTED REALITY DALAM APLIKASI PENGENALAN MONUMEN MENGGUNAKAN UNITY-3D

Muhammad Iqbal¹, Deni Arifianto²

¹iqb4l.joy@gmail.com

²deniarifianto@unmuhjember.ac.id

Teknik Informatika

Universitas Muhammadiyah Jember

Jln. Karimata No. 49, Telp (0331) 336728, Jember

ABSTRAK

Salah satu teknologi yang sudah banyak digunakan yaitu *Augmented Reality* (AR). Dalam penelitian ini penulis membahas sebuah aplikasi *Augmented Reality* yang berkaitan dengan dunia pendidikan. Salah satu peran AR dalam dunia pendidikan yaitu pada proses peningkatan minat belajar. Sifat dari AR ini adalah membantu pengguna agar lebih mudah mengenal suatu objek atau benda dalam bentuk 3D, akan tetapi masih banyak orang awam yang belum mengetahui batasan-batasan dalam penggunaan AR. Untuk mendukung hal tersebut maka dibutuhkan aplikasi *Augmented Reality* yang mampu memberikan dukungan pada proses penelitian yang tepat. Proses penelitian AR ini dengan cara menggunakan tabel *blackbox* yang meliputi pengecekan tombol, jarak kamera, sudut, pencahayaan, putaran, oklusi, dan perangkat. Melihat permasalahan yang terjadi, penulis merancang sebuah aplikasi *Augmented Reality* dengan menggunakan metode *waterfall*. Penelitian dilakukan dengan mencari jarak minimum jangkauan kamera, sudut minimum jangkauan kamera, minimum intensitas cahaya terhadap *marker* pada tiga perangkat android yang akan menentukan alternatif yang optimal.

Kata kunci : *Augmented Reality*, 3D, Android, *waterfall*.

I. PENDAHULUAN

Latar Belakang

Di era yang modern saat ini, perkembangan teknologi tidak mungkin dihindari lagi. Perkembangan teknologi tersebut telah memasuki segala bidang kehidupan tidak terkecuali bidang pendidikan. Pada tahap pendidikan, anak-anak cenderung tertarik pada media pembelajaran yang didalamnya terdapat objek 3 dimensi (3D) yang menarik dan mudah untuk digunakan. Sedangkan pada saat ini metode pembelajaran masih banyak menggunakan media gambar dan tulisan berupa buku maupun alat peraga lainnya.

Augmented Reality atau disingkat AR adalah sebuah teknologi yang menggabungkan antara objek virtual dengan objek nyata. Menurut Ronald Azuma pada tahun 1997, *Augmented Reality* adalah variasi dari *Virtual Reality*. Teknologi *Virtual Reality* benar-benar membuat pengguna tenggelam dalam sebuah lingkungan sintetik. Ketika pengguna tenggelam dalam lingkungan tersebut, pengguna tidak bisa melihat dunia nyata. *Augmented Reality* mengijinkan pengguna untuk berinteraksi secara *real time* dengan sistem. *Augmented Reality* merupakan suatu konsep perpaduan antara

Virtual Reality dengan *World Reality*. Sehingga objek-objek *virtual* 2 Dimensi (2D) atau 3 Dimensi (3D) seolah-olah terlihat nyata dan menyatu dengan dunia nyata.

Dalam kasus ini, peneliti ingin memberikan pembelajaran yang menarik minat dan perhatian pengguna untuk meningkatkan kemauan belajar dengan menggunakan perkembangan teknologi *smartphone* android yang didalamnya terdapat materi pengenalan monumen dengan judul “*Analisa Augmented Reality Dalam Aplikasi Pengenalan Monumen Menggunakan Unity-3D*”.

Rumusan Masalah

Berdasarkan latar belakang diatas maka rumusan masalah yang dapat diambil adalah:

1. Bagaimana pengaruh jarak terhadap *marker*?
2. Bagaimana pengaruh cahaya terhadap *marker*?
3. Bagaimana pengaruh sudut terhadap *marker*?

Tujuan

Sesuai dengan permasalahan yang telah dijelaskan, maka tujuan yang akan dicapai adalah:

1. Mengetahui pengaruh jarak terhadap *marker*.
2. Mengetahui pengaruh cahaya terhadap *marker*.
3. Mengetahui pengaruh sudut terhadap *marker*.

Manfaat

Dengan adanya penelitian ini, diharapkan dapat menambah pengetahuan pengguna android tentang *Augmented Reality* dan menambah pemahaman pengguna tentang batasan-batasan penggunaan *Augmented reality*.

Batasan Masalah

Batasan masalah dalam pembuatan tugas akhir kali ini adalah:

1. Perangkat yang digunakan sistem operasi android.

2. Monumen yang dapat ditampilkan adalah objek yang sudah tersedia, yaitu:
 - a. Patung liberty
 - b. Pyramid
 - c. Menara eiffel
 - d. Colloseum
3. *Marker* yang digunakan adalah gambar yang sudah tersedia.

II. TINJAUAN PUSTAKA

AR (*Augmented Reality*)

Augmented reality merupakan upaya penggabungan dunia nyata dengan dunia *virtual* yang dibuat melalui komputer sehingga batas antara keduanya sangat tipis. Dengan teknologi *Augmented Reality* kita dapat menyisipkan suatu informasi tertentu ke dalam dunia maya dan menampilkannya di dunia nyata dengan bantuan perlengkapan seperti webcam, komputer, HP, android, maupun kacamata khusus.

Selain menambahkan benda maya dalam lingkungan nyata, *Augmented Reality* juga berpotensi menghilangkan benda-benda yang sudah ada. Menambah sebuah lapisan gambar maya dimungkinkan untuk menghilangkan atau menyembunyikan lingkungan nyata dari pandangan pengguna (Azuma, Ronald T, 1997).

Marker Augmented Reality

Aplikasi *Augmented Reality* ini berjalan dengan memindai tanda atau yang lebih sering disebut dengan *marker*. *Marker* biasanya merupakan ilustrasi hitam dan putih persegi dengan batas hitam tebal dan latar belakang putih.

System Development Life Cycle (SDLC)

Model SDLC *waterfall* merupakan proses pengembangan perangkat lunak tradisional yang umum digunakan dalam proyek-proyek perangkat lunak. Metode ini adalah model sekuensial, sehingga penyelesaian satu set kegiatan menyebabkan dimulainya aktivitas berikutnya. Model metode ini disebut

waterfall karena proses mengalir secara sistematis dari satu tahap ketahap lainnya (Imam, 2008).

Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah keluarga notasi grafis yang didukung oleh meta-model tunggal yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek.

Pemodelan visual didefinisikan sebagai proses pemodelan sistem informasi menggunakan pengaturan standar elemen grafik. Tujuan pemodelan visual adalah untuk memungkinkan adanya komunikasi antara pengguna, pengembang, penganalisis, *tester*, *manager*, dan siapapun yang terlibat dalam proyek, menunjukkan interaksi antara pengguna dengan sistem, obyek-obyek dalam sistem dan antar sistem sendiri (Sholih, 2006).

Unity 3D

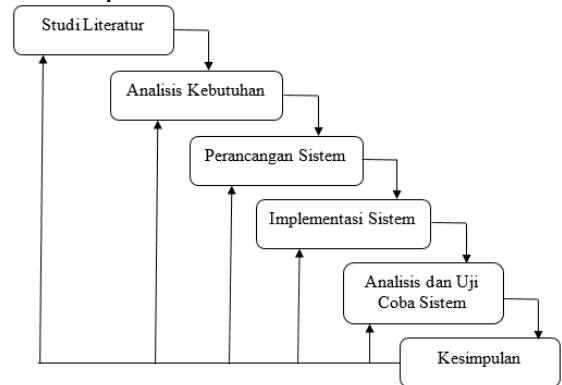
Unity 3D adalah *game engine* yang dikembangkan oleh *Unity Technologies* untuk mempermudah bagi para pengembang *game*. Dengan menggunakan unity 3D, pengembang *game* dapat membangun video *game* lebih cepat dan mudah dibanding sebelumnya. Di masa lalu, dalam membuat *game* dibutuhkan komputer yang memenuhi seluruh ruangan. Hari ini, dengan unity 3D pengembang *game* sudah bisa membuat sebuah *game* (Romi, 2003).

III. METODE PENELITIAN

3.1 Metodologi Penelitian

Adapun metode yang digunakan yaitu menggunakan metode SDLC (*System Development Life Cycle*) dengan model *Waterfall*. Dimana berisi rangkaian aktivitas proses yang teratur dan disajikan dalam proses yang terpisah, seperti spesifikasi kebutuhan, implementasi desain perangkat lunak, uji coba dan sebagainya

(Kristianto, 2004:11). Berikut alur penelitian yang digambarkan dalam diagram alur penelitian:



Gambar 3.1. diagram *waterfall* penelitian

3.1.1 Studi Literatur

Dalam penulisan penelitian ini penulis mencari informasi dan teori yang berhubungan dan sesuai dengan topik yang akan dibuat. Pencarian teori dan informasi akan didapatkan melalui buku-buku, internet, dan hasil penelitian maupun karya ilmiah yang terdahulu dengan topik yang sesuai. Nantinya bahan tersebut akan dibuat sebagai acuan dan batasan-batasan kegiatan untuk membuat aplikasi. Batasan yang dimaksud adalah konsep aplikasi yang sederhana, dan dapat di mengerti oleh pengguna.

3.1.2 Analisis Kebutuhan

Berikut adalah kebutuhan yang dibutuhkan oleh pengguna:

1. Media pembelajaran yang menarik.
2. *Smartphone* android untuk menjalankan aplikasi tersebut.
3. Media belajar yang interaktif.
4. Penggunaan gadget untuk media pembelajaran.

3.1.3 Rancangan Use Case Diagram Aplikasi

Tujuan dari pembuatan *use case* adalah untuk mendapatkan dan menganalisis informasi persyaratan yang cukup untuk mempersiapkan model yang mengkomunikasikan apa yang diperlukan dari perspektif pengguna. Dengan kata lain

use case diagram menekankan pada apa yang diperbuat oleh sistem. *Use case* diagram akan digambarkan sebuah interaksi antara aktor dengan sistem. Aktor sebagai pengguna atau orang yang akan menggunakan aplikasi edukasi. Pada tahap awal pengguna akan *download* dahulu *marker* yang akan digunakan untuk memunculkan animasi monumen, semisal gambar monumen borobudur yang muncul, maka pengguna dapat melihat animasi monumen tersebut. jika pengguna mengubah ke *marker* yang lain, maka animasi yang muncul akan berbeda. Berikut contoh alur yang digambarkan melalui *use case* diagram.



Gambar 3.2. Contoh *Use Case* aplikasi

3.1.4 Implementasi Sistem

Penulis melakukan pemfokusan pada pembuatan komponen dan fitur-fitur yang ada pada aplikasi edukasi pengenalan jenis monumen menggunakan teknik 3D. Penulis mengimplementasikan aplikasi yang telah dibangun dengan menggunakan data-data sehingga peneliti mengetahui apakah sistem tersebut bekerja secara maksimal sesuai dengan yang diharapkan.

3.1.5. Analisis dan Uji Coba Sistem

Uji coba dilakukan untuk mengetahui kelemahan dan kesalahan sebuah sistem yang telah dibuat sehingga dapat dikembangkan lebih lanjut untuk menjadi sebuah sistem dan aplikasi yang layak digunakan untuk tipe *handphone* yang sistem operasinya android. Pengujian yang nantinya dilakukan adalah pengisian *blackbox* untuk menguji keakuratan penelitian.

1. Lingkungan uji coba

Untuk mendukung aplikasi edukasi berbasis android, perlu adanya dukungan *hardware* maupun *software* yang memadai, sehingga pembangunan aplikasi yang akan dilakukan dapat menghasilkan aplikasi yang sesuai dengan kebutuhan yang ada. Berikut kebutuhan yang nantinya perlu sebagai pendukung :

a. Kebutuhan *hardware* (perangkat keras)

Adapun perangkat keras yang dibutuhkan dalam aplikasi yang dibangun dan diimplementasikan pada *smartphone* android sebagai berikut :

- 1) Processor Intel(R) Core i3-6100U (2.3 GHz, 3MB L3 Cache)
- 2) RAM 2.GB DDR4 Memory
- 3) Harddisk 500GB
- 4) NVIDIA GeForce 940MX 2GB VRAM

b. Kebutuhan *software* (perangkat lunak)

Perangkat lunak yang digunakan dalam pembuatan dan pengujian aplikasi edukasi pengenalan monumen ini yaitu :

- 1) Sistem operasi Microsoft windows 10
- 2) Unity-3D
- 3) Blender
- 4) Vuforia
- 5) Android Studio

2. Blackbox

Blackbox juga akan diisi untuk memperkuat hasil penelitian. Berikut adalah conth dari tabel pengujian *blackbox*:

Tabel 3.1. Tabel pengujian black box

| No | Skenario | Test Case | Harapan | Hasil |
|----|--------------|---|---|-------|
| 1 | Tombol | Menekan tombol-tombol yang ada pada aplikasi. | Tombol berjalan sesuai dengan fungsinya. | |
| 2 | Jarak kamera | Mengukur jarak kamera dengan <i>marker</i> . | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d. | |
| 3 | Sudut | Mengukur kamera dengan <i>marker</i> dari berbagai sudut. | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d. | |

| | | | | |
|---|---------------|--|--|--|
| 4 | Pencahaya | Mengukur dengan intensitas cahaya lebih. | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d. | |
| | | Mengukur dengan intensitas cahaya normal. | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d. | |
| | | Mengukur dengan intensitas cahaya kurang. | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d. | |
| 5 | Putaran | Menggerakkan kamera memutar <i>marker</i> . | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d sesuai pergerakan kamera. | |
| | | Memutar <i>marker</i> . | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d sesuai pergerakan <i>marker</i> . | |
| 6 | Oklusi | Menghalangi sebagian <i>marker</i> . | Kamera dapat menampilkan objek 3d. | |
| 7 | <i>Device</i> | Menjalankan aplikasi pada <i>device</i> dengan spesifikasi yang berbeda. | Aplikasi dapat berjalan pada <i>device</i> yang berbeda. | |

3.1.6. Tahap Kesimpulan

Tahapan ini merupakan bagian yang penting, dikarenakan semua ulasan penelitian dan uji coba akan disimpulkan untuk memperkuat hasil uji coba.

IV. IMPLEMENTASI DAN UJI COBA

4.1. Implementasi AR

Tahap pertama dalam pembuatan aplikasi ini adalah menentukan gambar yang akan dijadikan *marker* menggunakan aplikasi pengolah gambar. Dalam gal ini penulis menggunakan Adobe Photoshop CC2014.

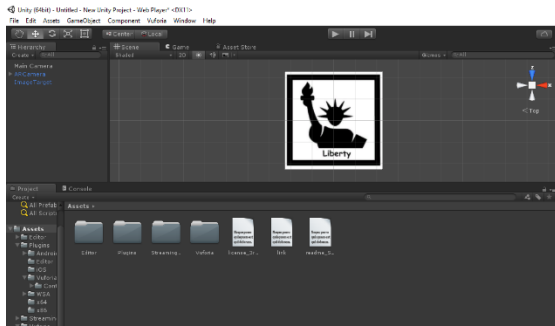


Gambar 4.1. Pembuatan *marker*

Lalu memproses gambar tersebut menjadi *package* yang dapat dideteksi oleh unity dengan mengupload ke *database target manager* vuforia. Untuk membuat *database*, daftar atau *register* terlebih dahulu pada *website* resmi vuforia (<https://developer.vuforia.com/>). Setelah itu *upload* gambar untuk dijadikan sebuah *marker*.

Gambar 4.2. Membuat *database*

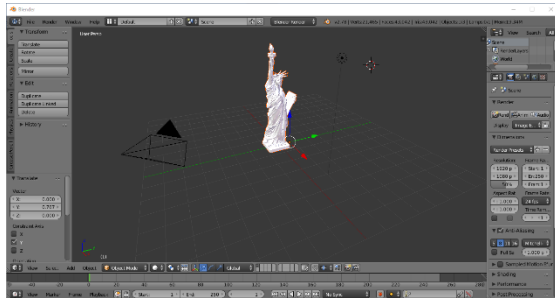
Selanjutnya *download database* yang sudah dibuat, lalu *import* kedalam unity.



Gambar 4.3. import database

4.1. Implementasi Objek Pada Image Target

Dalam pembuatan monumen 3 dimensi penulis men-download objek melalui internet yang nanti akan di import kedalam software Blender 2.78 untuk di export ke dalam format lain.



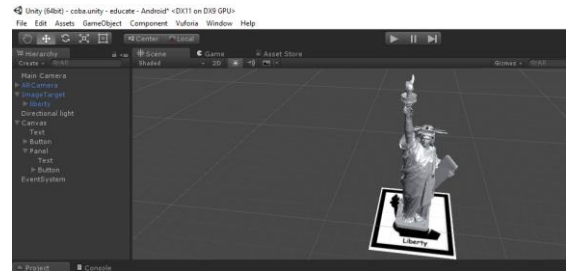
Gambar 4.4. user interface Blender 2.78

Langkah berikutnya adalah meng-export objek 3D yang sudah jadi ke dalam format fbx.

| Name | Date modified | Type | Size |
|--------------------|--------------------|----------|----------|
| 0.stl | 1/9/2013 6:24 PM | Objek 3D | 282 KB |
| 1.stl | 1/9/2013 6:24 PM | Objek 3D | 209 KB |
| 2.stl | 1/9/2013 6:24 PM | Objek 3D | 1,355 KB |
| 3.stl | 1/9/2013 6:24 PM | Objek 3D | 374 KB |
| bre_of_liberty.stl | 1/9/2013 6:24 PM | Objek 3D | 2,102 KB |
| liberty.obj | 3/16/2017 12:49 AM | OBJ File | 3,253 KB |
| untitled.mtl | 3/16/2017 12:49 AM | MTL File | 1 KB |
| liberty.fbx | 5/18/2017 9:52 AM | FBX File | 1,450 KB |

Gambar 4.5. file fbx

Tahap selanjutnya adalah memindahkan objek 3 dimensi ke dalam assets image target, dengan cara drag model 3 dimensi yang sudah di export ke dalam assets image target.



Gambar 4.6. drag objek 3 dimensi

4.2. Hasil Aplikasi

Aplikasi ini dibangun sebagai alat untuk menampilkan bentuk monumen secara 3 dimensi, dimana bentuk 3D ini akan ditampilkan pada sebuah marker atau gambar monumen yang ada pada gambar yang telah dibuat. Dengan dibangunnya aplikasi ini diharapkan dapat menambah pengetahuan kepada pengguna sehingga dengan penyajian informasi yang lebih nyata dapat menggugah minat pengguna untuk mempelajari augmented reality.



Gambar 4.7. tampilan aplikasi

Pada gambar 4.7 kotak nomor 1 adalah tampilan awal dari aplikasi augmented reality pengenalan monumen. Sedangkan pada kotak nomor 2 adalah tampilan aplikasi pada saat kamera menyorot marker.

4.3. Pengujian Aplikasi

Pengujian aplikasi ini menggunakan teknik pengujian blackbox. Dimana pada tahap ini peneliti akan menguji fungsionalitas dari fitur yang disediakan aplikasi.

Tabel 4.1. pengujian blackbox

| No | Skenario | Test Case | Harapan | Hasil |
|----|----------|---|--|-------|
| 1 | Tombol | Menekan tombol-tombol yang ada pada aplikasi. | Tombol berjalan sesuai dengan fungsinya. | Valid |

| | | | | |
|---|----------------|--|--|-------|
| 2 | Jarak kamera | Mengukur jarak kamera dengan <i>marker</i> . | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d. | Valid |
| 3 | Sudut | Mengukur kamera dengan <i>marker</i> dari berbagai sudut. | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d. | Valid |
| 4 | Pencahayaannya | Mengukur dengan intensitas cahaya lebih. | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d | Valid |
| | | Mengukur dengan intensitas cahaya normal. | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d. | Valid |
| | | Mengukur dengan intensitas cahaya kurang. | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d. | Valid |
| 5 | Putaran | Menggerakkan kamera memutar <i>marker</i> . | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d sesuai pergerakan kamera. | Valid |
| | | Memutar <i>marker</i> . | Kamera dapat mendeteksi <i>marker</i> dan menampilkan objek 3d sesuai pergerakan <i>marker</i> . | Valid |
| 6 | Oklusi | Menghalangi sebagian <i>marker</i> . | Kamera tidak dapat menampilkan objek 3d. | Valid |
| 7 | <i>Device</i> | Menjalankan aplikasi pada <i>device</i> dengan spesifikasi yang berbeda. | Aplikasi dapat berjalan pada <i>device</i> yang berbeda. | Valid |

Pengujian berikutnya adalah pengujian respon *time loading*, dimana jika aplikasi dijalankan pada *device smartphone* yang mempunyai spesifikasi berbeda, maka hasil respon *time* juga akan berbeda. Pengujian ini hanya akan dilakukan pada saat *loading* ke kamera *smartphone*, dimana proses ini yang akan menentukan perbedaan respon *time*. Pada pengujian aplikasi ini akan dilakukan pada 3 *device smartphone* yang berbeda.

Tabel 4.2. *Device* pengujian aplikasi

| a. <i>Device</i> 1 | b. <i>Device</i> 2 | c. <i>Device</i> 3 |
|--------------------------------|----------------------------|----------------------------|
| Xiaomi mi4 | Evercoss R50B | Asus T00Q |
| Spesifikasi: | Spesifikasi: | Spesifikasi: |
| Prosesor Quad Core Max 2.5 GHz | Prosesor Quad Core 1.3 GHz | Prosesor Dual Core 1.2 GHz |
| RAM 3 GB | RAM 1 GB | RAM 1 GB |
| Kamera 13MP | Kamera 8MP | Kamera 8MP |
| Resolusi layar 5.0 Inch | Resolusi layar 5.0 Inch | Resolusi layar 4.5 Inch |
| 1080 x 1920 pixels | 480 x 854 pixels | 480 x 854 pixels |
| Android v6.0 (marshmallow) | Android v6.0 (marshmallow) | Android v4.4.2 (kitkat) |

Pada tabel 4.3 dibawah ini merupakan hasil pengujian respon *time loading* kamera *smartphone*.

Tabel 4.3. hasil pengujian respon *time*

| No | Proses | Respon Time (s) | | |
|----|--|-----------------|-----------------|-----------------|
| | | <i>Device</i> 1 | <i>Device</i> 2 | <i>Device</i> 3 |
| 1 | <i>Loading</i> kamera menu <i>explore</i> monumen 3D | 3.12 | 7.28 | 17.69 |

Tahap berikutnya adalah pengujian jarak jangkauan kamera *smartphone* terhadap *marker* yang disorot.

Tabel 4.4. pengujian jarak jangkauan kamera

| No | Proses | Jarak Jangkauan (cm) | | |
|----|---|----------------------|------------|-------------|
| | | Device 1 | Device 2 | Device 3 |
| 1 | Jarak jangkauan kamera terhadap <i>marker</i> | 10.3 – 87.2 | 9.1 – 73.8 | 10.1 – 72.2 |

Pengujian berikutnya adalah pengaruh sudut penyorotan kamera terhadap *marker*.

Tabel 4.5. pengujian jangkauan sudut

| No | Proses | Jangkauan sudut (°) | | |
|----|---|---------------------|----------|----------|
| | | Device 1 | Device 2 | Device 3 |
| 1 | Jangkauan sudut kamera terhadap <i>marker</i> | 39 - 141 | 39 - 141 | 39 - 141 |

Pada pengujian intensitas cahaya ini peneliti menguji 3 *device* untuk menyorot *marker* dengan intensitas cahaya kurang, normal, dan lebih.

Tabel 4.6. pengujian intensitas cahaya

| No | Proses | Device 1 | Device 2 | Device 3 |
|----|--------------------------|----------|----------|----------|
| 1 | Intensitas cahaya kurang | Y | T | Y |
| 2 | Intensitas cahaya normal | Y | T | Y |
| 3 | Intensitas cahaya lebih | Y | Y | Y |

V. KESIMPULAN

5.1 Kesimpulan

1. Pada pengujian jarak jangkauan kamera terhadap *marker*, jarak ketiga *device* memiliki *range* minimal yang relatif sama yaitu 9cm – 10cm.

2. Pada pengujian jangkauan sudut kamera terhadap *marker*, ketiga *device* memiliki *range* jangkauan yang sama.
3. Pada pengujian intensitas cahaya, kamera sangat berpengaruh pada terdeteksinya *marker*.

5.2 Saran

Karena keterbatasan ilmu dan biaya yang tidak seberapa maka untuk tahap pengembangan selanjutnya ada beberapa hal yang menurut penulis perlu untuk ditambahkan ke aplikasi tersebut, antara lain :

1. Menambahkan objek penelitian yang lain.
2. Penambahan perangkat *platform* untuk sistem operasi yang lain.
3. Penambahan animasi pada objek 3D yang ditampilkan.

DAFTAR PUSTAKA

- Azuma. 1997. “*A Survei of Augmented Reality*”. Presence: Teleoperators and Virtual Environments
- Imam. 2008. “*Proses Pemodelan Software Dengan Metode Waterfall dan Extreme Programming: Studio Perbandingan*”. Jogja : Universitas Gadjah Mada
- Kristanto. 2004. “*Rekayasa Perangkat Lunak (Konsep Dasar)*”. Yogyakarta: Gava Media
- Romi. 2003. “*Pengantar Unified Modelling Language (UML)*”.
- Sholih. 2006. “*Pemodelan Sistem Informasi Berorientasi Objek dengan UML*”. Yogyakarta: Graha Ilmu