

**PENERAPAN TRANSFORMASI GEOMETRI
UNTUK VISUALISASI OBJEK MAYA
BERBASIS ANDROID**

SKRIPSI

Oleh:

MUH. ROBBI DINAK
NIM. 09650181



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK
IBRAHIM
MALANG
2016**

HALAMAN PENGAJUAN
PENERAPAN TRANSFORMASI GEOMETRI
UNTUK VISUALISASI OBEJEK MAYA
BERBASIS ANDROID

SKRIPSI

Diajukan kepada:

Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer

Oleh:

MUHAMMAD ROBBI DINAK
NIM. 09650181

JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2016

**HALAMAN PERSETUJUAN
PENERAPAN TRANSFORMASI GEOMETRI
UNTUK VISUALISASI OBJEK MAYA
BERBASIS ANDROID**

SKRIPSI

Oleh:

MUHAMMAD ROBBI DINAK
NIM. 09650181

Telah Diperiksa dan Disetujui untuk Diuji:

Tanggal, 30 Juni 2016

Dosen Pembimbing I,

Dosen Pembimbing II,


Dr. Cahyo Crvsdian
NIP. 19740424 200901 1 008


Roro Inda Melani, M.T., M.Sc.
NIP. 19780925 200501 2 008

Mengetahui:
Ketua Jurusan Teknik Informatika,


Dr. Cahyo Crvsdian
NIP. 19740424 200901 1 008

LEMBAR PENGESAHAN
PENERAPAN TRANSFORMASI GEOMETRI
UNTUK VISUALISASI OBJEK MAYA
BERBASIS ANDROID

SKRIPSI

Oleh:

MUH. ROBBI DINAK
NIM. 09650181

Telah Dipertahankan di Depan Dewan Penguji
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
pada Tanggal 30 Juni 2016

Susunan Dewan Penguji

- | | | |
|------------------|---|---|
| 1. Penguji Utama | : | <u>Irwan Budi Santoso, M.Kom</u>
NIP. 19770103 201101 1 004 |
| 2. Ketua | : | <u>M. Amin Harivadi, MT</u>
NIP. 19670118 200501 1 001 |
| 3. Sekretaris | : | <u>Dr. Cahyo Crvsdian</u>
NIP. 19740424 200901 1 008 |
| 4. Anggota | : | <u>Roro Ina Melani, MT., M.Sc</u>
NIP. 19780925 200501 2 008 |

Tanda Tangan

()
()
()
()

Disahkan Oleh:
Ketua Jurusan Teknik Informatika,



Dr. Cahyo Crvsdian
NIP. 19740424 200901 1 008

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Muhammad Robbi Dinak
NIM : 09650181
Jurusan : Teknik Informatika
Fakultas : Sains dan Teknologi
Judul Skripsi : **PENERAPAN TRANSFORMASI GEOMETRI UNTUK VISUALISASI OBJEK MAYA BERBASIS ANDROID**

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 30 Juni 2016

Yang Membuat Pernyataan,



Muh. Robbi Dinak
NIM. 09650181

HALAMAN MOTTO

إِنَّ مَعَ الْعُسْرِ يُسْرًا ﴿٦﴾ فَإِذَا فَرَغْتَ فَانصَبْ ﴿٧﴾ وَإِلَىٰ رَبِّكَ فَارْغَب ﴿٨﴾

Artinya: “Sesungguhnya sesudah kesulitan itu ada kemudahan. Maka apabila kamu telah selesai (dari suatu urusan), kerjakanlah dengan sungguh-sungguh (urusan) yang lain, dan hanya kepada Tuhanmulah hendaknya kamu berharap.” (QS. Al-Insyirah: 6-8)

“Kesuksesan adalah buah dari kesabaran, usaha keras dan doa”



HALAMAN PERSEMBAHAN

Segala puji bagi Allah Tuhan semesta alam

Kupersembahkan sebuah karya kecil ini untuk orang-orang terkasih:

Bapak dan Ibuku tercinta

Sugiarto dan Nu'em,

Kalianlah sumber semangatku,

Terima kasih atas kasih sayang dan doa yang tak pernah putus untukku.

Adikku, Burhanuddin Al-Hafidzi dan Rizki Mubarak

Kamu sumber inspirasiku.

Seluruh keluargaku, atas semua doa dan dukungannya.

Teman-teman almamater, terima kasih telah berjuang bersama hingga akhir waktu kita.

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT. karena atas rahmat dan karunia-Nya penulis dapat menyelesaikan penelitian skripsi dengan judul **“Penerapan Transformasi Geometri Untuk Visualisasi Objek Maya Berbasis Android”**. Skripsi ini merupakan prasyarat untuk memenuhi syarat utama kelulusan program pendidikan Strata 1 pada Jurusan Teknik Informatika, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Banyak sekali rintangan dan kesulitan yang harus penulis hadapi dalam menyelesaikan skripsi ini. Akan tetapi dengan banyaknya dorongan dari semua pihak yang senantiasa memberikan dukungan dan semangat, skripsi ini pun dapat terselesaikan dengan baik. Oleh karena itu, penulis ucapkan terima kasih yang sebesar-besarnya kepada semua pihak yang turut membantu dalam penyelesaian skripsi ini baik secara langsung maupun tidak langsung.

Dalam kesempatan ini, penulis ingin mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Dr. Cahyo Crysdiان selaku Ketua Jurusan Teknik Informatika UIN Maliki yang telah mendukung semua proses penelitian penulis serta selaku dosen pembimbing I yang telah memberikan bimbingan serta saran-saran dalam proses penyelesaian penelitian dan penyusunan laporan skripsi ini.
2. Dosen pembimbing II yang telah memberikan masukan dan bimbingannya dalam proses penyusunan laporan skripsi ini.

3. Dr. Suhartono, M. Kom. selaku dosen wali yang telah memberikan bimbingannya selama penulis menjadi mahasiswa walinya.
4. Seluruh staf dosen dan admin Jurusan Teknik Informatika UIN Maliki yang senantiasa memberikan ilmu dan bantuan dalam proses pembelajaran selama penulis kuliah di UIN Maliki ini.
5. Rekan-rekan Teknik Informatika terutama angkatan 2009 yang senantiasa berbagi ilmu dalam proses perkuliahan dan berjuang bersama selama menjadi mahasiswa.
6. Terakhir kepada seluruh pihak yang telah membantu dalam penyelesaian skripsi ini yang tidak bisa penulis sebutkan satu persatu.

Akhir kata, semoga Allah SWT. membalas segala kebaikan yang telah diterima penulis. Kata maaf penulis ucapkan atas segala kekurangan dan keterbatasan ini. Penulis berharap semoga laporan skripsi ini dapat bermanfaat bagi seluruh pihak yang membutuhkan.

Malang, 30 Juni 2016

Penulis

DAFTAR ISI

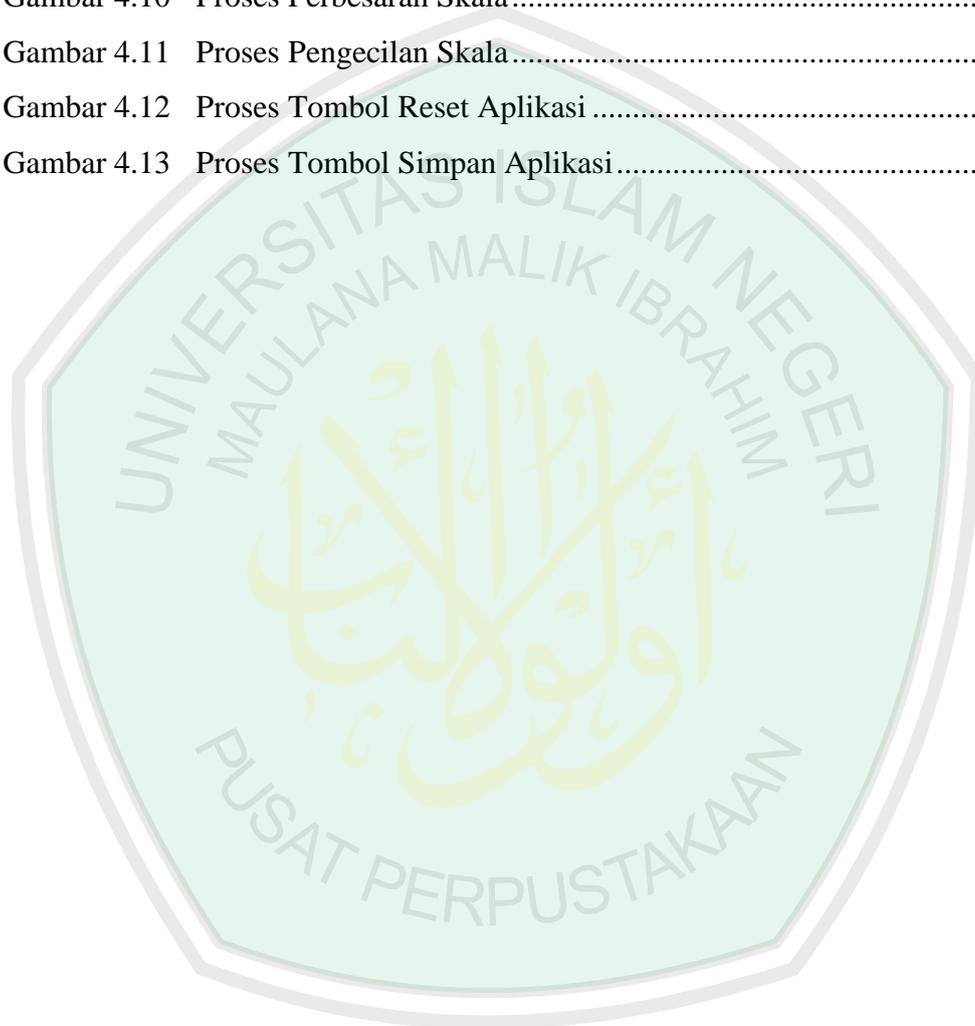
HALAMAN JUDUL.....	i
HALAMAN PENGAJUAN.....	ii
HALAMAN PERSETUJUAN.....	iii
LEMBAR PENGESAHAN	iv
SURAT PERNYATAAN.....	v
HALAMAN MOTTO	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR	xii
DAFTAR TABEL.....	xiv
DAFTAR SOURCE CODE	xv
ABSTRAK	xvi
ABSTRACT.....	xvii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan Penelitian	2
1.4 Manfaat Penelitian	2
1.5 Batasan Masalah.....	3
1.6 Sistematika Penulisan	3
BAB IIKAJIAN PUSTAKA.....	5
2.1 Penelitian Terkait	5
2.2 Visualisasi	6
2.3 Pemodelan 3D	8
2.4 Transformasi Geometri	9
2.4.1 Translasi	10
2.4.2 Rotasi.....	11
2.4.3 Skala.....	11

2.5	Penggunaan Min3D dan OpenGL pada Android.....	12
BAB III DESAIN SISTEM.....		14
3.1	Perancangan Sistem	14
3.3.1	Flowchart Sistem.....	14
3.3.2	Use Case Diagram.....	17
3.3.3	Activity Diagram.....	18
3.4	Perancangan Pembuatan Objek Maya.....	23
3.5	Desain User Interface.....	27
3.5.1	Storyboard Aplikasi.....	27
3.5.2	Desain Interface Splash Screen	28
3.5.3	Desain Interface Halaman Menu Objek.....	29
3.5.2	Desain Interface Halaman Utama Aplikasi.....	29
BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM.....		32
4.1	Implementasi Sistem	32
4.1.1	Implementasi Interface.....	33
4.1.2	Implementasi Kode Program Rancangan Kerja Sistem	35
4.2	Pengujian Sistem.....	47
4.2.1	Pengujian Splash Screen	47
4.2.2	Pengujian Menu Objek.....	48
4.2.3	Pengujian Halaman Utama.....	49
4.2.4	Pengujian Visualisasi Objek Maya Pada Kamera.....	50
4.2.5	Pengujian Transformasi Geometri	51
4.2.6	Pengujian Tombol Reset	53
4.2.7	Pengujian Tombol Simpan.....	54
4.4	Hasil Pengujian	55
4.4	Integrasi Nilai Islam.....	57
BAB V PENUTUP.....		67
5.1	Kesimpulan	67
5.2	Saran.....	67
DAFTAR PUSTAKA		69

DAFTAR GAMBAR

Gambar 2.1	Proses Translasi.....	10
Gambar 2.2	Proses Rotasi	11
Gambar 3.1	Diagram Blok Aplikasi	14
Gambar 3.2	Flowchart Splash Screen Aplikasi	14
Gambar 3.3	Flowchart Umum Sistem Aplikasi.....	15
Gambar 3.4	Flowchart Menampilkan Objek 3D.....	16
Gambar 3.5	Use Case Diagram Aplikasi	18
Gambar 3.6	Activity Diagram Splash Screen	19
Gambar 3.7	Activity Diagram Memilih Objek	20
Gambar 3.8	Activity Diagram Translasi Objek Geometri	20
Gambar 3.9	Activity Diagram Rotasi Objek Geometri.....	21
Gambar 3.10	Activity Diagram Scala Objek Geometri	22
Gambar 3.11	Activity Diagram Reset Objek Geometri	22
Gambar 3.12	Activity Diagram Simpan Objek Geometri.....	23
Gambar 3.13	Alur Pembuatan Objek.....	24
Gambar 3.14	Menu Draw Pada Google SketchUp	25
Gambar 3.15	Peletakan Objek Geometri	25
Gambar 3.16	Push dan Pull Objek Geometri.....	26
Gambar 3.17	Pemodelan Objek Geometri	26
Gambar 3.18	Pemberian warna dan Lighting	27
Gambar 3.19	Eksport Objek 3 Dimensi	27
Gambar 3.20	Desain Interface Splash Screen Aplikasi	29
Gambar 3.21	Desain Interface Menu Utama Aplikasi.....	30
Gambar 3.22	Desain Interface Halaman Utama Aplikasi.....	30
Gambar 4.1	Tampilan Layout Splash Screen Aplikasi	34
Gambar 4.2	Tampilan Layout Menu Objek dan Halaman Utama Aplikasi.....	35
Gambar 4.3	Tampilan Proses Splash Screen	48
Gambar 4.4	Tampilan Menu Objek Sebelum Diklik	49
Gambar 4.5	Tampilan Menu Objek Setelah Diklik dan Scroll	49

Gambar 4.6	Tampilan Proses Halaman Utama.....	50
Gambar 4.7	Proses Rendering Objek Pada Kamera	50
Gambar 4.8	Proses Pengujian Translasi.....	51
Gambar 4.9	Proses Rotasi Terhadap Sumbu x.....	52
Gambar 4.10	Proses Perbesaran Skala.....	53
Gambar 4.11	Proses Pengecilan Skala.....	53
Gambar 4.12	Proses Tombol Reset Aplikasi	54
Gambar 4.13	Proses Tombol Simpan Aplikasi.....	55



DAFTAR TABEL

Tabel 3.1	Storyboard Aplikasi Visualisasi Objek Maya Berbasis Android	28
Tabel 4.1	Spesifikasi perangkat.....	33
Tabel 4.2	Hasil Pengujian Black Box	55



DAFTAR SOURCE CODE

Source Code 4.1	Kode Program Splash Screen pada AndroidManifest.....	36
Source Code 4.2	Kode Program Splash Screen pada MainActivity.....	37
Source Code 4.3	Kode Program Inisialisasi Menu Objek	37
Source Code 4.4	Kode Program Penambahan Objek	38
Source Code 4.5	Kode Program Set Default Objek	38
Source Code 4.6	Kode Program List Nama Objek.....	39
Source Code 4.7	Kode Program Tombol View Objek	39
Source Code 4.8	Kode Program Pembuatan Ukuran Kamera.....	40
Source Code 4.9	Kode Program Parameter dan Pengaturan Kamera.....	41
Source Code 4.10	Kode Program Menampilkan View Kamera.....	42
Source Code 4.11	Kode Program Merubah Format Image	43
Source Code 4.12	Kode Program Rendering dan Overlay Objek	43
Source Code 4.13	Kode Program Proses Awal Translasi.....	44
Source Code 4.14	Kode Program Aksi Translasi	45
Source Code 4.15	Kode Program Transformasi Rotasi.....	46
Source Code 4.16	Kode Program Transformasi Skala	46
Source Code 4.17	Kode Program Aksi Tombol Skala	47

ABSTRAK

Robbi Dinak, Muhammad. 2016. **Penerapan Transformasi Geometri Untuk Visualisasi Objek Maya Berbasis Android**. Jurusan Teknik Informatika Fakultas Sains Dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang, Pembimbing: (I) Dr. Cahyo Crys dian dan (II) Roro In da Melani, MT., M.Sc.

Kata Kunci : *Transformasi Geometri, Visualisasi Objek Maya, Android, Library Min3d*

Perkembangan teknologi di bidang grafika computer begitu pesat. Terbukti dengan banyaknya aplikasi yang menggunakan grafika komputer antara lain video game, CAD, dan simulator. Grafika komputer sendiri tidak terlepas dari transformasi geometri. Dengan transformasi geometri, suatu objek geometri dapat dipindahkan dan diubah bentuknya. Masalah yang timbul, yaitu bagaimana mengimplementasikan transformasi geometri tersebut ke dalam suatu aplikasi. Transformasi geometri yang dapat dilakukan oleh aplikasi ini berupa translasi, rotasi, dan penskalaan. Selain itu, aplikasi tersebut dapat digunakan dengan mudah oleh pengguna untuk memvisualisasikan objek maya dan mentransformasikannya. Agar menghasilkan suatu aplikasi yang dapat mengimplementasikan transformasi geometri, aplikasi dibuat dengan bahasa pemrograman Android Studio. Library Min3D digunakan untuk menampilkan objek maya yang dibuat dan Android Studio untuk membuat antarmuka pengguna dan fungsi-fungsi transformasi. Sebagai implementasi, aplikasi dibuat dalam bentuk aplikasi Android yang dapat menerapkan transformasi 3 dimensi pada objek geometri. Pengujian dilakukan dengan melihat apakah aplikasi yang dibuat berjalan baik dalam artian dapat divisualisasikan dan ditransformasikan dengan tepat. Hasil yang diperoleh dari pengujian black box memperlihatkan bahwa program secara keseluruhan berjalan dengan baik, walaupun walaupun masih diperlukan beberapa perbaikan di beberapa bagian.

ABSTRACT

Robbi Dinak, Muhammad. 2016. **Implementation of Geometric Transformation for Visualization Virtual Object Based Android.** Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang, Supervisor: (I) Dr. Cahyo Crysdiand dan (II) Roro Inda Melani, MT., M.Sc.

Keywords : *Geometric Transformation, Visualization Virtual Object, Android, Min3D Library*

Computer graphics technology grows rapidly. It can be seen by many application that use computer graphics such as video game, CAD, and simulator. Computer graphics itself are not free from geometric transformation. By geometric transformation, a geometric object can be moved and its shape can be changed. The problem that appears is how to implement geometric transformation into an application. Geometric transformations that can be done by the application are translation, rotation, and scaling. Besides that, the application can be used easily by user to visualize virtual objects and transform them. In order to produce an application that can implement geometric transformation, the application is made by using Android Studio programming language. Min3D library is used to show virtual objects and Android Studio is used to make user interface and functions of transformation. As implementation, application is made in an Android form that can do 3 dimension transformation on geometric objects. The testing is performed by seeing whether the application doing well, its means can virtualize objects and transform correctly. The results coordinates from the black box testing show a whole application doing well, although there is need some slight improvement in some part.

الملخص

رابي ديناك، محمد. ٢٠١٦. تطبيق التحول الهندسة لتصوركائنات مايا القائم على الروبوت. قسم المعلوماتية كلية العلوم والتكنولوجيا جامعة الإسلامية الحكومية مولانا مالك إبراهيم مالانج، المشرف (١) الدكتور جهيو كريسديان، الماجستير و (٢) رورو إندا ميلاني، الماجستير

كلمات الرئيسية: التحول الهندسة، التصور كائن الظاهري ، الروبوت، *Library Min3d*

التطورات التكنولوجية في مجال رسومات الحاسوب بسرعة جدا. يتضح من العديد من التطبيقات التي تستخدم رسومات الحاسوب يعنى الألعاب الفيديو، CAD، وجهاز محاكاة. رسومات الكمبيوتر لا يتم فصل أنفسهم من التحول الهندسة. مع تحول للهندسة، هندسة كائن يمكن نقلها وتغير الشكل. المشكلة هي كيفية تنفيذ التحول الهندسة في تطبيق. التحول الهندسة الذي يمكن أن تقوم به هذا التطبيق مثل الترجمة، والتناوب، والتحجيم. وبالإضافة إلى ذلك، يمكن للتطبيق استخدامها بسهولة المستخدم لتصور الكائن الظاهري والتحويل. من أجل إنتاج التطبيقات التي يمكن أن تنفذ التحول الهندسة ، والتطبيقات التي تم إنشاؤها مع لغة برمجة الروبوت الاستوديو. *Library Min3d* تستخدم لعرض الأشياء الافتراضية التي تم إنشاؤها والروبوت الاستوديو لإنشاء واجهة المستخدم وظائف التحول. للتنفيذ، يتم تقديم الطلب في شكل تطبيقات الروبوت التي يمكن تنفيذ التحول من ثلاثة وجوه هندسة الأبعاد. ويتم اختبار من خلال رؤية ما إذا كان التطبيق جعلت سواء من حيث تشغيل يمكن تصور وتحويلها بشكل مناسب. النتائج التي تم الحصول عليها من اختبار الصندوق الأسود يدل على أن البرنامج لكل سارت بشكل جيد، مازال هناك حاجة بعض التحسينات في بعض الأجزاء

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Perkembangan teknologi di bidang grafika komputer begitu pesat. Terbukti dengan banyaknya aplikasi yang menggunakan grafika komputer antara lain video game, CAD, dan simulator. Grafika komputer sendiri tidak terlepas dari transformasi geometri, dengan transformasi geometri, suatu objek geometri dapat dipindahkan dan diubah bentuknya. Banyak sekali teknologi baru baik itu perangkat keras maupun perangkat lunak yang memanfaatkan transformasi geometri, khususnya pada bidang visual yaitu suatu aplikasi terbaru pada sistem komputer yang menggabungkan antara objek maya ke dalam sebuah lingkungan nyata dan menampilkannya dalam waktu nyata.

Perkembangan teknologi smartphone sendiri juga tidak kalah pesatnya. Ponsel yang canggih memiliki banyak sekali keunggulan dalam fitur-fiturnya, yang mana masing-masing fitur dapat menghasilkan hasil yang luar biasa jika dimanfaatkan dengan baik. Android Studio memiliki fitur dalam merealisasikan objek 3 dimensi pada Android dengan menggunakan fitur library yang ada pada Android Studio (salah satunya library Min3D dan OpenGL), objek 3 dimensi yang telah dibuat sebelumnya dapat ditampilkan dan direalisasikan pada layar kamera.

Aplikasi ini nantinya akan dibuat dengan mengintegrasikan software Google SketchUp dalam pembuatan objek dan Android Studio

dalam perealisasiannya, serta teknik transformasi geometri untuk penyajian objeknya. Atas dasar pemikiran itulah peneliti berharap dapat menghasilkan suatu aplikasi berbasis Android yang dapat melakukan transformasi geometri pada objek maya yang dibuat dan juga berharap aplikasi yang dibuat memiliki antarmuka yang mudah digunakan (*user friendly*) sehingga mempermudah pentransformasian pada objek maya.

1.2 Rumusan Masalah

Berdasarkan latar belakang pada rumusan masalah, dapat ditarik permasalahan yakni:

1. Bagaimana memvisualisasikan objek maya pada lingkungan nyata?
2. Bagaimana mengimplementasikan metode transformasi geometri untuk visualisasi objek maya?

1.3 Tujuan Penelitian

Tujuan yang ingin dicapai dari penelitian ini adalah untuk memvisualisasikan objek maya pada lingkungan nyata dengan memanfaatkan metode transformasi geometri dalam penyajian objeknya.

1.4 Manfaat Penelitian

Hasil dari penelitian ini diharapkan bisa memberikan manfaat diantaranya, yaitu :

1. Mengetahui bahwa aplikasi yang dibuat dapat memvisualisasikan objek maya pada gadget berplatform Android.

2. Mengetahui bahwa aplikasi yang dibuat dapat melakukan transformasi geometri pada objek maya, serta sebagai dasar pengembangan aplikasi visualisasi objek maya pada penelitian selanjutnya.

1.5 Batasan Masalah

Agar penelitian yang dilakukan tidak menyimpang, maka diperlukan adanya batasan masalah, yaitu :

1. Hanya dapat digunakan pada gadget berplatform Android yang memiliki layar sentuh.
2. Pemilihan benda dan warna objek yang terbatas pada keinginan peneliti.
3. Hanya dapat melakukan 3 transformasi dasar geometri yakni translasi, rotasi dan skala.

1.6 Sistematika Penulisan

Penulisan skripsi ini disusun dalam lima bab dengan sistematika penulisan sebagai berikut:

- **BAB I** Pendahuluan

Pada bab ini berisi tentang latar belakang penulisan skripsi, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah dan sistematika penulisan.

- **BAB II** Kajian Pustaka

Berisi tentang teori yang berkaitan dengan objek permasalahan yang diambil dan penelitian-penelitian terkait yang pernah dilakukan oleh peneliti sebelumnya.

- **BAB III Desain Sistem**

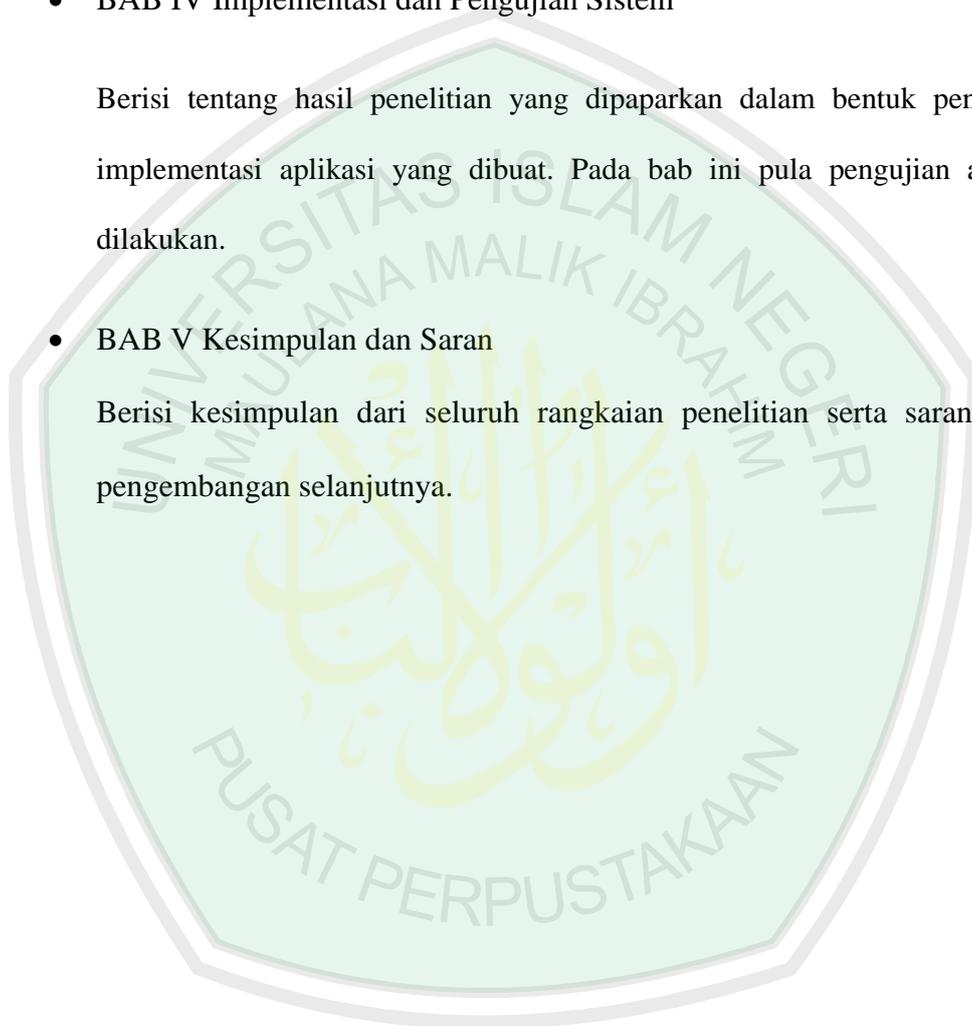
Pada bab ini akan dijelaskan mengenai tahap perancangan dalam mendesain sistem aplikasi akan dibuat.

- **BAB IV Implementasi dan Pengujian Sistem**

Berisi tentang hasil penelitian yang dipaparkan dalam bentuk penjelasan implementasi aplikasi yang dibuat. Pada bab ini pula pengujian aplikasi dilakukan.

- **BAB V Kesimpulan dan Saran**

Berisi kesimpulan dari seluruh rangkaian penelitian serta saran untuk pengembangan selanjutnya.



BAB II

KAJIAN PUSTAKA

2.1 Penelitian Terkait

Putranto (2011) dalam penelitiannya meneliti tentang visualisasi 3 dimensi struktur rangka pada manusia. Penelitian tersebut bertujuan untuk menghadirkan sebuah aplikasi pembelajaran multimedia yang berbentuk perangkat ajar untuk radiographer yang menghadirkan model 3 dimensi yaitu struktur rangka manusia. Dalam penelitian tersebut teknik pemodelan yang digunakan dalam membuat animasi 3 dimensi menggunakan teknik polygonal modeling, yang mana teknik tersebut dalam perancangan model 3 dimensi yang ditampilkan memberikan detail dan tekstur yang lebih mudah pada objek 3 dimensi.

Yohannis (2006) dalam penelitiannya meneliti tentang aplikasi grafika komputer untuk transformasi geometri 3 dimensi. Dalam penelitian tersebut peneliti mengimplementasikan transformasi geometri dengan menggunakan bahasa pemrograman Delphi dan API OpenGL. Di mana OpenGL digunakan untuk menampilkan objek 3 dimensi dan Delphi sendiri digunakan untuk membuat antarmuka pengguna. Aplikasi tersebut dibuat dalam bentuk aplikasi studio yang dapat menerapkan transformasi 3 dimensi pada objek geometri. Pengujiannya dilakukan dengan membandingkan hasil transformasi aplikasi dan hasil transformasi melalui perhitungan untuk melihat apakah aplikasi dapat mentransformasi dengan tepat. Hasil yang di dapat pada penelitian tersebut yakni koordinat-koordinat hasil transformasi aplikasi sama dengan hasil perhitungan. Pengujian juga dilakukan dengan membandingkan framerate antara fungsi

transformasi aplikasi dan fungsi transformasi bawaan OpenGL. Hasilnya, fungsi transformasi bawaan OpenGL memiliki framerate lebih tinggi dibandingkan fungsi aplikasi.

Shirley (2011) dalam penelitiannya meneliti tentang real time augmentation of video with 3D graphics. Dalam penelitian tersebut peneliti menggunakan library Min3D untuk menampilkan model 3 dimensi ke dalam aplikasi. Library Min3D di sini berfungsi untuk mengimport model dan tekstur dari berbagai format termasuk file .obj dan file .3ds, sedangkan OpenGL (Open Graphic Library) digunakan untuk merender objek dan meletakkannya pada layar kamera.

2.2 Visualisasi

Visual berhubungan erat dengan mata dan penglihatan. Menurut beberapa ahli, visual juga merupakan salah satu bagian dari aktivitas belajar, dimana aktivitas belajar itu sendiri terdiri atas: somatic (belajar dengan bergerak dan berbuat), auditori (belajar dengan berbicara dan mendengar), intelektual (belajar dengan memecahkan masalah dan merenung), dan visual (belajar dengan cara melihat, mengamati, dan menggambarkan). Keempat aktivitas belajar tersebut harus dikuasai supaya proses belajar dapat berlangsung secara optimal.

Visualisasi adalah rekayasa dalam pembuatan gambar, diagram atau animasi untuk penampilan suatu informasi. Secara umum, visualisasi dalam bentuk gambar baik yang bersifat abstrack maupun nyata telah dikenal sejak awal dari peradaban manusia. Contoh dari hal ini meliputi lukisan di dinding-dinding

gua dari manusia purba, bentuk huruf hieroglif Mesir, system geometri Yunani, dan teknik pelukisan dari Leonardo da Vinci untuk tujuan rekayasa dan ilmiah, dll (Herbert dan James, 1998).

Pada Saat ini visualisasi telah berkembang dan banyak dipakai untuk keperluan ilmu pengetahuan, rekayasa, visualisasi desain produk, pendidikan, multimedia interaktif, kedokteran, dll. Pemakaian dari grafika computer merupakan perkembangan penting dalam dunia visualisasi, setelah ditemukannya teknik garis perspektif pada zaman Renaissance.

Visualisasi 2 dimensi yang lebih dikenal dengan 2D atau bidang, merupakan bentuk dari benda yang memiliki panjang dan lebar. Penggambarannya dilakukan dengan berpatokan pada titik sumbu x (datar) dan sumbu y (tegak). Koordinat tersebut dihitung mulai dari sudut kiri atas layar. Semakin ke kanan, nilai koordinat x semakin bertambah. Semakin ke bawah, nilai koordinat y semakin bertambah. Besarnya nilai koordinat ini dihitung dalam satuan pixel. Agar dapat tampil dengan sempurna, gambar yang akan ditampilkan harus memiliki nilai koordinat x dan y minimum 0 dan maximum sebesar resolusi yang digunakan.

Visualisasi 3 dimensi yang dikenal dengan 3D atau ruang adalah bentuk dari benda yang memiliki panjang, lebar, dan tinggi. Penggambarannya tidak jauh beda dengan 2 dimensi yakni titik koordinatnya berpatoakn pada sumbu x dan sumbu y, tetapi juga memiliki sumbu lain yang arahnya tegak lurus (menembus) bidang atau bias disebut dengan sumbu z.

2.3 Pemodelan 3D

Pemodelan adalah membentuk suatu benda-benda atau objek. Membuat dan mendesain objek tersebut sehingga terlihat seperti hidup. Sesuai dengan objek dan basisnya, proses ini secara keseluruhan dikerjakan di komputer. Melalui konsep dan proses desain, keseluruhan objek bias diperlihatkan secara 3 dimensi, sehingga banyak yang menyebut hasil ini sebagai pemodelan 3 dimensi (Nalwan, 1998).

Ada beberapa aspek yang harus dipertimbangkan bila membangun model objek, kesemuanya memberi kontribusi pada kualitas hasil akhir. Hal-hal tersebut meliputi metode untuk mendapatkan atau membuat data yang mendeskripsikan objek, tujuan dari model, tingkat kerumitan, perhitungan biaya, kesesuaian dan kenyamanan, serta kemudahan manipulasi model.

Proses pemodelan 3D membutuhkan perancangan yang dibagi dengan beberapa tahapan untuk pembentukannya, antara lain :

- a. Objek apa yang ingin dibentuk sebagai objek dasar
- b. Metode pemodelan objek 3D

Teknik *Polygonal modeling* adalah teknik membuat model dengan memakai objek-objek *geometry* dasar yang kemudian dikembangkan menjadi objek model yang lebih kompleks. Teknik *polygonal modeling* ini lazimnya adalah menggunakan objek dasar pada standar *primitive geometry* dan kemudian dimodifikasi menjadi objek. Cara menggunakan teknik ini mulanya objek standar *primitive geometry* terlebih dahulu dikonversi menjadi *editable mesh* atau *editable*

poly yang kemudian dimanipulasi atau dilakukan pengeditan pada vertex, edge, face, poly, border, atau element yang dapat digunakan untuk membentuk suatu objek sesuai keinginan.

c. Pencahayaan dan animasi gerakan objek sesuai dengan urutan proses yang akan dilakukan.

Ada 3 teknik pencahayaan pada 3 dimensi, antara lain:

1. Lampu utama (*Key light*)

Key light merupakan pencahayaan utama dari gambar kita, dan merepresentasikan bagian paling terang sekaligus mendefinisikan bayangan pada gambar. *Key light* juga merepresentasikan pencahayaan paling dominan seperti matahari dan lampu interior. Meski demikian

2. Lampu pengisi (*Fill light*)

Berfungsi untuk melembutkan sekaligus mengisi bagian gelap yang diciptakan oleh *Key light*. *Fill light* juga berfungsi menciptakan kesan 3 dimensi.

3. Cahaya latar (*Back light*)

Back light berfungsi untuk menciptakan pemisahan antara objek utama dengan objek pendukung. Dengan diletakkan pada bagian belakang benda *back light* menciptakan “garis pemisah” antara objek utama dengan latar belakang pendukungnya.

2.4 Transformasi Geometri

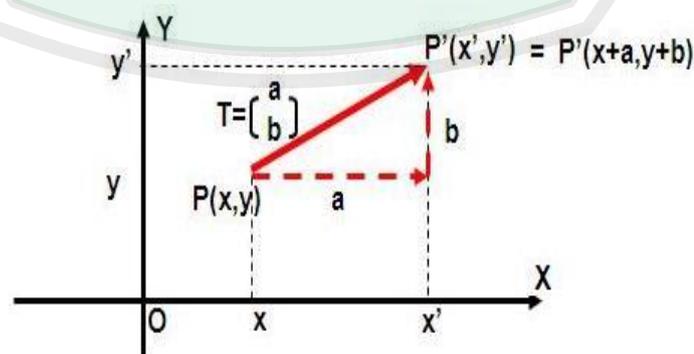
Transformasi merupakan suatu metode untuk mengubah lokasi suatu titik pembentuk objek, sehingga objek tersebut mengalami perubahan. Perubahan

objek dengan mengubah koordinat dan ukuran suatu objek disebut dengan transformasi geometri. Transformasi geometri adalah operasi yang diberikan pada gambaran geometri dari suatu objek untuk mengubah posisinya, orientasinya atau ukurannya (Hearn dan Baker, 2004). Jadi setiap operasi yang dapat mengubah posisi, orientasi, dan ukuran dari gambaran objek geometri dapat disebut sebagai transformasi geometri. Ada tiga transformasi dasar yang dapat dilakukan terhadap vertex, yakni translasi, penskalaan, dan rotasi.

2.4.1 Translasi

Translasi adalah transformasi yang memindahkan setiap titik pada bidang menurut jarak dan arah tertentu. Transformasi translasi juga merupakan operasi yang menyebabkan perpindahan objek tiga dimensi dari satu tempat ke tempat yang lainnya. Translasi T dapat dinyatakan dalam bentuk pasangan terurut dua bilangan ab dan dituliskan sebagai: $T = ab$.

Jadi bayangan titik $P(x,y)$ oleh translasi $T=ab$ adalah $P'(x',y')$ dengan $x'=x+a$ dan $y'=y+b$.



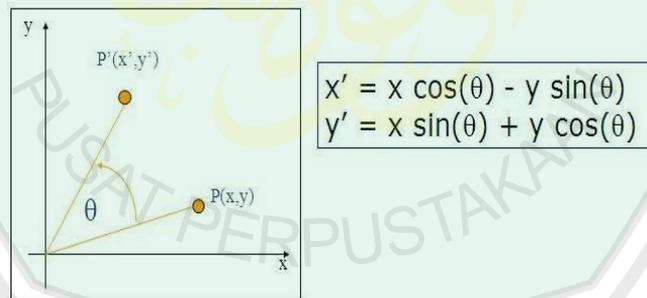
Gambar 2.1 Proses Translasi

2.4.2 Rotasi

Rotasi adalah transformasi yang memetakan setiap titik pada bidang ketitik lainnya dengan cara memutar pada pusat titik tertentu. Transformasi rotasi dilakukan dengan memindahkan semua titik-titik dari suatu objek ke posisi yang baru dengan memutar titik-titik tersebut dengan sudut dan sumbu putar yang ditentukan (Hearn dan Baker, 2004).

Arah rotasi disepakati dengan aturan bahwa jika perputaran berlawanan dengan arah jarum jam, maka rotasi bernilai positif. Sedangkan jika perputaran searah dengan jarum jam, maka rotasi bernilai negatif.

Bayangan titik $P(x,y)$ yang dirotasikan terhadap pusat $O(0,0)$ sebesar θ adalah $P'(x',y')$ dengan $X' = x \cos \theta - y \sin \theta$ dan $Y' = x \sin \theta + y \cos \theta$.



Gambar 2.2 Proses Rotasi

2.4.3 Skala

Transformasi penskalaan merupakan transformasi yang berfungsi mengubah ukuran suatu objek (Hearn dan Baker, 2004). Jadi skala adalah transformasi yang mengubah ukuran atau skala suatu bangun geometri

(perbesaran/pengecilan), tetapi tidak mengubah bentuk bangunan tersebut. Bayangan titik $P(x,y)$ oleh dilatasi $[O,k]$ adalah $P'(x',y')$ dengan $X' = kx$ dan $Y' = ky$

2.5 Penggunaan Min3D dan OpenGL pada Android Studio

Library Min3D merupakan framework open source untuk Android yang digunakan untuk mengimport dan memuat model 3 dimensi ke dalam aplikasi. Library Min3D memungkinkan pengimportan model dan tekstur dari berbagai format termasuk file .obj dan file.3ds.

Library Min3D dan Android Studio perlu dihubungkan sehingga proses pengimportan objek menggunakan Min3D dapat dilakukan. Proses penghubungannya ini dilakukan dengan menambahkan kode program pada blok dependencies yang ada pada *build.gradle* Android Studio.

OpenGL (Open Graphics Library) pada penelitian ini digunakan untuk proses penggambaran atau rendering objek pada layar kamera. Ada beberapa tahapan dalam penggunaan OpenGL antara lain:

1. Proses penggambaran di sini dihubungkan dengan cara mengimplementasikan `GLSurfaceView` dan `GLSurfaceView.Renderer` pada Android Studio. `GLSurfaceView` di sini digunakan untuk penggambaran pada konten view, sedangkan `GLSurfaceView.Renderer` digunakan untuk mengontrol gambar pada konten view.

2. Proses selanjutnya yakni dengan mengkompres atau merubah image ke dalam format yang ada pada OpenGL. Method YUV image digunakan untuk mengkompres data yuv ke dalam image, dengan memanfaatkan integer rect(kiri, kanan, atas, bawah, tinggi dan lebar).
3. Tahap selanjutnya yakni dengan mengatur SurfaceView pada kamera menjadi translucent. Hal ini dilakukan untuk membuat kamera menjadi transparan sehingga objek dapat ditampilkan dalam satu view.



BAB III

DESAIN SISTEM

3.1 Perancangan Sistem

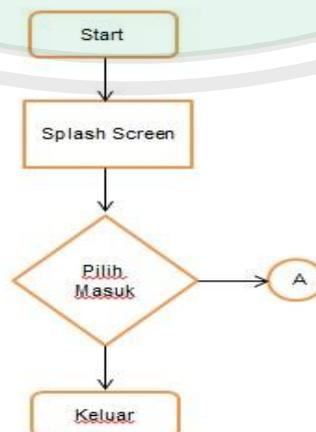
Penjelasan proses sistem penerapan transformasi geometri untuk visualisasi objek maya berbasis Android dapat digambarkan pada diagram blok seperti pada Gambar 3.1.



Gambar 3.1 Diagram Blok Aplikasi

3.3.1 Flowchart Sistem

Perancangan flowchart menggambarkan tahapan proses dari suatu sistem, termasuk sistem multimedia (Suyanto, 2003:64). Alur kerja aplikasi penelitian yang akan dibuat secara umum ditunjukkan dalam Gambar 3.2

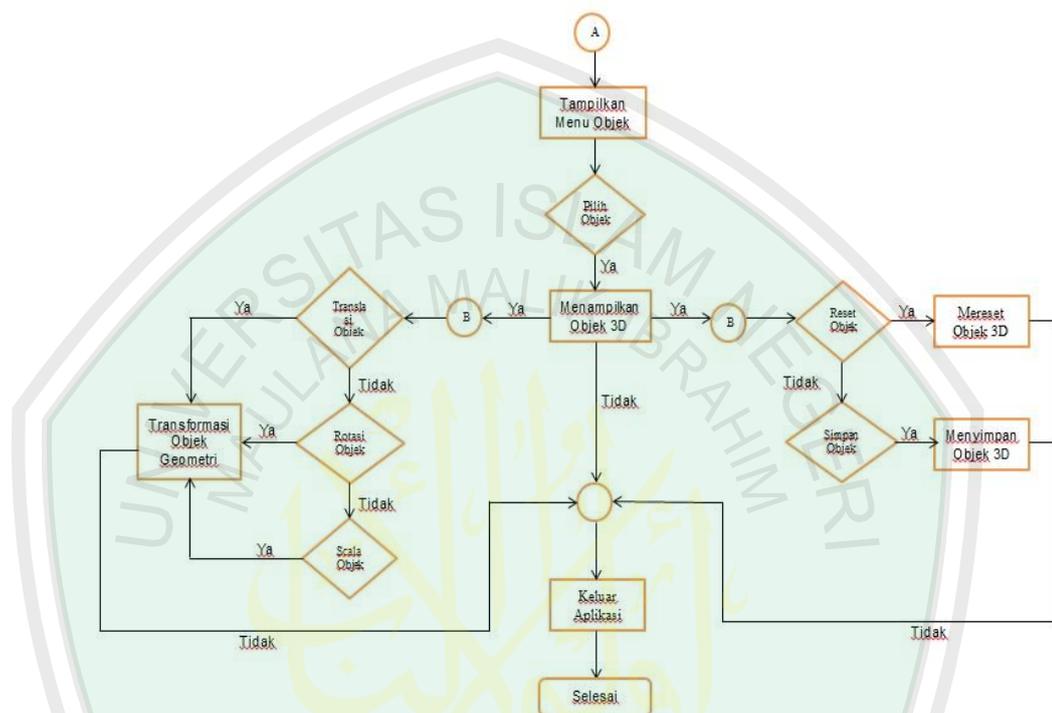


Keterangan

A : Halaman Utama

Gambar 3.2 Flowchart Splash Screen Aplikasi

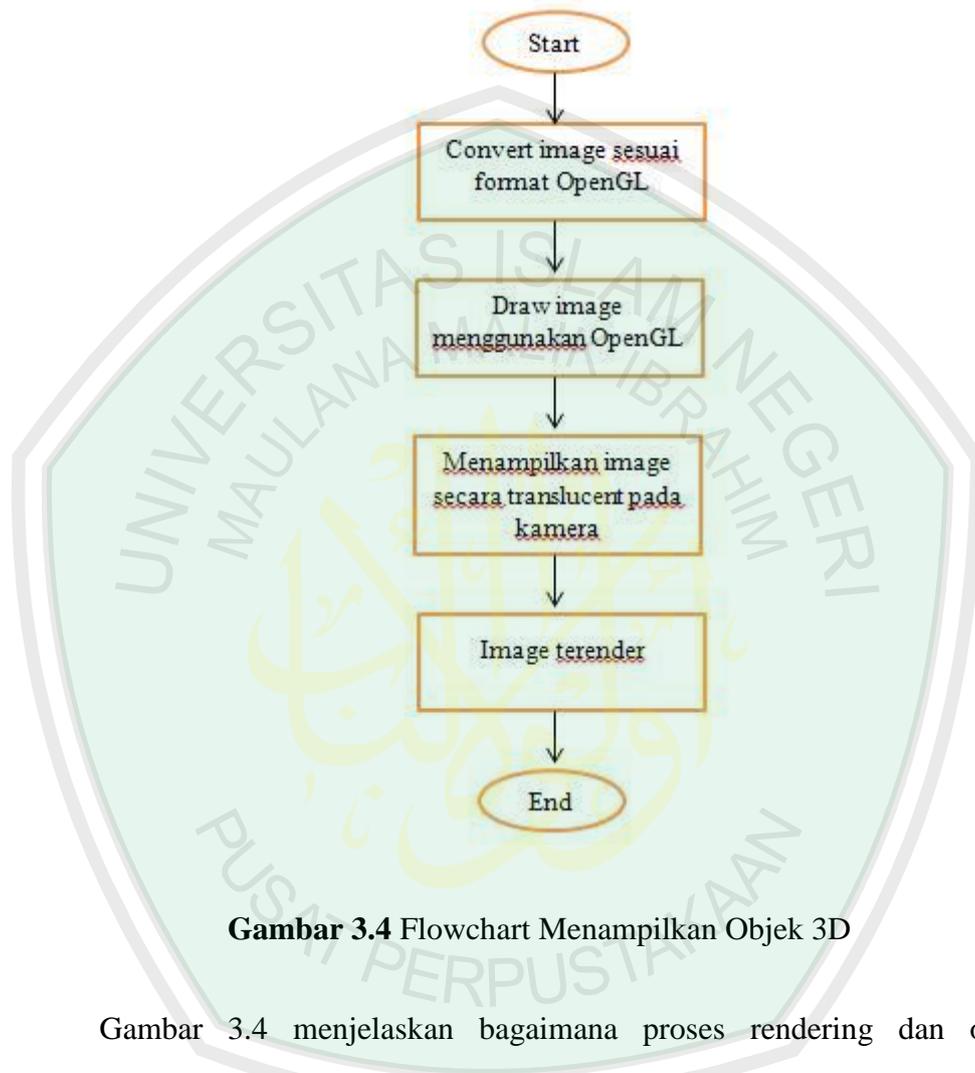
Gambar 3.2 menjelaskan tampilan pembuka dari aplikasi. Saat user memilih memilih masuk, maka terjadi 2 proses yakni masuk halaman utama atau ke proses selanjutnya.



Gambar 3.3 Flowchart Umum Sistem Aplikasi

Pada Gambar 3.3 setelah tampilan splash screen, pada layar aplikasi akan menampilkan pilihan menu objek maya berupa image 2D, yang mana image 2D ini digunakan sebagai pilihan gambar untuk menampilkan objek 3 dimensi yang diinginkan. Saat image 2D tadi dipilih, maka akan terjadi proses rendering untuk menampilkan objek yang dipilih. User juga dapat melakukan transformasi pada objek geometri tersebut yang meliputi proses transformasi translasi, rotasi, maupun penskalaan. Tahap akhir, user dapat menyimpan hasil model 3 dimensi tadi berupa image atau gambar yang langsung masuk pada internal memory Android.

Pada proses menampilkan objek seperti pada Gambar 3.3 terdapat proses rendering dan overlay objek maya pada kamera seperti pada Gambar 3.4



Gambar 3.4 Flowchart Menampilkan Objek 3D

Gambar 3.4 menjelaskan bagaimana proses rendering dan overlay dilakukan sehingga menghasilkan satu SurfaceView pada kamera. Proses awal rendering yakni image atau objek yang sebelumnya ada pada resource dipanggil menggunakan Min3D lalu mengkonvert image menggunakan format OpenGL. Android biasanya menggunakan YUV image dalam mengkonvert ke dalam bentuk gambar. Selanjutnya terjadi proses draw image yang telah di terkonvert pada layar kamera, lalu image yang tertampil pada layar kamera dirubah ke

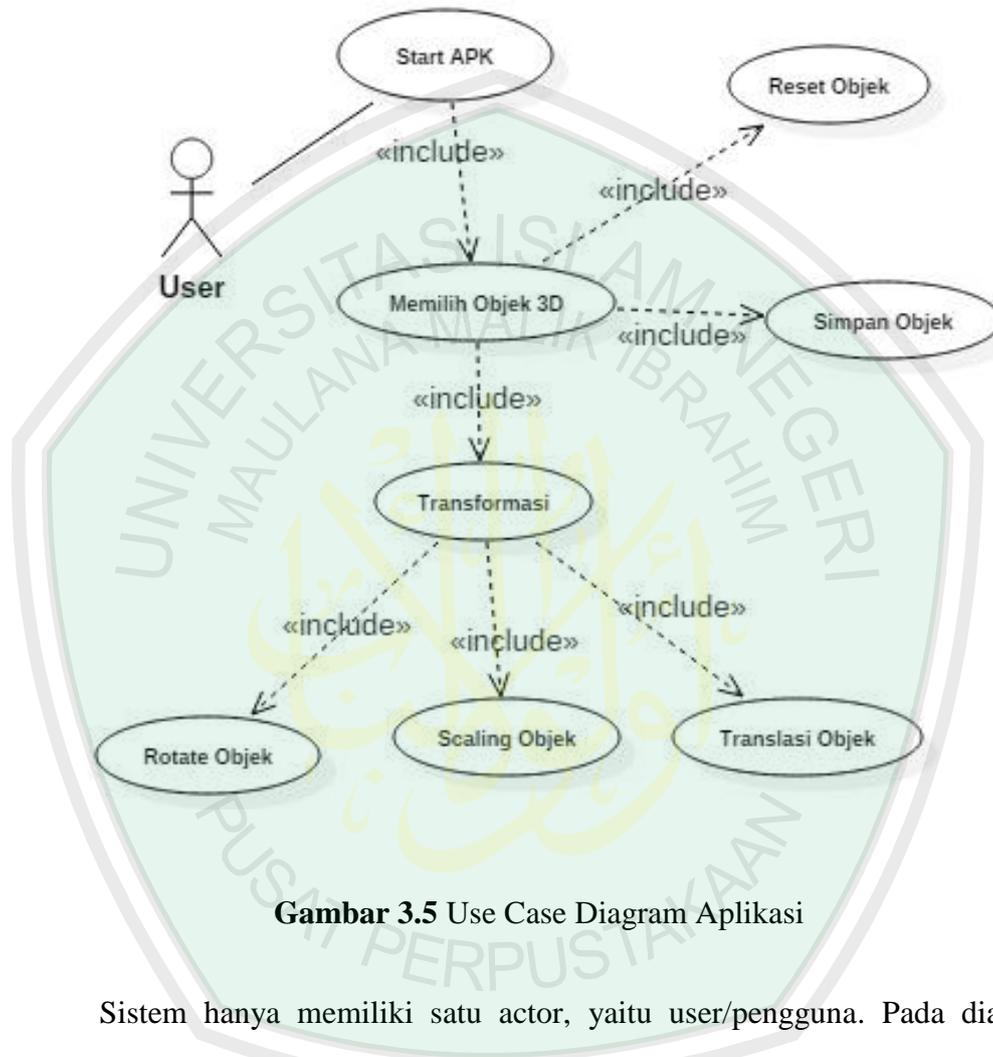
dalam translucent. Translucent di sini adalah proses menyamarkan SurfaceView pada kamera dan image menjadi satu sehingga SurfaceView yang ada tampak terlihat hanya satu view (overlay) dan hasilnya image berhasil terrender pada kamera.

3.3.2 Use Case Diagram

Use case merupakan rangkaian scenario yang mengidentifikasi urutan pemakaian aplikasi yang dibangun. Tujuan pembuatan use case di sini untuk mendefinisikan kebutuhan fungsional dan operasional sistem. Skenario penggunaan aplikasi desain interior rumah 3dimensi adalah sebagai berikut :

1. User start APK/Aplikasi. Pengguna menyalakan aplikasi yang sebelumnya telah di execute terlebih dahulu.
2. User memilih objek 3 dimensi. Pengguna memilih objek 3dimensi yang diinginkan sesuai dengan menu image 2dimensi yang ada di layar menu.
3. User melakukan transformasi terhadap objek geometri. Pengguna dapat melakukan proses transformasi pada objek geometri. Transformasi yang dapat dilakukan meliputi translasi, penskalaan, dan rotasi.
4. User dapat mereset ulang objek. Pengguna dapat mereset posisi, ukuran maupun sudut rotasi objek agar kembali ke posisi awal dimana objek pertama kali di inputkan.
5. User dapat menyimpan objek. Pengguna dapat menyimpan hasil program 3D tadi, yang nantinya outputnya berupa image/gambar.

Skenario di atas kemudian diubah ke dalam bentuk diagram use case seperti pada Gambar 3.5



Gambar 3.5 Use Case Diagram Aplikasi

Sistem hanya memiliki satu actor, yaitu user/pengguna. Pada diagram terlihat bahwa pengguna dapat melakukan 3 macam transformasi, yaitu translasi, rotasi, dan penskalaan. User juga dapat melakukan reset objek dari awal dan juga dapat menyimpan objek 3dimensi tadi berupa image atau gambar.

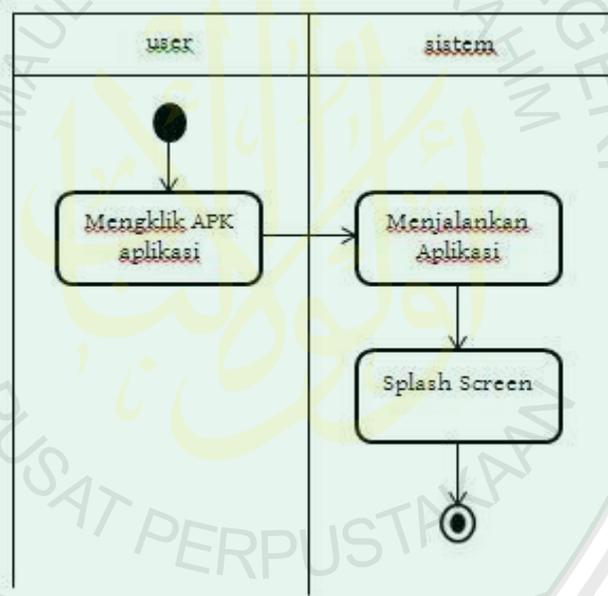
3.3.3 Activity Diagram

Activity diagram menggambarkan berbagai aliran aktivitas dalam sistem yang dirancang, bagaimana masing-masing fungsionalitas bekerja, dan bagaimana

suatu fungsionalitas berakhir. Activity diagram memodelkan event-event yang terjadi pada use case. Activity diagram dari aplikasi yang akan dibangun adalah sebagai berikut:

1. Activity Diagram Start APK

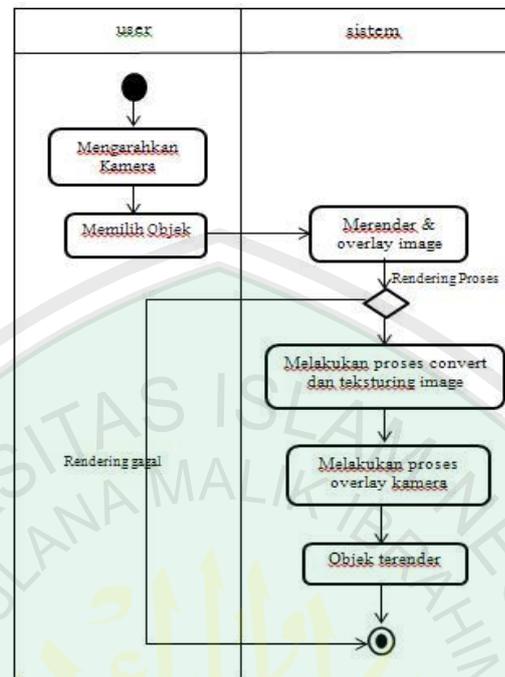
Merupakan aktifitas awal memulai menjalankan aplikasi, terdapat aktifitas setelah apk atau aplikasi dijalankan yakni proses splash screen yang berjalan beberapa menit sebelum memasuki halaman utama seperti pada Gambar 3.6



Gambar 3.6 Activity diagram Splash Screen

2. Activity Diagram Memilih Objek

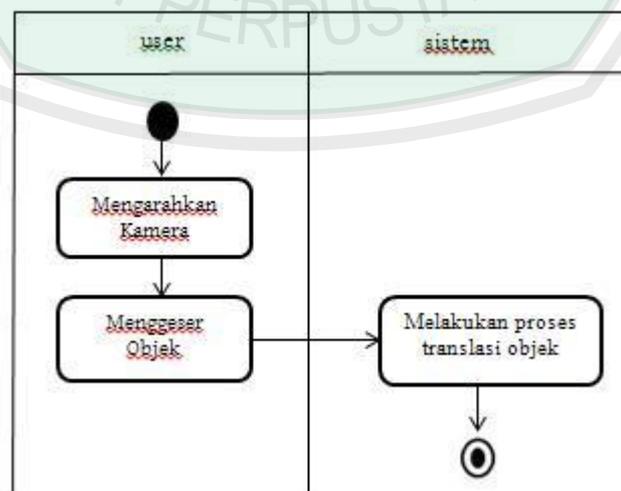
Merupakan aktifitas dalam memilih menu objek maya yang diinginkan, yang mana prosesnya dengan mengarahkan kamera ke latar yang diinginkan selanjutnya user dapat memilih objek berupa image 2D pada layar menu seperti pada Gambar 3.7



Gambar 3.7 Activity Diagram Memilih Objek

3. Activity Diagram Translasi Objek

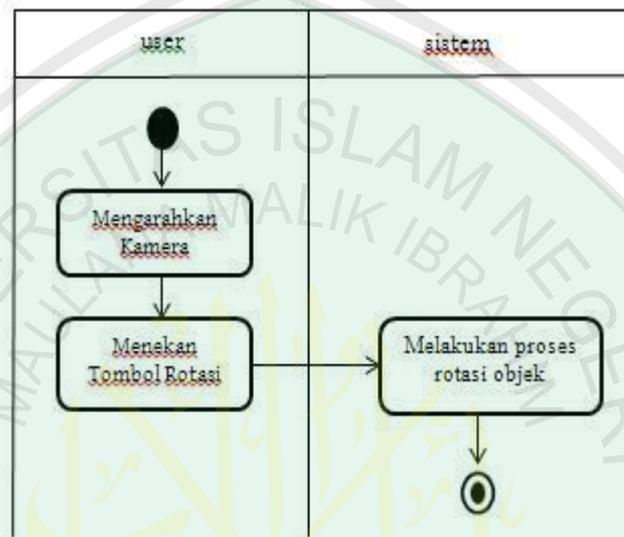
Merupakan aktifitas dalam proses transformasi translasi pada geometri, dengan melakukan pergeseran secara langsung (touch screen) pada objek seperti pada Gambar 3.8.



Gambar 3.8 Activity Diagram Translasi Objek Geometri

4. Activity Diagram Rotasi Objek

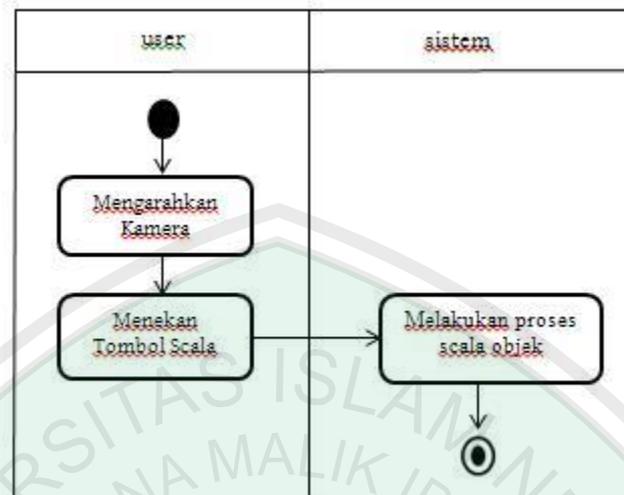
Merupakan aktifitas dalam proses transformasi translasi pada geometri, dengan menekan tombol rotasi, maka objek akan mengalami rotasi seperti pada Gambar 3.9.



Gambar 3.9 Activity Diagram Rotasi Objek Geometri

5. Activity Diagram Scala Objek

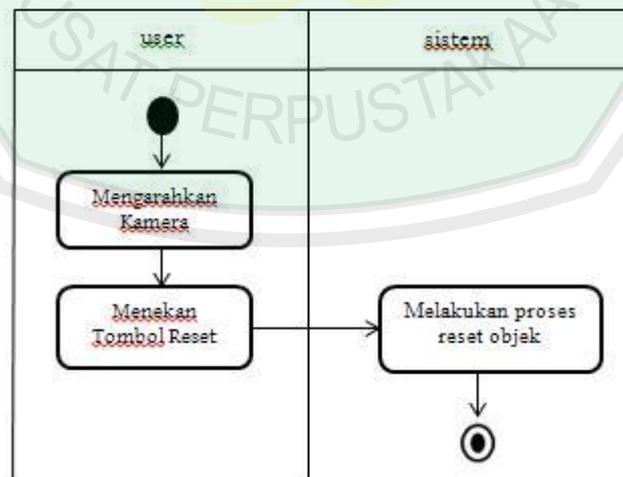
Merupakan aktifitas dalam proses transformasi translasi pada geometri, dengan menekan tombol scala plus, maka objek akan mengalami transformasi perbesaran skala. Sedangkan tombol scala minus menjadikan objek mengalami transformasi pengecilan skala seperti pada Gambar 3.10.



Gambar 3.10 Activity Diagram Scala Objek Geometri

6. Activity Diagram Reset Objek

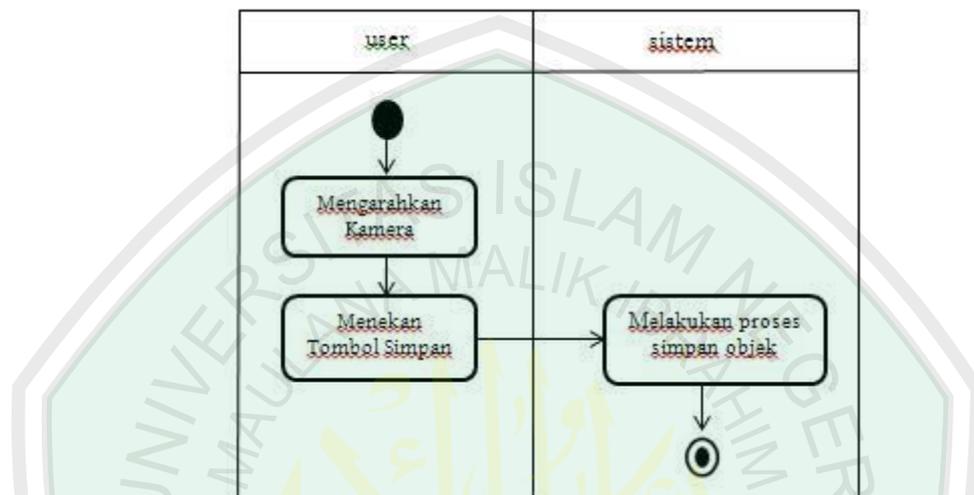
Merupakan aktifitas dalam proses reset pada objek dengan menekan tombol reset, maka objek akan mengalami reset atau kembali ke kondisi awal secara default seperti pada Gambar 3.11



Gambar 3.11 Activity Diagram Reset Objek Geometri

7. Activity Diagram Simpan Objek

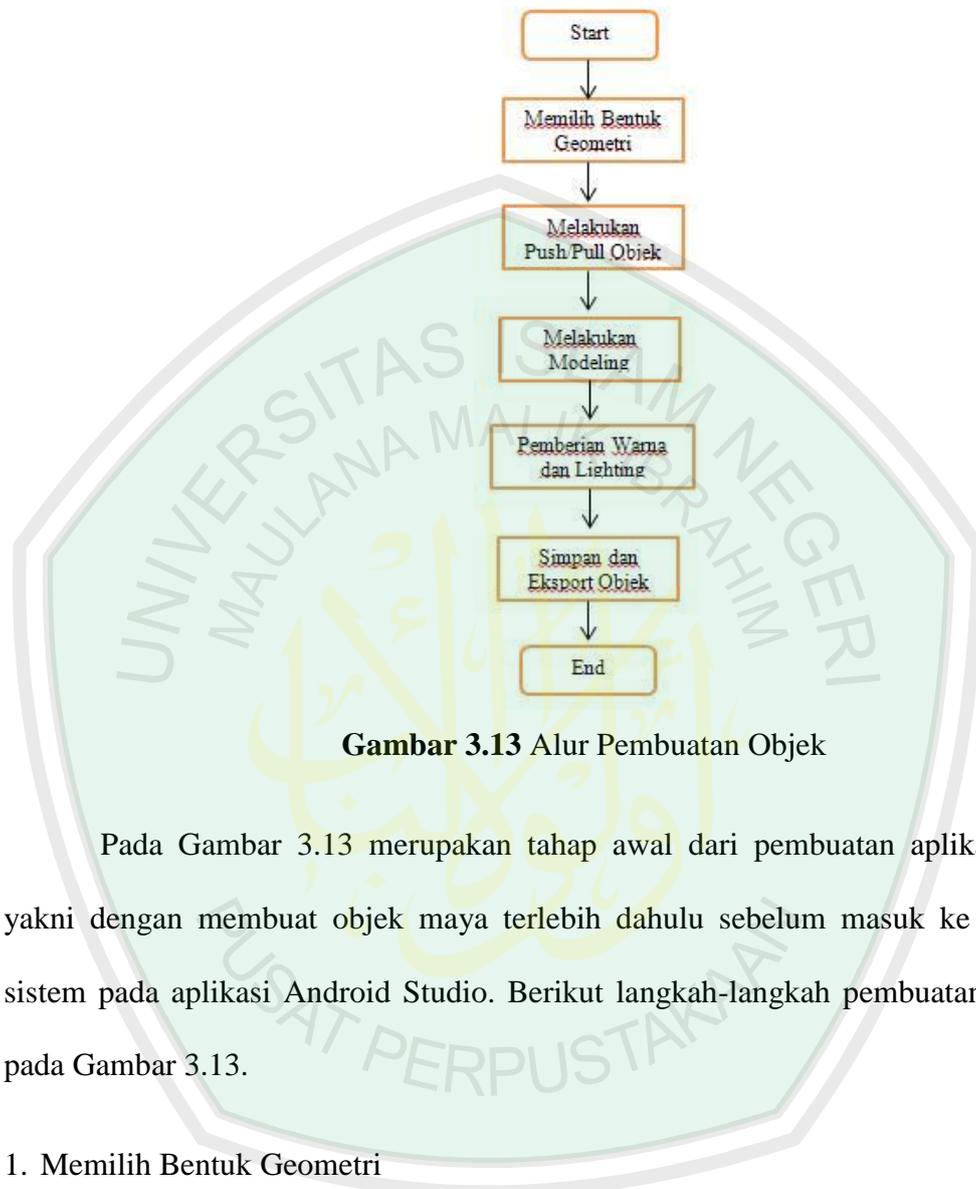
Merupakan aktifitas dalam proses menyimpan objek dengan menekan tombol simpan, maka objek akan tersimpan pada internal memory user seperti pada Gambar 3.12



Gambar 3.12 Activity Diagram Simpan Objek Geometri

3.4 Perancangan Pembuatan Objek Maya

Perancangan pembuatan objek maya 3 dimensi pada aplikasi ini dibuat menggunakan Google SketchUp. Objek yang ada pada penelitian berjumlah 5 buah meliputi kursi, lemari, kulkas, kasur, dan tv. Alur pembuatan objek maya ini secara umum dapat dilihat pada Gambar 3.13.

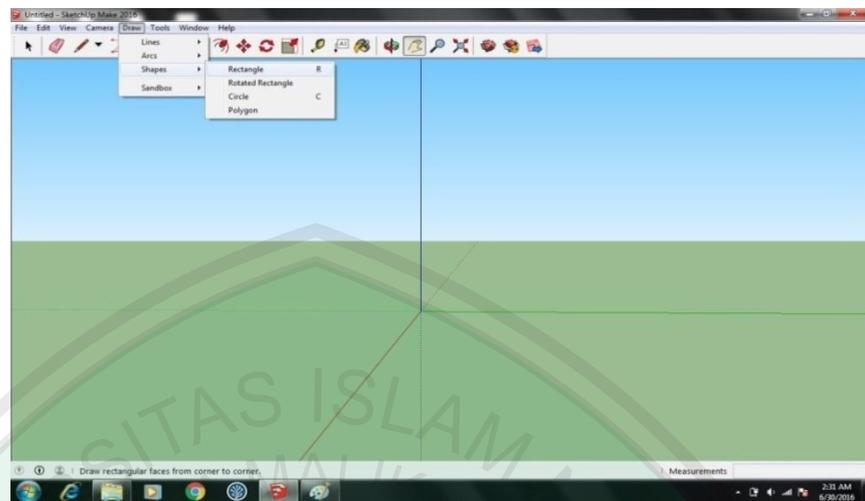


Gambar 3.13 Alur Pembuatan Objek

Pada Gambar 3.13 merupakan tahap awal dari pembuatan aplikasi ini, yakni dengan membuat objek maya terlebih dahulu sebelum masuk ke proses sistem pada aplikasi Android Studio. Berikut langkah-langkah pembuatan objek pada Gambar 3.13.

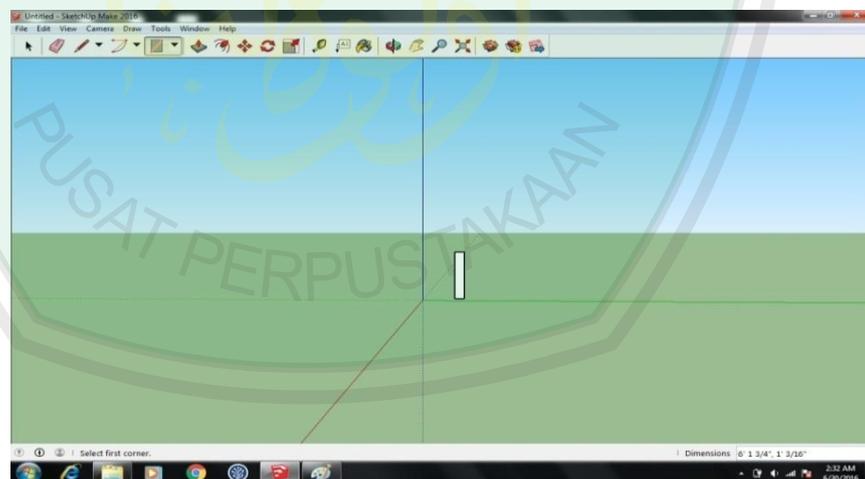
1. Memilih Bentuk Geometri

Pada tahap ini menu draw pada GoogleSketchUp digunakan untuk pemilihan bentuk maupun garis geometri yang diinginkan baik itu rectangle, circle, polygon dsb seperti pada Gambar 3.14



Gambar 3.14 Menu Draw Pada Goggle SketchUp

Pada Gambar 3.15 menjelaskan bagaimana hasil peletakan objek geometri pada layar. Pada tahap ini kita bisa meletakkan bentuk geometri sesuai sudut yang diinginkan

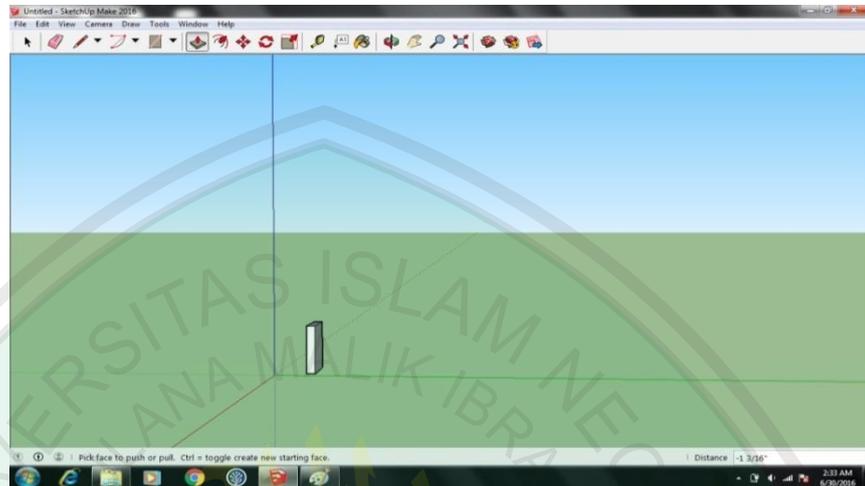


Gambar 3.15 Peletakan Objek Geometri

2. Melakukan Push/Pull Objek

Pada tahap ini tombol push/pull digunakan untuk menarik/melebarkan objek geometri sesuai keinginan, hal ini dilakukan

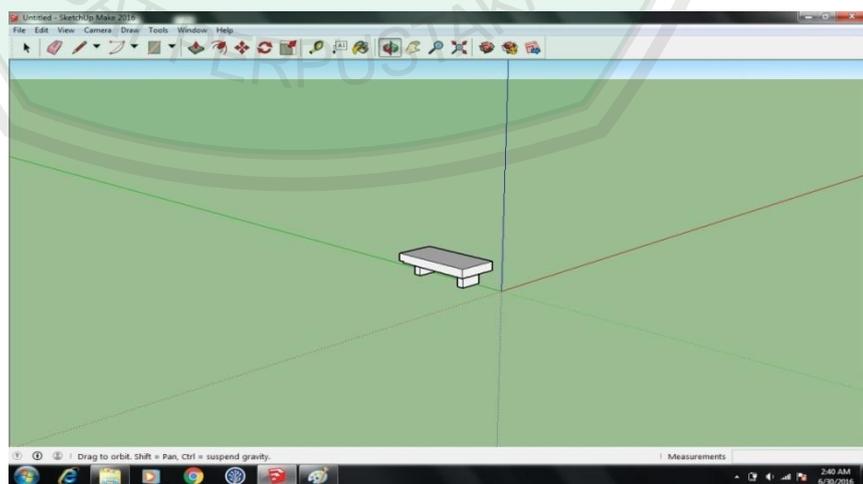
untuk memberikan ruang atau dimensi pada geometri seperti pada Gambar 3.16



Gambar 3.16 Push dan Pull Objek Geometri

3. Melakukan Modeling

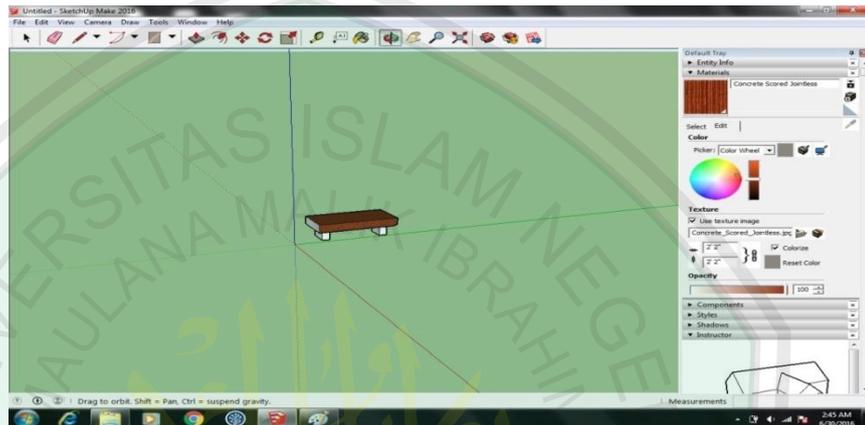
Merupakan proses lanjutan dari push/pull objek, di mana tahap ini merupakan tahap proses pemodelan seperti pada penjelasan sub bab 2.3. Pada tahap ini objek geometri dapat dipahat sedemikian rupa sehingga membentuk objek 3 dimensi yang diinginkan seperti pada Gambar 3.17.



Gambar 3.17 Pemodelan Objek Geometri

4. Pemberian Warna dan Lighting

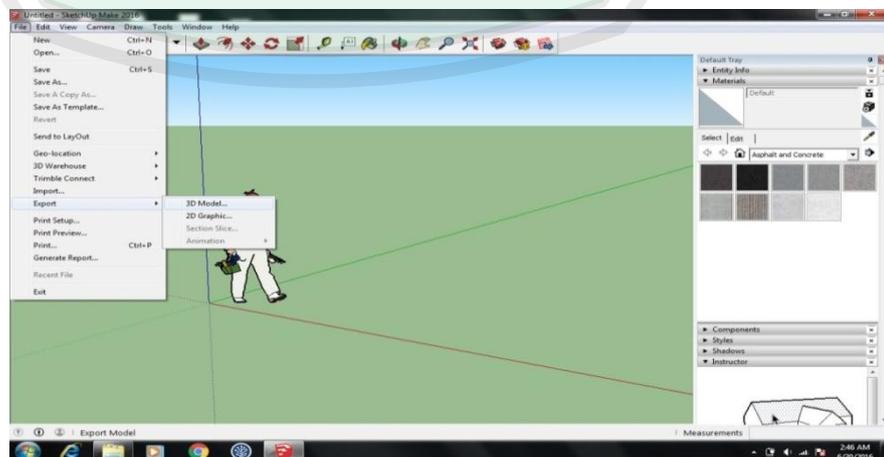
Pada tahap ini pewarnaan dan lighting dilakukan. Pemberian warna dan lighting pada objek digunakan untuk memberikan kesan agar objek tampak real dan halus seperti pada Gambar 3.18



Gambar 3.18 Pemberian Warna dan Lighting

5. Menyimpan dan Ekspor Objek

Pada tahap ini objek geometri yang telah dibuat tadi disimpan, lalu di ekspor menjadi format (.obj). Format .obj dipilih karena sifatnya yang universal dan memudahkan dalam proses rendering nantinya seperti pada Gambar 3.19



Gambar 3.19 Simpan dan Ekspor Objek 3 Dimensi

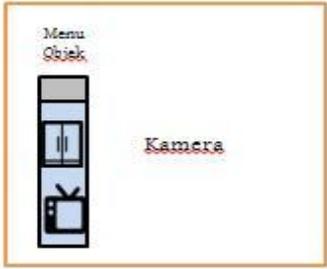
3.5 Desain User Interface

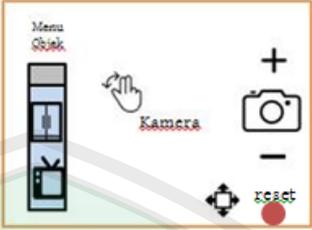
Desain user interface merupakan rancangan tampilan dari aplikasi yang akan dibuat. Berikut adalah perancangan antarmuka dari aplikasi penelitian mulai dari splash screen, halaman menu, dan halaman utama.

3.5.1 Storyboard Aplikasi

Storyboard adalah visualisasi ide dari aplikasi yang akan dibangun, sehingga dapat memberikan gambaran dari aplikasi yang akan dibuat. Berikut merupakan story board dari aplikasi visualisasi objek maya menggunakan transformasi geometri berbasis android dpaat dilihat pada Table 3.1.

Tabel 3.1 Storyboard Aplikasi Visualisasi Objek Maya Berbasis Android

Scene	Nama	Gambar	Keterangan
1	Splash Screen		Splash screen berisi logo aplikasi yang muncul beberapa detik sebelum masuk ke halaman utama aplikasi.
2	Halaman Menu Objek		Berisi menu objek maya 2D yang digunakan sebagai pemilihan objek maya pada

			aplikasi.
3.	Halaman Utama		Berisi gambar menu objek dan tombol-tombol untuk melakukan proses pada aplikasi.

3.5.2 Desain Interface Splash Screen

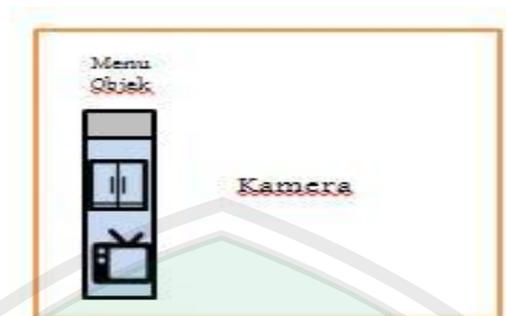
Splash Screen merupakan halaman muka sebelum masuk ke halaman utama aplikasi. Splash Screen di sini hanya beberapa detik saja sebelum masuk ke halaman utama, berisi logo dan nama aplikasi yang dibuat seperti pada Gambar 3.20



Gambar 3.20 Desain Interface Splash Screen Aplikasi

3.5.3 Desain Interface Halaman Menu Objek

Halaman menu objek merupakan menu untuk memilih objek maya yang diinginkan. Berupa image 2 dimensi pada layar menu, yang kemudian saat dipilih image tadi, maka akan muncul objek 3 dimensi sesuai image yang ada seperti pada Gambar 3.21.



Gambar 3.21 Desain Interface Menu Utama Aplikasi

3.5.4 Desain Interface Halaman Utama Aplikasi

Halaman utama merupakan halaman utama dalam aplikasi, berisi menu objek dan tombol-tombol dalam pembuatannya seperti pada Gambar 3.22



Gambar 3.22 Desain Interface Halaman Utama Aplikasi

Penjelasan tombol-tombol pada halaman utama aplikasi antara lain:

- Tombol Menu Objek merupakan menu untuk menampilkan pilihan objek pada kamera, yang mana objek maya telah dibuat sebelumnya menggunakan Google SketchUp.
- Tombol Touch Objek merupakan layar sentuh pada objek maya, yang nantinya digunakan untuk proses transformasi translasi.

- Tombol Plus merupakan tombol untuk melakukan transformasi skala perbesaran pada objek maya.
- Tombol Minus merupakan tombol untuk melakukan transformasi skala pengecilan pada objek maya.
- Tombol Reset merupakan tombol untuk mengatur ulang atau mengembalikan objek maya kembali ke awal secara default.
- Tombol Simpan merupakan tombol untuk menyimpan hasil aplikasi dalam bentuk image atau gambar.



BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

Pada bab ini membahas mengenai implementasi dan pengujian pada aplikasi. Implementasi adalah bagaimana mewujudkan hasil dari perancangan sehingga menghasilkan suatu aplikasi yang dapat bekerja dengan baik. Setelah itu, pengujian dilakukan untuk mengetahui apakah aplikasi sudah dapat bekerja sebagaimana mestinya atau belum. Jika hasilnya belum maka aplikasi masih perlu disempurnakan.

4.1 Implementasi Sistem

Tahap implementasi pengembangan perangkat lunak merupakan proses perubahan spesifikasi sistem menjadi sistem yang dapat dijalankan. Tahap ini merupakan lanjutan dari proses perancangan, yaitu proses pemrograman perangkat lunak sesuai dengan spesifikasi dan desain sistem. Tujuan dari implementasi sistem adalah untuk menerapkan perancangan yang telah dilakukan terhadap sistem, sehingga user dapat memberikan masukan untuk dilakukan perbaikan terhadap sistem agar sistem menjadi lebih baik.

Aplikasi penerapan transformasi geometri untuk visualisasi objek maya berbasis Android ini menggunakan beberapa spesifikasi perangkat lunak maupun perangkat keras dalam pembuatannya. Berikut adalah spesifikasi perangkat yang digunakan dalam pembuatan aplikasi:

Tabel 4.1 Spesifikasi Perangkat

Hardware	Software
Intel(R) Core(TM)2Duo CPU T6400 @ 2.00GHz (2 CPUs), ~2.0GHz	Sistem operasi Windows 7 Home Premium 32 bit
RAM 1 GB	Android Studio
HDD 500 GB	Google SketchUp
	JDK 8
	Adobe Photoshop CS5

4.1.1 Implementasi Interface

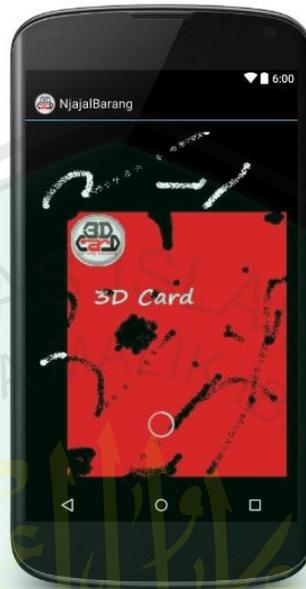
Implementasi Interface merupakan implementasi dari perancangan tampilan dari program yang bisa dinikmati oleh *user*. Implementasi *User Interface* harus dibuat sesuai dengan perancangan dengan tetap mengutamakan kenyamanan dalam mengoperasikan program (*user friendly*).

Berikut merupakan rancangan tampilan desain interface dari aplikasi penerapan transformasi geometri untuk visualisasi objek maya berbasis Android.

a. Implementasi Tampilan Splash Screen

Implementasi tampilan *Splash Screen* atau tampilan pembuka saat menjalankan aplikasi visualisasi objek maya. Tampilan pembuka di sini menggunakan *Splash Screen* dalam penerapannya, yang mana memiliki interval

waktu beberapa detik saja sebelum memasuki halaman utama seperti pada Gambar 4.1



Gambar 4.1 Tampilan Layout Splash Screen Aplikasi

b. Implementasi Tampilan Halaman Menu Objek

Implementasi halaman menu objek merupakan tampilan di mana pemilihan objek dilakukan dan tiap tombol menu objek diklik maka hanya akan menampilkan 4 macam objek sedangkan untuk mendapatkan objek yang lain dapat dilakukan scroll pada menu objek seperti pada Gambar 4.2.

c. Implementasi Tampilan Halaman Utama Aplikasi

Implementasi tampilan halaman utama aplikasi merupakan halaman keseluruhan aplikasi baik itu tampilan menu maupun tombol-tombol untuk melakukan pemrosesan seperti yang telah dijelaskan pada bab sebelumnya.

Implementasi tampilan halaman utama dari aplikasi ini dapat dilihat pada Gambar

4.2



Gambar 4.2 Tampilan Layout Menu Objek dan Halaman Utama Aplikasi

4.1.2 Implementasi Kode Program Rancangan Kerja Sistem

Implementasi Kode Rancangan Kerja Sistem adalah Implementasi dari model representasi aliran proses perangkat lunak yang telah dirancang akan disajikan dalam *Flowchart Diagram* yang disajikan dengan kode program.

Implementasi dari *Flowchart Diagram* digunakan untuk menjelaskan cara kerja semua sistem yang ada pada aplikasi penerapan transformasi geometri untuk visualisasi objek maya berbasis Android baru secara kode program. Berikut adalah implementasi dari Rancangan Kerja Sistem dari aplikasi penerapan transformasi geometri untuk visualisasi objek maya berbasis Android.

a. Implementasi Splash Screen

Implementasi kode program dari pembuatan splash screen ini menjelaskan proses saat splash screen berjalan dengan memberikan nilai interval untuk mengatur waktu lama Splash Screen kemudian langsung otomatis masuk ke halaman utama aplikasi. Pengaturan waktu lama interval dapat dirubah-rubah sesuai keinginan dengan mengganti nilai interval pada kode program.

Berikut kode program dari splash screen aplikasi penerapan transformasi geometri untuk visualisasi objek maya pada Android dapat dilihat pada Source Code 4.1 dan 4.2.

```

</activity>
    <activity android:name=".SplashScreen"
        android:noHistory="true"
        android:screenOrientation="portrait">
        <intent-filter>
            <action android:name="android.intent.action.MAIN"
/>
                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

```

Source Code 4.1 Kode Program Splash Screen pada AndroidManifest

```

package com.skripsi.r3dcard;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;

public class SplashScreen extends Activity {

    //set waktu lama splash
    private static int splashInterval = 2000;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.splash_screen);

        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                //TODO method auto
                Intent i = new Intent(SplashScreen.this,
MainRenderActivity.class);

```

```

        startActivity(i);
        this.finish();
    }

    private void finish() {

    }

    }, splashInterval);
};
}

```

Source Code 4.2 Kode Program Splash Screen pada MainActivity

b. Implementasi Kode Program Menu Objek

Implementasi kode program menu Objek menjelaskan proses pembuatan menu objek hingga listing menu-menu objek apa saja yang akan ditampilkan pada aplikasi, yang mana jumlah objek yang dapat di tampilkan terbatas dari peneliti. Implementasi kode program dari menu objek dapat ditunjukkan pada source berikut.

```

public void inisialisasiMenuNya() {
    MenuObjek3d = (LinearLayout) findViewById(R.id.menu3d);
    ImageButton btnMenu[] = new
ImageButton[nama2object.size()];
    for (int i = 0; i < btnMenu.length; i++) {
        final int indexBT = i;
        final String nama = nama2object.get(i);
        btnMenu[i] = new ImageButton(context);
        btnMenu[i].setId(i);
        btnMenu[i].setOnClickListener(new OnClickListener() {
            @Override
            public void onClick(View v) {
                addObject3d(indexBT);
            }
        });

        btnMenu[i].setLayoutParams(new LayoutParams(
            LayoutParams.WRAP_CONTENT,
            LayoutParams.WRAP_CONTENT));
        String namaObjek = nama.split("_")[0];
        int id = context.getResources().getIdentifier(
            "icon_object_" + namaObjek, "drawable",
            context.getPackageName());
        btnMenu[i].setImageResource(id);

        btnMenu[i].setBackgroundResource(android.R.color.transparent);
        MenuObjek3d.addView(btnMenu[i]);
    }
}

```

Source Code 4.3 Kode Program Inisialisasi Menu Objek

Source Code 4.3 merupakan tahap awal penginisialisasian dalam pembuatan menu yang akan ditampilkan.

```

public void addObject3d(int index) {
    scene.reset();
    indexObjectMuncul = index;
    String namaObject = nama2object.get(index);
    String[] splitNama = namaObject.split("_");
    String TipeData = splitNama[splitNama.length - 1];
    if (TipeData.equals("obj")) {
        IParser parser = Parser.createParser(Parser.Type.OBJ,
            getResources(),
            "com.skripsi.r3dcard:raw/" +
nama2object.get(index), true);
        parser.parse();
        objModel = parser.getParsedObject();
    }
}

```

Source Code 4.4 Kode Program Penambahan Objek

Source Code 4.4 merupakan tahapan pemanggilan objek objek yang sebelumnya telah tersimpan pada resource.

Langkah selanjutnya yakni menset objek maya sesuai default awal yang telah ditentukan seperti pada Source Code 4.5.

```

objModel.scale().x = 0.07f;
objModel.scale().y = 0.07f;
objModel.scale().z = 0.07f;

objModel.position().x = 0.0f;
objModel.position().y = 0.0f;
objModel.position().z = 0.0f;

objModel.rotation().x = 45;
objModel.rotation().y = -45;

objModel.colorMaterialEnabled(true);
scene.backgroundColor().setAll(0x00000000);
scene.lightingEnabled(false);
scene.addChild(objModel);
}

```

Source Code 4.5 Kode Program Set Default Objek

Selanjutnya menyusun list objek apa saja yang akan ditampilkan pada kamera seperti pada Source Code 4.6

```

public void tampilkanObject3d() {
    // nama2object = new ArrayList<String>();
    // nama2object.add("camaro_obj");
    // nama2object.add("monster_obj");
    // nama2object.add("komputer_obj");
    // nama2object.add("monster_3ds");
    // nama2object.add("komputer_obj");
    // nama2object.add("kursi_obj");
    nama2object.add("almari_obj");
    nama2object.add("kursi_obj");
    nama2object.add("kulkas_obj");
    nama2object.add("tv_obj");
    nama2object.add("tempattidur_obj");
    nama2object.add("almari_obj");
    nama2object.add("kulkas_obj");
    nama2object.add("tv_obj");
}

```

Source Code 4.6 Kode Program List Nama Objek

```

    ScrollViewObject = (ScrollView)
    findViewById(R.id.ScroolViewObject2d);

    btnShowHideObject = (ImageButton)
    findViewById(R.id.btnShowObject);

    btnShowHideObject.setImageResource(R.drawable.icon_menu_object);
    btnShowHideObject.setOnClickListener(new
    OnClickListener() {
        @Override
        public void onClick(View v) {
            if (ScrollViewObject.getVisibility() ==
    View.VISIBLE) {
                ScrollViewObject.setVisibility(View.INVISIBLE);
                btnShowHideObject
                .setImageResource(R.drawable.icon_menu_object);
            } else {
                ScrollViewObject.setVisibility(View.VISIBLE);
                btnShowHideObject
                .setImageResource(R.drawable.icon_menu_object_press);
            }
        }
    });

```

Source Code 4.7 Kode Program Tombol View Objek

Source Code 4.7 merupakan tahap aksi pada tombol menu objek untuk menampilkan dan menyembunyikan objek.

c. Implementasi Kode Program Visualisasi Objek Pada Kamera

Implementasi kode program visualisasi objek pada kamera menjelaskan proses rendering objek dan menampilkan objek pada kamera dengan cara menampilkan kamera live dibelakang objek 3 dimensi (overlay). Implementasi kode program dari rendering objek pada kamera dapat ditunjukkan pada source code berikut.

```
//Ukuran pas kamera
private Size getOptimalPreviewCameraSize(List<Size> sizes,
int w, int h) {
    final double ASPECT_TOLERANCE = 0.05;
    double targetRatio = (double) w / h;
    if (sizes == null)
        return null;

    Size optimalSize = null;
    double minDiff = Double.MAX_VALUE;

    int targetHeight = h;

    // Try to find a size match aspect ratio and size
    for (Size size : sizes) {
        double ratio = (double) size.width / size.height;
        if (Math.abs(ratio - targetRatio) > ASPECT_TOLERANCE)
            continue;
        if (Math.abs(size.height - targetHeight) < minDiff) {
            optimalSize = size;
            minDiff = Math.abs(size.height - targetHeight);
        }
    }

    // Cannot find the one match the aspect ratio, ignore the
    requirement
    if (optimalSize == null) {
        minDiff = Double.MAX_VALUE;
        for (Size size : sizes) {
            if (Math.abs(size.height - targetHeight) <
minDiff) {
                optimalSize = size;
                minDiff = Math.abs(size.height -
targetHeight);
            }
        }
    }
    return optimalSize;
}
}
```

Source Code 4.8 Kode Program Pembuatan Ukuran Kamera

Source Code 4.8 menjelaskan tahap pembuatan ukuran pas kamera pada semua smartphone selama tidak melebihi batas ukuran yang ditentukan.

```

//Parameter awal camera
@Override
public void surfaceChanged(SurfaceHolder holder, int format,
int width,
int height) {
// Now that the size is known, set up the camera
parameters and begin
// the preview.
Camera.Parameters parameters = camera.getParameters();

List<Size> sizes = parameters.getSupportedPreviewSizes();
Size optimalSize = getOptimalPreviewCameraSize(sizes,
width, height);
parameters.setPreviewSize(optimalSize.width,
optimalSize.height);
camera.setParameters(parameters);
// camera.startPreview();
if (previewing) {
camera.stopPreview();
previewing = false;
}

//Parameter lanjutan camera
@Override
public void surfaceDestroyed(SurfaceHolder holder) {
camera.stopPreview();
camera.setPreviewCallback(null);
camera.release();
camera = null;
previewing = false;
}

//setting kamera
public void surfaceCreated(SurfaceHolder holder) {
try {
camera = Camera.open();
if (camera != null) {
Parameters p = camera.getParameters();
p.setFlashMode(Parameters.FLASH_MODE_OFF);
camera.setParameters(p);
}
} catch (Exception e) {
e.printStackTrace();
}
}
}

```

Source Code 4.9 Kode Program Parameter dan Pengaturan Kamera

Source Code 4.9 merupakan tahap dalam pembentukan parameter-parameter pada kamera live dan pengaturannya.

Langkah selanjutnya yaitu menampilkan view kamera secara live. Kode program dari proses ini dapat dilihat pada Source Code 4.10.

```

//Tampilkan view pada kamera

```

```

public void tampilkanViewKamera() {
    // tampilkan view kamera-----
-----
    getWindow().setFormat(PixelFormat.UNKNOWN);
    surfaceView = (SurfaceView)
    findViewById(R.id.surfaceViewRender);
    surfaceHolder = surfaceView.getHolder();
    surfaceHolder.addCallback(this);

    surfaceHolder.setType(SurfaceHolder.SURFACE_TYPE_PUSH_BUFFERS);
    // -----
}

```

Source Code 4.10 Kode Program Menampilkan View Kamera

Tahap selanjutnya adalah mengkompres atau merubah image menjadi format pada OpenGL. Kode program dari proses ini dapat dilihat pada Source Code 4.11

```

//Kompres yuvimage dan bitmap factory
if (camera != null) {
    try {
        camera.setPreviewDisplay(surfaceHolder);
        camera.setPreviewCallback(new PreviewCallback() {
            @Override
            public void onPreviewFrame(byte[] data,
Camera c) {
                if (savePreview) {
                    Camera.Parameters params =
c.getParameters();
                    int w =
params.getPreviewSize().width;
                    int h =
params.getPreviewSize().height;
                    int format =
params.getPreviewFormat();
                    YuvImage image = new YuvImage(data,
format, w, h,
                    null);

                    ByteArrayOutputStream out = new
ByteArrayOutputStream();
                    Rect area = new Rect(0, 0, w, h);
                    image.compressToJpeg(area, 50, out);
                    Bitmap bm =
BitmapFactory.decodeByteArray(
                    out.toByteArray(), 0,
out.size());
                    // ivCam.setImageBitmap(bm);
                    r.saveScreen(bm);
                    savePreview = false;
                }
            }
        });
        camera.startPreview();
    }
}

```

```

        previewing = true;
    } catch (IOException e) {
        e.printStackTrace();
    }
}
}

```

Source Code 4.11 Kode Program Merubah Format Image

Selanjutnya adalah melakukan rendering pada image dan meletakkannya secara transparent pada view kamera, sehingga kamera live dan objek 3 dimensi saling menumpuk namun terkesan hanya ada satu view yang ditampilkan. Kode program dari proses ini dapat dilihat pada Source Code 4.12.

```

// tampilkan objek 3d-----
    _glSurfaceView = (GLSurfaceView)
findViewById(R.id.surfaceviewclass);
    glSurfaceViewConfig();
    _glSurfaceView.setRenderer(r);

    _glSurfaceView.setRenderMode(GLSurfaceView.RENDERMODE_CONTINUOUSL
Y);
    // -----

}

// Setting kamera agar tampak transparan
@Override
protected void glSurfaceViewConfig() {
    _glSurfaceView.setEGLConfigChooser(8, 8, 8, 8, 16, 0);
    _glSurfaceView.getHolder().setFormat(PixelFormat.TRANSLUCENT);
}

```

Source Code 4.12 Kode Program Rendering dan Overlay Objek

d. Implementasi Kode Program Transformasi Geometri

Implementasi kode program proses transformasi geometri untuk menampilkan objek maya pada aplikasi ini memiliki beberapa proses operasi pada proses transformasi geometri, yakni translasi, rotasi dan skala.

Tahap pertama yakni proses transformasi translasi dengan melakukan perpindahan benda dari satu titik ke titik yang lain. Perhitungan dalam melakukan pergeseran pada objek dapat dilihat pada source code berikut.

```
//Geser Object 3d
@Override
public boolean onTouchEvent(MotionEvent event) {
    if (true) {
        switch (event.getAction()) {
            // Action started
            default:
            case MotionEvent.ACTION_DOWN:
                lastX = event.getX();
                lastY = event.getY();
                if (!resetPertama) {
                    resetSek();
                    resetPertama = true;
                }
                if (textHint.isShown()) {
                    textHint.setVisibility(View.INVISIBLE);
                }
                break;

                // Action ongoing
            case MotionEvent.ACTION_MOVE:
                float dX = lastX - event.getX();
                float dY = lastY - event.getY();
                lastX = event.getX();
                lastY = event.getY();
                posisi_x -= (dX / 100f) / 2;
                posisi_y += (dY / 100f) / 2;

                objModel.position().x = posisi_x;
                objModel.position().y = posisi_y;

                break;

            // Action ended
            case MotionEvent.ACTION_CANCEL:
            case MotionEvent.ACTION_UP:
                lastX = event.getX();
                lastY = event.getY();
                break;
        }
    }
    return true;
}
```

Source Code 4.13 Kode Program Proses Awal Translasi

Source Code 4.13 menjelaskan tahapan awal pada proses translasi dengan mendapatkan nilai x dan y terlebih dahulu lalu mengaktifkan tipe datanya (boolean) menjadi true.

Selanjutnya melakukan perhitungan pada posisi x dan y berada, dengan cara mendapatkan nilai posisi awal lalu dikurangi dengan nilai posisi sekarang. Selanjutnya untuk mendapatkan posisi akhir dilakukan perhitungan dengan cara (nilai hasil / 100f) / 2 lalu menghentikan proses translasi seperti pada Source Code 4.14

```
// Action ongoing
case MotionEvent.ACTION_MOVE:
    float dX = lastX - event.getX();
    float dY = lastY - event.getY();
    lastX = event.getX();
    lastY = event.getY();
    posisi_x -= (dX / 100f) / 2;
    posisi_y += (dY / 100f) / 2;

    objModel.position().x = posisi_x;
    objModel.position().y = posisi_y;

    break;

// Action ended
case MotionEvent.ACTION_CANCEL:
case MotionEvent.ACTION_UP:
    lastX = event.getX();
    lastY = event.getY();
    break;
}
return true;
}
```

Source Code 4.14 Kode Program Aksi Translasi

Tahap kedua yakni proses transformasi rotasi dengan menekan tombol rotasi maka akan terjadi perputaran pada objek tanpa membuat objek berpindah. Proses rotasi dilakukan dengan cara memberikan nilai awal terlebih dahulu seberah jauh nilai rotasi yang dilakukan. Selanjutnya untuk proses perhitungan dilakukan dengan cara mengurangi nilai awal tadi dengan seberapa jauh nilai rotasi dilakukan seperti pada Source Code 4.15

```
//Rotasi
seekRotate = (SeekBar) findViewById(R.id.seekBar1);
seekRotate.setProgress(50);
```

```

        seekRotate
            .setOnSeekBarChangeListener(new
SeekBar.OnSeekBarChangeListener() {
                int rotateSeekAwal = 50;

                public void onProgressChanged(SeekBar
seekBar,
                                                    int progress,
boolean fromUser) {
                    int seekbarskrng =
seekRotate.getProgress();
                    int selisihseekbar = rotateSeekAwal -
seekbarskrng;

                    besarrotasi_y -= selisihseekbar / 4;
                    objModel.rotation().y = Rotasi_y +
besarrotasi_y;// besarrotasi_y;
                }

                public void onStartTrackingTouch(SeekBar
seekBar) {
                    rotateSeekAwal = 50;
                }

                public void onStopTrackingTouch(SeekBar
seekBar) {
                    seekRotate.setProgress(50);
                }
            });

```

Source Code 4.15 Kode Program Transformasi Rotasi

Tahap terakhir yakni proses transformasi skala merupakan proses perubahan ukuran objek baik itu perbesaran maupun pengecilan. Kode program dari proses rotasi dapat dilihat pada source code berikut.

```

        btZoomIn = (ImageButton) findViewById(R.id.btnZoomIn);
        btZoomIn.setOnTouchListener(new OnTouchListener() {
            @Override
            public boolean onTouch(View v, MotionEvent event) {
                scalingObj(+0.001f);
                return true;
            }
        });
        btZoomOut = (ImageButton) findViewById(R.id.btnZoomOut);
        btZoomOut.setOnTouchListener(new OnTouchListener() {
            @Override
            public boolean onTouch(View v, MotionEvent event) {
                scalingObj(-0.001f);
                return true;
            }
        });
    }
}

```

Source Code 4.16 Kode Program Transformasi Skala

Source Code 4.16 menjelaskan tentang bagaimana proses implementasi transformasi skala dilakukan, dengan memberikan parameter nilai pada masing-masing titik x, y, dan z. Selanjutnya memberikan aksi pada tombol skala, untuk melakukan transformasi skala baik itu perbesaran maupun pengecilan.

```
//Ubah ukuran object 3d
void scalingObj(float scaling) {
    if (objModel.scale().x >= 0 && objModel.scale().y >= 0
        && objModel.scale().z >= 0) {
        objModel.scale().x += scaling;
        objModel.scale().y += scaling;
        objModel.scale().z += scaling;
    } else {
        objModel.scale().x = objModel.scale().x + 1 /
1000.0f;
        objModel.scale().y = objModel.scale().y + 1 /
1000.0f;
        objModel.scale().z = objModel.scale().z + 1 /
1000.0f;
    }
}
```

Source Code 4.17 Kode Program Aksi Tombol Skala

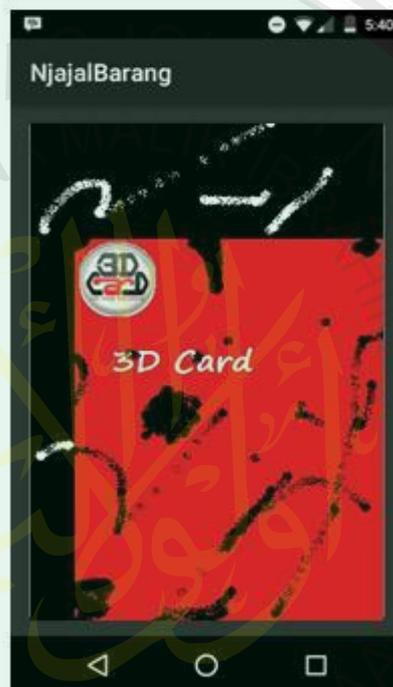
Source Code 4.17 menjelaskan bagaimana aksi pada tombol skala di mana saat melakukan perbesaran skala nilai float objek ditambah sebesar +0,001f, sebaliknya saat melakukan pengecilan skala nilai float objek dikurangi sebesar -0,001f.

4.2 Pengujian Sistem

Pengujian sistem dilakukan untuk mengetahui program yang dibuat mampu berjalan dengan baik atau tidak, serta untuk mengetahui kesesuaian antara alur sistem dengan alur perancangan proses yang telah dibuat dan juga untuk mengetahui kekurangan program yang telah dibuat.

4.2.1 Pengujian Splash Screen

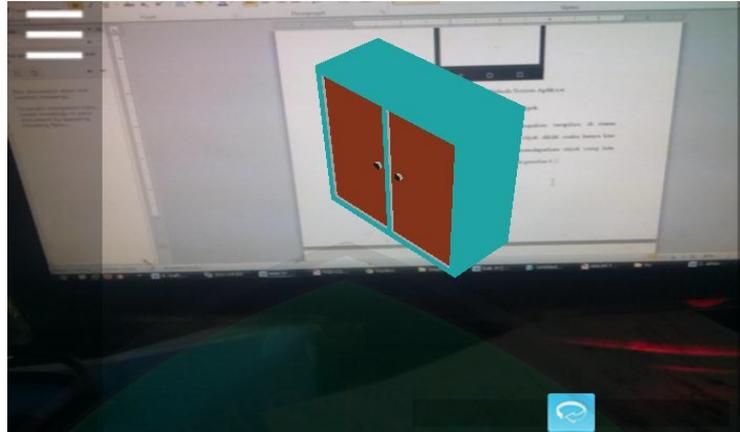
Pengujian Splash Screen pada aplikasi penerapan transformasi geometri untuk visualisasi objek maya berbasis Android ini dilakukan saat pertama kali aplikasi apk dijalankan. Hasil pengujian menunjukkan bahwa proses perpindahan scene awal intro splash screen sampai memasuki halaman utama berhasil dilakukan seperti pada gambar 4.3



Gambar 4.3 Tampilan Proses Splash Screen

4.2.2 Pengujian Menu Objek

Pada pengujian menu objek ini saat tombol menu objek di pilih, maka akan menampilkan list nama-nama menu objek dan akan ditampilkan sesuai urutan seperti pada alur diagram yang dibuat. Hasil pengujian menunjukkan bahwa scene menu objek berhasil sesuai alur diagram yang dibuat seperti pada Gambar 4.4 dan Gambar 4.5



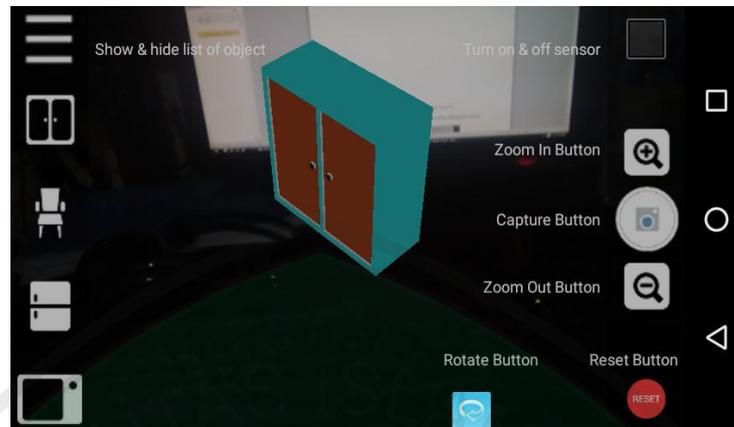
Gambar 4.4 Tampilan Menu Objek Sebelum Diklik



Gambar 4.5 Tampilan Menu Objek Setelah Diklik dan Scroll

4.2.3 Pengujian Halaman Utama

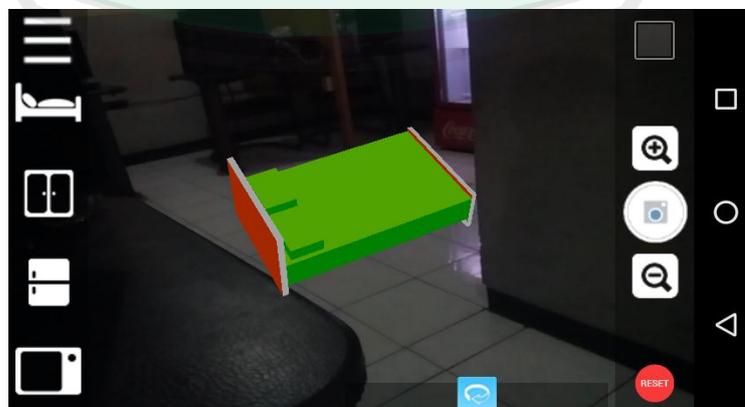
Pengujian dimulai setelah proses splash screen selesai dilakukan, maka akan terjadi perpindahan scene otomatis ke halaman utama aplikasi. Selanjutnya user dapat mengarahkan kamera ke layar secara real dan memilih tombol-tombol yang tersedia selama pemrosesan seperti pada Gambar 4.6



Gambar 4.6 Tampilan Proses Halaman Utama

4.2.4 Pengujian Visualisasi Objek Maya Pada Kamera

Pengujian selanjutnya adalah pengujian pada proses visualisasi atau rendering objek pada kamera. Saat user memilih objek maya berupa image 2 dimensi pada menu, maka aplikasi akan melakukan proses rendering image menjadi model 3 dimensi dan meletakkan image tersebut pada layar kamera secara live. Hasil pengujian menunjukkan proses rendering image objek maya pada kamera berhasil dibuat seperti pada alur pembuatan pada bab sebelumnya, Proses rendering objek maya pada kamera dapat di lihat seperti pada Gambar 4.7



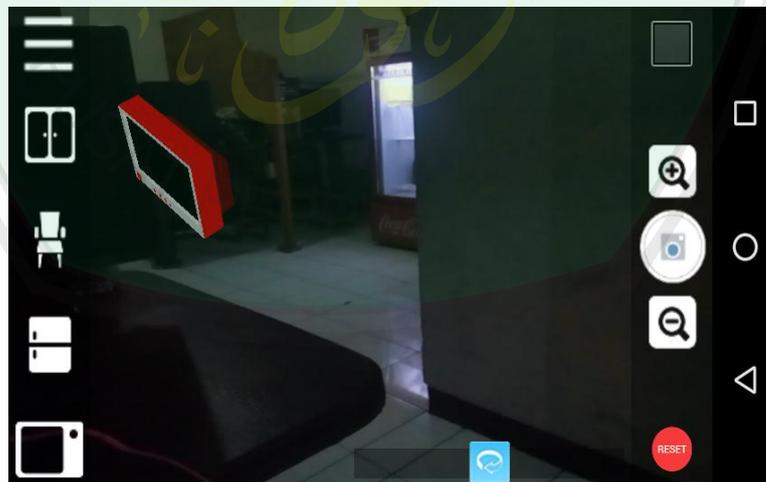
Gambar 4.7 Proses Rendering Objek Pada Kamera

4.2.5 Pengujian Transformasi Geometri

Pengujian transformasi geometri ini dilakukan untuk mengetahui proses transformasi yang terjadi pada objek maya yang dibuat. Terdapat 3 pengujian transformasi geometri yakni pengujian translasi, pengujian rotasi, dan pengujian skala. Berikut adalah gambar dari pengujian transformasi geometri untuk visualisasi objek maya berbasis Android.

a. Pengujian Transformasi Translasi

Pengujian transformasi translasi dilakukan saat user menekan *touch screen* objek dan dilakukan pergeseran pada objek dari titik satu ke titik lain. Hasil pengujian dari proses transformasi translasi atau pergeseran ini berjalan sesuai dengan perhitungan yang telah dijelaskan pada sub bab 2.4.1. Hal ini dapat dilihat seperti pada Gambar 4.8



Gambar 4.8 Proses Pengujian Translasi

b. Pengujian Transformasi Rotasi

Pengujian transformasi rotasi dilakukan saat user menekan tombol rotasi maka objek maya tadi akan terjadi proses perputaran tanpa mengubah posisi maupun ukuran objek. Hasil pengujian dari proses transformasi translasi atau pergeseran ini berjalan sesuai dengan perhitungan yang telah dijelaskan pada sub bab 2.4.2, hanya saja pada proses rotasi ini dilakukan pada sumbu x saja sedangkan sumbu y dan z tidak ada. Proses pengujian transformasi rotasi dapat dilihat pada Gambar 4.9



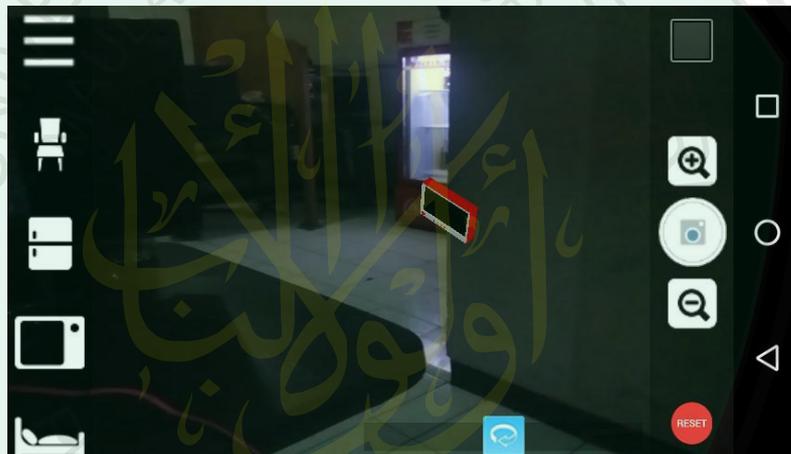
Gambar 4.9 Proses Rotasi Terhadap Sumbu x

c. Pengujian Transformasi Skala

Pengujian transformasi skala dilakukan saat user menekan tombol skala maka objek maya tadi akan terjadi proses penskalaan baik itu perbesaran skala maupun pengecilan skala. Hasil pengujian dari proses transformasi skala ini berjalan sesuai dengan perhitungan yang telah dijelaskan pada sub bab 2.4.3, di mana objek akan mengalami penambahan atau pengurangan ukuran sebesar nilai tertentu yang telah dijelaskan sebelumnya seperti pada Gambar 4.10 dan Gambar 4.11



Gambar 4.10 Proses Perbesaran Skala



Gambar 4.11 Proses Pengecilan Skala

4.2.6 Pengujian Tombol Reset

Pengujian tombol reset dilakukan untuk mengetahui fungsi dari tombol reset pada aplikasi. Tombol reset di sini digunakan untuk mengembalikan posisi objek maya agar kembali ke ukuran dan posisi semula secara default yang telah di inialisasi pada pembahasan sebelumnya. Hasil dari pengujian tombol reset yakni objek berhasil dikembalikan ke posisi awal seperti pada Gambar 4.12



Gambar 4.12 Proses Tombol Reset Aplikasi

4.2.7 Pengujian Tombol Simpan

Pengujian tombol simpan dilakukan untuk mengetahui fungsi dari tombol simpan pada aplikasi. Tombol simpan di sini digunakan untuk menyimpan objek maya yang telah divisualisasikan tadi ke dalam bentuk image berupa gambar dalam format JPG. Hasil dari pengujian tombol simpan di sini berupa image gambar dalam format .jpg yang tersimpan langsung pada memory internal smartphone user setelah tombol simpan ditekan. Hanya saja pada proses penyimpanan di sini file gambar hanya dapat sekali saja tersimpan dalam satu waktu, jadi saat tombol simpan diklik kedua kalinya maka gambar yang tersimpan sebelumnya akan hilang dan tergantikan dengan file image yang baru. Pengujian tombol simpan aplikasi dapat dilihat pada Gambar 4.13



Gambar 4.13 Proses Tombol Simpan Aplikasi

4.3 Hasil Pengujian

Hasil Pengujian (testing) yang telah dilakukan pada aplikasi yang dibuat, untuk mengetahui tingkat keberhasilan aplikasi apakah sudah sesuai dengan rancangan dapat dilihat pada Tabel 4.2

Tabel 4.2 Hasil Pengujian *Black-Box*

No.	Rancangan Proses	Hasil yang diharapkan	Hasil	Keterangan
1	Klik APK aplikasi	Menampilkan Splash Screen sebelum masuk halaman utama	ok	Sesuai dengan rancangan yang dibuat
2	Tunggu proses splash screen selesai	Masuk halaman utama	ok	Sesuai dengan rancangan yang dibuat
3	Klik "Menu Objek" pada halaman	Menampilkan menu-menu objek maya yang	ok	Sesuai dengan rancangan yang

	utama	diinginkan		dibuat
4	Klik “Touch Screen” pada objek	Melakukan proses transformasi translasi	ok	Sesuai dengan rancangan yang dibuat
5	Klik “Tombol Plus” pada halaman utama	Melakukan proses transformasi perbesaran skala	ok	Sesuai dengan rancangan yang dibuat
6	Klik “Tombol Retasi” pada halaman utama	Melakukan proses transformasi rotasi	kurang	Kurang penambahan rotasi pada sumbu putar y dan z
9	Klik “Tombol Reset” pada halaman utama	Melakukan reset pada objek secara default	ok	Sesuai dengan rancangan yang dibuat
10	Klik “Tombol Simpan” pada halaman utama	Melakukan proses simpan objek berupa image	kurang	Hanya bisa menyimpan satu image dalam satu kali penyimpanan

Berdasarkan data hasil pengujian *black-box* pada Tabel 4.2 dapat dilakukan analisa hasil pengujian. Hasil yang diperoleh dari pengujian memperlihatkan bahwa program secara keseluruhan berjalan dengan baik, walaupun perlu ada sedikit perbaikan di beberapa bagian.

4.4 Integrasi Nilai Islam

Ada petuah yang sangat berharga mengenai pentingnya penguasaan bahasa, yaitu *“jika ingin mengenal suatu bangsa, kuasailah bahasanya”*. Petuah ini mempunyai arti bahwa jika kita ingin mengenal, memahami, atau bahkan berdialog dengan suatu bangsa, baik manusia maupun binatang, maka kuasailah bahasanya. Jika kita ingin berdialog dengan orang Inggris, maka kuasailah dan gunakanlah bahasa Inggris. Jika kita ingin berdialog dengan orang Malaysia, maka kuasailah dan gunakanlah bahasa melayu. Jika kita ingin berdialog, mengerti, atau memahami ayat-ayat Qualiyah, yaitu al-Qur’an, maka kuasailah bahasa Arab. Lalu, jika kita ingin berdialog, mengerti, atau memahami ayat-ayat kauniyah, yaitu alam semesta, jagad raya dan isinya, maka bahasa apa yang harus kita kuasai? Bahasa apa yang harus kita gunakan untuk memahami? Jawabannya adalah MATEMATIKA.

Perhatikan tata surya, matahari, bulan, bumi serta planet-planet yang lain. Semuanya berbentuk pola dan memiliki ukuran dan takarannya. Al-Qur’an sudah menyatakan bahwa segala sesuatu telah diciptakan secara matematis. Perhatikan firman Allah surat Al-Qamar ayat 49:

إِنَّا كُلَّ شَيْءٍ خَلَقْنَاهُ بِقَدَرٍ

“Sesungguhnya kami menciptakan segala sesuatu menurut ukuran”

Semua yang ada di alam ini ada ukurannya, ada hitungannya, ada rumusnya, atau ada persamaannya. Ahli matematika atau fisika tidak membuat suatu rumus sedikitpun. Mereka hanya menemukan rumus atau persamaan. Albert

Einstein tidak merumuskan $e=mc^2$, dia hanya menemukan dan menyimbolkannya. Rumus-rumus yang ada sekarang bukan diciptakan manusia, tetapi sudah disediakan. Lihatlah bagaimana Archimedes menemukan hitungan mengenai volume benda melalui media air. Hukum Archimedes itu sudah ada sebelumnya, dan dialah yang menemukan pertama kali melalui hasil menelaah dan membaca ketetapan Allah SWT.

Mengamati dan menemukan keteraturan, kecermatan, kerapian, dan ketelitian aturan atau hukum-hukum dalam alam semesta, Albert Einstein dengan penuh ketakjuban mengatakan *"Tuhan tidak sedang bermain dadu"*. Tuhan tidak sedang main-main, tidak sedang melakukan percobaan, tidak bermain peluang dalam menciptakan alam semesta. Namun, ungkapan Einstein inipun sebenarnya juga basi, karena sekitar 1200 tahun sebelumnya Al-Qur'an surat Al-Anbiya' ayat 16 menyatakan

وَمَا خَلَقْنَا السَّمَاءَ وَالْأَرْضَ وَمَا لَآ عِيبَ بَيْنَهُمَا

"Dan tidaklah Kami ciptakan langit dan bumi dan segala yang ada di antara keduanya dengan bermain-main."

Salah satu kegiatan matematika adalah kalkulasi atau menghitung, sehingga tidak salah jika kemudian ada yang menyebut matematika adalah ilmu hitung atau *ilmu al-hisab*. Dalam urusan hitung menghitung ini, Allah SWT adalah ahlinya. Allah SWT sangat cepat dalam menghitung dan sangat teliti. Kita perhatikan salah satu ayat Al-Qur'an yang menjelaskan bahwa Allah SWT sangat

cepat dalam membuat perhitungan dan sangat teliti. Dalam Al-Qur'an surat An-Nur ayat 39 disebutkan

... وَاللَّهُ سَرِيعُ الْحِسَابِ

“Dan Allah sangat cepat perhitungannya”

Lalu, siapa yang dapat menghitung dengan cepat kalau bukan ahli matematika? Siapa yang dapat menentukan aturan-aturan, rumus-rumus, ukuran-ukuran, dan hukum-hukum jagad raya dengan begitu telitinya kalau bukan ahli matematika? Lalu, kalau Allah SWT serba maha dalam matematika, mengapa kita tidak mau mempelajarinya? Bagaimana kita memahami alam semesta yang menggunakan bahasa matematika kalau kita tidak menguasai matematika?

Geometri adalah sebagian dari matematika yang mengambil persoalan mengenai ukuran, bentuk, dan kedudukan serta sifat ruang. Ilmu geometri secara harfiah berarti pengukuran tentang bumi, yakni ilmu yang mempelajari hubungan di dalam ruang. Bahasa yang digunakan dalam matematika adalah bahasa simbol. Simbol dalam matematika akan bermakna ketika suatu konteks dikaitkan kepadanya. Simbol dalam matematika merupakan hasil abstraksi dari dunia nyata. Suatu simbol sebenarnya mewakili suatu objek baik objek nyata maupun objek abstrak yang bersifat ide. Objek yang diwakili suatu simbol tidak selamanya harus ditampilkan dalam bentuk konkretnya, tetapi dapat diwujudkan dalam bentuk semikonkret, bentuk visualisasi, atau bentuk gambarnya.

Saat suatu bahasa simbol ditemukan tanpa diikuti objek yang diwakili, maka seseorang dapat memahami sesuai objek yang dia berikan meskipun objek

tersebut berbeda dengan objek sebenarnya yang diwakili oleh simbol tersebut. Penentuan objek pada suatu simbol sangat tergantung pada konteks atau sudut pandang atau bahkan tingkat imajinasi orang yang membacanya. Tingkat imajinasi akan mempengaruhi variasi objek yang dikaitkan dengan simbol dan akhirnya juga mempengaruhi visualisasi dari simbol tersebut.

Berikut ini akan diberikan contoh bagaimana memaknai bahasa simbol $x = 3$ yang pada akhirnya dapat memberikan gambaran dan analogi dalam khasanah pemikiran Islam, khususnya dalam memahami Al-Qur'an.

Pertama, ketika menemukan simbol $x = 3$, seseorang mungkin tidak dapat membacanya. Dia dapat melihatnya, tetapi tidak dapat mengejanya. Dia tidak mengetahui bentuk-bentuk yang tertulis dalam simbol tersebut. Jelas, orang ini tidak akan dapat memberikan makna pada simbol tersebut.

Kedua, ketika menemukan simbol $x = 3$, seseorang mungkin dapat membacanya, tetapi tidak mengetahui apa sebenarnya yang ingin disampaikan simbol $x = 3$ tersebut. Dia mengetahui bentuk-bentuk yang tertulis dalam simbol tersebut. Dia tahu huruf x , tanda $=$, dan angka 3, tetapi tidak mengerti x apa dan 3 apa. Jelas, orang ini juga tidak akan dapat memberikan makna pada simbol tersebut.

Ketiga, ketika menemukan simbol $x = 3$, seseorang mungkin dapat membacanya, dan mulai mengaitkan objek-objek pada bentuk-bentuk yang tertulis dalam simbol tersebut. Dia mulai mengerti, dalam sudut pandang dan imajinasinya, bahwa ada objek yang sama dengan 3. Entah objek apa dan 3 dalam

satuan apa saja. Simbol “=” dapat dimaknainya harga, jumlah, atau lainnya. Pada level ini, selain dapat memaknai, seseorang umumnya dapat memberikan visualisasi.

Seseorang yang berpikir matematis geometris dapat mengatakan $x = 3$ sebagai suatu titik pada garis bilangan real (R). Orang ini hanya dapat mengatakannya sebagai titik. Dia tidak dapat memaknai lebih dari itu, karena imajinasinya hanya pada dimensi satu, yaitu R . Seseorang yang imajinasinya lebih tinggi dari dimensi satu (R), yakni pada dimensi dua (R^2), tidak hanya dapat memaknai sebagai titik, tetapi juga dapat memaknainya sebagai garis sejajar sumbu Y yang melalui titik $(3,0)$.

Seseorang yang imajinasinya lebih tinggi dari dimensi dua (R^2), yakni pada dimensi tiga (R^3), tidak hanya dapat memaknai sebagai titik dan garis, tetapi juga dapat memaknainya sebagai bidang sejajar sumbu Y dan sumbu Z yang melalui titik $(5,0,0)$.

Level imajinasi dalam matematika tidak hanya terbatas pada dimensi tiga, tetapi masih ada dimensi empat, lima, bahkan dimensi *takhingga*. Semakin tinggi imajinasi seseorang, maka semakin kompleks visualisasi yang dapat dibuatnya terhadap suatu simbol. Visualisasi itu sendiri masih memerlukan pemaknaan tersendiri. Pertanyaan mengenai apa yang dikehendaki simbol itu, akan berimplikasi pada pertanyaan apa yang ada di balik visualisasi itu. Pemahaman pada visualisasi pada akhirnya merupakan pemahaman pada simbol.

Ilustrasi pemaknaan terhadap simbol $x = 3$ secara matematis geometris ini akan dianalogikan terhadap penafsiran QS Al-Fajr ayat 1-3 berikut :

وَالْفَجْرِ (١) وَالشَّفْعِ وَالْوَتْرِ (٢) وَالْوَتْرِ وَالشَّفْعِ (٣)

Artinya : “*Demi Fajar (1) Demi malam yang sepuluh (2) Dan genap dan yang ganjil (3)*”

Ada apa dengan malam yang 10? Ada apa dengan bilangan genap dan ganjil? Mengapa Allah sampai bersumpah demi yang genap dan yang ganjil? Lihat bagaimana penafsiran yang ada mengenai QS Al-Fajr ayat 3 berikut.

Dalam tafsir Jalalain, kata “*syaf’i*” hanya diartikan sebagai “berpasangan” dan kata “*watr*” diartikan sebagai “sendirian” tanpa penjelasan lebih detail. Dalam tafsir Ibnu Katsir dan tafsir Al-Qurthubi terdapat banyak penafsiran pada kata “*syaf’i*” dan kata “*watr*” di antaranya :

- a. Sebagai hari *arafah* (tanggal 9) dan hari *nahar* (tanggal 10) bulan Dzul Hijjah.
- b. Sebagai shalat shubuh (2 rakaat) dan shalat maghrib (3 rakaat), atau bahkan shalat fardhu keseluruhan. Ada yang berraka’at genap dan berraka’at ganjil.
- c. Sebagai sumpah Allah SWT atas makhluk dan Dia sendiri. *Syaf’i* adalah makhluk dan yang *witr* adalah Allah SWT. Allah SWT adalah *witr.ganjil*, yaitu *wahid* (satu) sedangkan makhluk adalah *syaf’i* atau berpasangan. Ada langit dan bumi, ada darat dan laut, barat dan timur, baik dan jelek, pahit dan

manis, tinggi dan pendek, dan lainnya. Semua ciptaan Allah SWT adalah berpasangan sebagaimana disebutkan dalam QS Adz-Dzariyat ayat 49:

وَمِنْ كُلِّ شَيْءٍ خَلَقْنَا زَوْجَيْنِ لَعَلَّكُمْ تَذَكَّرُونَ

Artinya : “Dan segala sesuatu Kami ciptakan berpasang-pasangan supaya kamu mengingat kebesaran Allah”.

Sekarang kita akan melihat bagaimana kemungkinan seseorang akan memaknai dan memahami ayat tersebut secara matematis geometris. *Pertama*, seseorang tidak dapat membacanya. Jelas orang ini tidak dapat memahaminya. *Kedua*, seseorang dapat membacanya, tetapi tidak paham arti kata dalam bahasa tersebut. Orang ini juga tidak dapat memahaminya. *Ketiga*, seseorang dapat membacanya dan mengerti arti kata dalam bahasa tersebut. kata *syaf'i* dan *watr* dapat dikaitkan dengan objek tertentu seperti pada tafsir yang telah ada, mungkin dikaitkan dengan objek nyata seperti bilangan, tanggal, dan jumlah raka'at, atau bahkan objek abstrak lainnya. Pada level imajinasi tertentu, seseorang akan menemukan visualisasi.

Visualisasi yang dibuat berbentuk segitiga sama sisi dengan angka 1 di puncak segitiga dan angka 2 bagian bawah. Visualisasi ini kemudian memerlukan pemaknaan. Segitiga sama sisi tersebut dapat dipandang sebagai tanda panah yang menunjuk ke atas. Seakan panah tersebut mengatakan, dari bawah ke atas, dari 2 menuju 1, bukan dari 1 menuju 2. Pemaknaan ini sesuai dengan QS Al-Fajr ayat 3 yang menyebut **genap** lebih dahulu, baru yang **ganjil**. Mengapa genap dulu baru ganjil? Jika kemudian genap ditafsirkan “makhluk” yang diciptakan “berpasangan” dan ganjil ditafsirkan “khalik” yang “tunggal/ganjil”, maka

diperoleh makna bahwa untuk menuju yang ganjil, yang satu, yang wahid, yaitu Allah SWT maka mulailah dengan mengenal yang genap, yang berpasangan, yaitu makhluk (alam semesta dan isinya).

Pemaknaan ini sesuai dengan QS Adz-Dzariyat ayat 49 yang menyatakan bahwa segala yang berpasangan adalah sarana untuk mengingat kekuasaan Allah SWT. Bahkan dalam suatu hadits disebutkan bahwa “*berpikirlah tentang ciptaan Allah SWT, jangan berpikir tentang dzat Allah SWT*”. Artinya, bahwa untuk mengenal Allah SWT sarananya adalah dengan mengenal dan mempelajari ciptaan-Nya, yaitu dengan mengenal dan mempelajari alam semesta. Lalu mengapa segitiga sama sisi, bukan yang lain? Pertama perlu diingat kembali stempel kenabian Muhammad SAW yang memuat bangun segitiga sama sisi. Kedua, segitiga sama sisi menunjukkan keseimbangan dalam ukuran. Segala ciptaan Allah SWT adalah seimbang, teratur, dan disusun serapi-rapinya.

Media pembelajaran itu sendiri banyak sekali sebenarnya sudah ada dan diaplikasikan sejak zaman Rasulullah SAW. Beliau dalam mengajarkan ilmu pengetahuan kepada sahabat-sahabatnya tidak lepas dari adanya media sebagai sarana penyampaian materi ajarannya. Seperti yang ditunjukkan oleh hadits riwayat Bukhari dalam kitab Fathul Baari pada bab panjang angan-angan.

عَنْ عَبْدِ اللَّهِ رَضِيَ اللَّهُ عَنْهُ قَالَ : خَطَّ النَّبِيُّ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ خَطًّا مُرَبَّعًا ، وَخَطَّ خَطًّا فِي
الْوَسْطِ خَارِجًا مِنْهُ ، وَخَطَّ خَطًّا صِغَارًا إِلَى هَذَا الَّذِي فِي الْوَسْطِ مِنْ جَانِبِهِ الَّذِي فِي الْوَسْطِ
، وَقَالَ : (هَذَا الْإِنْسَانُ ، وَهَذَا أَجْلُهُ مُحِيطٌ بِهِ أَوْ قَدْ أَحَاطَ بِهِ وَهَذَا الَّذِي هُوَ خَارِجٌ أَمْلُهُ ، وَهَذِهِ
الْخَطُّ الصِّغَارُ الْأَعْرَاضُ ، فَإِنْ أَخْطَأَ هَذَا نَهَشَهُ هَذَا وَإِنْ أَخْطَأَ هَذَا نَهَشَهُ هَذَا)

(رواه البخاري)

Dari Abdullah RA ia berkata: “Nabi SAW membuat gambar persegi empat, lalu menggambar garis panjang di tengah persegi empat tadi dan keluar melewati batas persegi itu. Kemudian beliau juga membuat garis-garis kecil di dalam persegi tadi, disampingnya: (persegi yang digambar Nabi). Dan beliau bersabda: “Ini adalah manusia. Dan (persegi empat) ini adalah ajal yang mengelilinginya, dan garis (panjang) yang diluar ini, adalah cita-citanya. Dan garis-garis kecil ini adalah penghalang-penghalangnya. Jika ia berbuat salah, maka ia akan terkena garis ini, jika ia berbuat salah lagi maka garis ini akan mengenainya” (HR. Bukhari)

Nabi Muhammad SAW menjelaskan garis lurus yang terdapat di dalam gambar adalah manusia, gambar empat persegi yang melingkarinya adalah ajalnya, satu garis lurus yang keluar melewati gambar merupakan harapan dan angan-angan, sementara garis-garis kecil yang ada disekitar garis lurus dalam gambar adalah musibah yang selalu menghadang manusia dalam kehidupannya di dunia.

Melalui visualisasi gambar ini, Nabi SAW menjelaskan di hadapan para sahabatnya, bagaimana manusia dengan cita-citanya dan keinginan-keinginannya yang luas dan banyak, bisa terhalang dengan kedatangan ajal, penyakit-penyakit, atau usia tua. Nabi SAW memberi nasihat pada mereka untuk tidak (sekedah melamun) berangan-angan panjang saja (tanpa realisasi), dan mengajarkan pada mereka untuk mempersiapkan diri menghadapi kematian.

Berdasarkan penjelasan isi kandungan hadits riwayat Bukhari tersebut, Nabi SAW menggunakan sarana gambar-gambar tersebut untuk memberi

gambaran perumpamaan dan mempermudah dalam menyampaikan isi materi yang diajarkannya. Saat dikorelasikan dengan dunia pendidikan, hadits tersebut sangat berkaitan dengan penggunaan media berbentuk visualisasi untuk menyampaikan materi kepada para siswa.



BAB V

PENUTUP

5.1 Kesimpulan

Dari pembahasan yang telah diuraikan sebelumnya, maka penulis dapat menarik kesimpulan, bahwa pembuatan aplikasi penerapan transformasi geometri telah selesai dibuat sesuai tujuan penelitian. Pengujian pada aplikasi ini meliputi pengujian keseluruhan aplikasi. Terdapat 3 proses transformasi dasar dalam pemrosesan visualisasi objek maya yaitu translasi (pergerakan atau perpindahan), penskalaan (memperbesar atau memperkecil), dan rotasi (mereposisi semua titik pada objek).

Hasil yang diperoleh dari pengujian black box memperlihatkan bahwa program secara keseluruhan berjalan dengan baik, walaupun masih diperlukan beberapa perbaikan pada aplikasi khususnya proses transformasi rotasi pada sumbu lain dan tombol simpan.

5.2 Saran

Berdasarkan penelitian ini, saran untuk penelitian yang selanjutnya adalah pengembangan aplikasi visualisasi pada objek maya yang lebih sempurna agar dapat dijadikan bahan pembelajaran bagi ke depannya. Pengembangan dapat dilakukan pada beberapa bagian sebagai berikut:

1. Penambahan materi objek maya.

2. Pengembangan pada bagian rotasi objek tanpa harus menggunakan tombol, sehingga langsung memakai sensor yang ada pada Android.
3. Penambahan warna pada objek maya.
4. Penambahan cara penyimpanan selain dari memory internal.



DAFTAR PUSTAKA

- Abdusysyakir. 2006. *Ada Matematika dalam Al-Qur'an*. Malang: UIN-Press.
- Abdusysyakir. 2007. *Ketika Kyai Mengajar Matematika*. Malang: UIN-Press.
- Baker, M.P. dan Hearn, D. 2003. *Computer Graphics with OpenGL Third Edition*. New Jersey: Prentice Hall.
- Basuki, Achmad. 2009. *Grafik 3 Dimensi*. Surabaya: Politeknik Elektronika Negeri Surabaya.
- Benny A. Pribadi, 2011. *Komunikasi Visual*. Jakarta: PAU-PIPAI UT
- Herbert dan James. 1998. *Visualisasi seni warna dalam simbol*. Jakarta: PT. Gramedia Pustaka Utama.
- H.R. Bukhari. *Kitab Fathul Baari Bab Panjang Angan-Angan*. I/235-236.
- Nalwan, A. 1998. *Pemrograman Animasi dan Game Profesional*. Jakarta: Elex Media Komputindo.
- Purtanto, A.T. 2011. *Visualisasi 3 Dimensi Struktur Rangka Pada manusia*. Jakarta: Universitas Islam Negeri Syarif Hidayatullah
- Safaat, Nazrudin. 2011. *Android Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.
- Shirley, Paul. 2011. *Real Time Augmentation of Video with 3D Graphics*. Ireland: University of Dublin.
- Siregar, I.M. 2011. *Membongkar Source Code Berbagai Aplikasi Android*. Yogyakarta: Gava Media.
- Smith, Neil. 2015. *Android Studio Development Essentials Second Edition*. eBookFrenzy.
- Suyanto, M. 2003. *Multimedia Alat Untuk Meningkatkan Keunggulan Bersaing*. Yogyakarta: Penerbit Andi.
- Syahrir, A.S. 2015. *Visualisasi Struktur Rangka Manusia Berbasis Augmented Reality untuk Mata Pelajaran IPA pada SD Negeri 1 Daya Makassar*. Makassar: STIKOM.
- Tafsir Jalalain, Tafsir Ibnu Katsir dan Tafsir Al-Qurthubi. Juz 30: Ayat 1-3.

Yohannis, A.R. 2006. *Aplikasi Grafika Komputer Untuk Transformasi Geometri 3 Dimensi*. Semarang: Universitas Diponegoro.

Zapata, Belen Cruz. 2013. *Android Studio Application Development*. Birmingham: PACKT Publishing.

