



ITS
Institut
Teknologi
Sepuluh Nopember

TUGAS AKHIR - KI141502

Implementasi *Augmented Reality* untuk Visualisasi Desain Rumah Menggunakan Citra Denah

Muhammad Auliaramadani
NRP 0511164000062

Dosen Pembimbing I
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.

Dosen Pembimbing II
Shintami Chusnul Hidayati, S.Kom., M.Sc., Ph.D.

Departemen Teknik Informatika
Fakultas Teknologi Elektro Dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020



TUGAS AKHIR - KI141502

Implementasi *Augmented Reality* untuk Visualisasi Desain Rumah Menggunakan Citra Denah

Muhammad Auliaramadani
NRP 0511164000062

Dosen Pembimbing I
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.

Dosen Pembimbing II
Shintami Chusnul Hidayati, S.Kom., M.Sc., Ph.D.

Departemen Teknik Informatika
Fakultas Teknologi Elektro Dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember
Surabaya 2020

Halaman ini sengaja dikosongkan



UNDERGRADUATE THESES - KI141502

***Augmented Reality* Implementation for Home Design Visualization Using Floorplans**

Muhammad Auliaramadani
NRP 05111640000062

Supervisor I

Dr. Eng. Nanik Suciati, S.Kom., M.Kom.

Supervisor II

Shintami Chusnul Hidayati, S.Kom., M.Sc., Ph.D.

Department of Informatics Engineering
Faculty Of Intelligent Electrical And Informatics Technology
Sepuluh Nopember Institute of Technology
Surabaya 2020

Halaman ini sengaja dikosongkan

LEMBAR PENGESAHAN

IMPLEMENTASI AR UNTUK VISUALISASI DESAIN RUMAH MENGGUNAKAN CITRA DENAH

TUGAS AKHIR

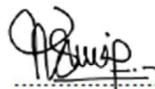
Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Bidang Studi Interaksi Grafika dan Seni
Program Studi S-1 Departemen Informatika
Fakultas Teknologi Elektro dan Informatika Cerdas
Institut Teknologi Sepuluh Nopember

Oleh:

MUHAMMAD AULIARAMADANI
NRP: 05111640000062

Disetujui oleh Pembimbing Tugas Akhir:

1. Dr. Eng. Nanik Suciati, S.Kom., M.Kom.
(NIP. 1971042 8199422 000)



(Pembimbing 1)

2. Shintami Chusnul Hidayati, S.Kom.,
M.Sc., Ph.D.
(NIDK. 0009018705)



(Pembimbing 2)

SURABAYA
AGUSTUS 2020

Halaman ini sengaja dikosongkan

Implementasi *Augmented Reality* untuk Visualisasi Desain Rumah Menggunakan Citra Denah

Nama Mahasiswa : Muhammad Auliaramadani
NRP : 05111640000062
Jurusan : Informatika FTEIC-ITS
Dosen Pembimbing 1 : Dr. Eng. Nanik Suciati, S.Kom.,
M.Kom.
Dosen Pembimbing 2 : Shintami Chusnul Hidayati, S.Kom.,
M.Sc., Ph.D.

ABSTRAK

Smartphone saat ini sudah mendukung fitur *augmented reality*. *Augmented reality* mampu menampilkan objek virtual ke dalam dunia nyata. Fitur pada *smartphone* tersebut tentunya dapat dimanfaatkan dalam bidang pendidikan dan bisnis bagi masyarakat umum dan mahasiswa, khususnya yang mempelajari bidang arsitektur. Dengan teknologi *augmented reality*, sebuah citra denah dapat dipindai menggunakan kamera dan kemudian divisualisasikan dengan model 3 dimensi secara langsung.

Untuk bisa memvisualisasikan citra denah dengan *augmented reality*, aplikasi memindai citra denah yang dikirimkan dari *smartphone* menggunakan algoritma pengenalan titik sudut dan dinding berbasis *deep learning* dan *integer programming*. Hasil pengenalan titik sudut dan dinding yang berupa vektor kemudian dikirimkan kembali ke *smartphone* untuk dibangun model 3 dimensinya menggunakan *Unity 3D*. Model 3 dimensi kemudian ditampilkan di atas marker citra denah menggunakan *Vuforia*.

Berdasarkan hasil pengujian, objek 3 dimensi telah dibangun sesuai dengan denah yang diberikan dengan cukup

akurat. Model 3 dimensi yang berhasil dibangun dapat digunakan untuk mendapatkan sudut pandang baru dari sebuah denah.

Kata kunci: Augmented Reality, Visualisasi Denah, Pengenalan Gambar

Augmented Reality Implementation for Home Design Visualization Using Floor Plans

Name : Muhammad Auliamadani
NRP : 0511164000062
Department : Informatics FTEIC-ITS
First Supervisor : Dr. Eng. Nanik Suciati, S.Kom., M.Kom.
Second Supervisor : Shintami Chusnul Hidayati, S.Kom.,
M.Sc., Ph.D.

ABSTRACT

Smartphones currently supports the augmented reality feature. Augmented reality is capable of displaying virtual objects into the real world. These features are recommended for use in education and business for the general public and students who studies architecture. With AR technology, a floorplan can be scanned using a camera and then visualized with a direct 3-dimensional model.

To visualize floor plans with augmented reality, the application firstly scans the floor plan images sent from the smartphone using a deep learning and integer programming-based corner and wall recognition algorithm. The results of the recognition of corner points and walls in the form of vectors are sent back to the smartphone to build a 3-dimensional model using Unity 3D. The 3-dimensional model is then added over the floor plan image marker using Vuforia.

Based on the test results, 3-dimensional models have been built in accordance with the provided floorplan with moderate accuracy. The 3-dimensional model that was successfully built can then be used to get a new perspective of the floorplan.

Keywords : Augmented Reality, Floorplan Visualization, Image Recognition

KATA PENGANTAR

Bismillaahirrahmaanirrahim.

Alhamdulillahirabbil'alamin, segala puji bagi Allah *subhaanahu wa ta'ala* atas segala karunia dan rahmat-Nya penulis dapat menyelesaikan tugas akhir yang berjudul:

“IMPLEMENTASI AR UNTUK VISUALISASI DESAIN RUMAH MENGGUNAKAN CITRA DENAH”

Selain ini, pada kesempatan ini penulis menghanturkan terima kasih sebesar-besarnya kepada pihak-pihak yang tanpa mereka, penulis tidak akan dapat menyelesaikan buku ini:

1. Allah SWT atas segala limpahan rahmat dan karunia-Nya penulis dapat menyelesaikan Tugas Akhir dan juga perkuliahan di Departemen Informatika ITS.
2. Keluarga saya, terutama Ayah dan Ibu penulis yang tiada hentinya memberikan dukungan doa, nasehat, dan selalu mengingatkan penulis agar tidak lupa menjaga kesehatan saat menyelesaikan Tugas Akhir ini. Serta adik-adik (Maulana, Aulia, dan Bani) yang selalu memberikan semangat serta doa.
3. Ibu Dr. Eng. Nanik Suciati, S.Kom., M.Kom. dan Ibu Shintami Chusnul Hidayati, S.Kom., M.Sc., Ph.D. selaku pembimbing I dan II yang telah membimbing dan memberikan motivasi, nasihat dan bimbingan dalam menyelesaikan Tugas Akhir ini.
4. Bapak dan Ibu dosen Departemen Informatika ITS yang telah banyak memberikan ilmu dan bimbingan yang tak ternilai harganya bagi penulis.

5. Rekan-rekan Admin Laboratorium IGS Ganen, Firman, Dewang, Habib, Yong, Bolt, Anwar, Imam, Giezka, Taqi, Anggun, Elin, dan Tari yang sudah membantu dan memberi hiburan saat bersama-sama berada di laboratorium IGS.
6. Sahabat penulis : Ai dan Nada yang sudah memberikan banyak pengalaman dan canda tawa selama masa perkuliahan.
7. Teman-teman TC 2016, kakak-kakak TC 2014 & 2015 dan adik-adik TC 2017 & 2018 yang tidak dapat disebutkan satu per satu yang selalu membantu, menghibur, menjadi tempat bertukar ilmu serta pembelajaran baru dan berjuang bersama-sama penulis.
8. Serta semua pihak yang telah turut membantu penulis dalam menyelesaikan Tugas Akhir ini.

Penulis menyadari bahwa laporan Tugas Akhir ini masih memiliki banyak kekurangan. Oleh karena itu dengan segala kerendahan hati penulis mengharapkan kritik dan saran dari pembaca untuk perbaikan penulis kedepannya. Selain itu, penulis berharap laporan Tugas Akhir ini dapat berguna bagi pembaca secara umum.

DAFTAR ISI

LEMBAR PENGESAHAN.....	v
ABSTRAK.....	vii
ABSTRACT	ix
KATA PENGANTAR	xi
DAFTAR ISI.....	xiii
BAB 1 BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Batasan Permasalahan.....	2
1.3 Rumusan Masalah.....	2
1.4 Tujuan	3
1.5 Manfaat.....	3
1.6 Metodologi.....	3
1.6.1 Penyusunan Proposal Tugas Akhir.....	3
1.6.2 Studi Literatur	4
1.6.3 Analisis dan Desain Sistem.....	4
1.6.4 Implementasi Perangkat Lunak	4
1.6.5 Pengujian dan Evaluasi	5
1.6.6 Penyusunan Buku	5
1.7 Sistematika Penulisan	5
BAB 2 BAB II DASAR TEORI.....	7
2.1 Visual Studio	7
2.2 <i>Augmented Reality</i>	7
2.3 Bahasa Pemrograman Python	8
2.4 Unity 3D	9
2.5 Pemodelan 3 Dimensi	9
2.6 Flask	10
2.7 <i>Raster-to-Vector Floorplan Transformation</i>	10
2.8 Vuforia.....	11
BAB 3 BAB III PERANCANGAN SISTEM.....	13

3.1	Analisis Sistem.....	13
3.1.1	Spesifikasi Kebutuhan Sistem	13
3.1.2	Identifikasi Pengguna.....	14
3.2	Perancangan Sistem.....	14
3.2.1	Deskripsi Umum Sistem	14
3.2.2	Arsitektur Sistem	16
3.3	Perancangan Skenario Aplikasi	20
3.3.1	Alur Aplikasi	21
3.3.2	Aturan Aplikasi.....	21
3.4	Perancangan Tampilan Antarmuka.....	22
3.4.1	Tampilan Menu.....	22
3.4.2	Tampilan Cara Penggunaan	23
3.4.3	Tampilan Kamera <i>Augmented Reality</i>	23
BAB 4	BAB IV IMPLEMENTASI	25
4.1	Lingkungan Implementasi Perangkat Lunak	25
4.2	Implementasi <i>Import Asset</i> dan <i>Package</i>	26
4.3	Implementasi pada Aplikasi.....	27
4.3.1	Implementasi Halaman Menu	27
4.3.2	Implementasi Halaman Cara Penggunaan.....	29
4.3.3	Implementasi Kamera <i>Augmented Reality</i>	30
BAB 5	BAB V UJI COBA DAN EVALUASI	35
5.1	Lingkungan Uji Coba	35
5.2	Pengujian Fungsionalitas	36
5.2.1	Uji Coba Halaman Awal.....	36
5.2.2	Uji Coba Halaman Cara Penggunaan	38
5.2.3	Uji Coba Halaman Kamera <i>Augmented Reality</i>	40
5.2.4	Uji Coba Pengenalan Denah.....	42
5.2.5	Pengujian Pengguna	54
5.2.6	Skenario Pengujian Pengguna.....	54
5.2.7	Daftar Penguji Perangkat Lunak	55
5.2.8	Hasil Pengujian Pengguna.....	56
5.2.9	Kritik dan Saran Pengguna	57

5.3 Evaluasi Pengujian	58
BAB 6 BAB VI KESIMPULAN DAN SARAN.....	59
6.1 Kesimpulan.....	59
6.2 Saran	59
DAFTAR PUSTAKA	61
LAMPIRAN.....	62
BIODATA PENULIS.....	68

Halaman ini sengaja dikosongkan

DAFTAR GAMBAR

Gambar 2-1 Contoh aplikasi <i>Augmented Reality</i>	8
Gambar 3-1 Rancangan arsitektur aplikasi.....	16
Gambar 3-2 Alur kerja sistem aplikasi.....	17
Gambar 3-3 Contoh dinding yang dibuat menggunakan koordinat dilihat dari atas	19
Gambar 3-4 Contoh posisi dinding yang dibuat menggunakan nilai tengah koordinat.....	19
Gambar 3-5 Rancangan antarmuka halaman menu.....	22
Gambar 3-6 Rancangan antarmuka halaman Pemilihan Gambar	23
Gambar 3-7 Rancangan antarmuka halaman Kamera AR.....	24
Gambar 4-1 Implementasi <i>import asset</i> dan <i>package</i>	26
Gambar 4-2 Implementasi antarmuka Menu.....	27
Gambar 4-3 Implementasi antarmuka Cara Penggunaan	29
Gambar 4-4 Implementasi antarmuka Kamera AR	31
Gambar 5-1 Implementasi antarmuka Menu.....	36
Gambar 5-2 Implementasi antarmuka Cara Penggunaan	39
Gambar 5-3 Implementasi antarmuka Kamera AR	40

Halaman ini sengaja dikosongkan

DAFTAR TABEL

Tabel 3-1 Kebutuhan fungsional sistem.....	13
Tabel 3-2 Kebutuhan non-fungsional sistem.....	14
Tabel 4-1 Spesifikasi perangkat komputer	25
Tabel 4-2 Spesifikasi perangkat <i>smartphone</i>	26
Tabel 5-1 Spesifikasi Lingkungan Uji Coba	35
Tabel 5-2 Hasil uji coba halaman Menu.....	37
Tabel 5-3 Hasil uji coba halaman Pemilihan Gambar.....	39
Tabel 5-4 Hasil uji coba halaman Kamera AR.....	41
Tabel 5-5 Hasil uji coba pengenalan denah.....	43
Tabel 5-6 Rentang nilai	55
Tabel 5-7 Kuesioner Pengguna	55
Tabel 5-8 Daftar Penguji Perangkat Lunak.....	56
Tabel 5-9 Hasil Akhir Pengujian Pengguna	56
Tabel 5-10 Kritik dan Saran Pengguna	57

Halaman ini sengaja dikosongkan

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi saat ini menyebabkan banyak bisnis memanfaatkan teknologi virtual reality dan augmented reality untuk menarik perhatian pelanggan. Hal ini menyebabkan perkembangan teknologi virtual reality dan augmented reality menjadi topik yang populer di Indonesia. Salah satu bidang yang mampu memanfaatkan implementasi teknologi augmented reality untuk bisnis adalah bidang arsitektur, seperti menampilkan preview dalam bentuk model 3 dimensi dari citra denah yang ingin dilihat oleh pelanggan, sehingga produk dapat dilihat secara langsung oleh pelanggan tanpa membuang waktu untuk menyiapkan produk setiap kali ada pelanggan dan juga mampu melayani beberapa pelanggan di waktu yang sama.

Banyak metode yang dapat digunakan untuk mengimplementasi augmented reality, salah satunya dengan memanfaatkan homografi dan ekstraksi fitur untuk mencari penanda yang telah ditentukan sebelumnya dan menggambar model di atasnya. Tugas akhir ini akan menggunakan algoritma yang dikembangkan oleh Liu *et al* [1], dengan memanfaatkan PyTorch. Setelah mendapatkan hasil pengenalan dari algoritma berupa titik koordinat, titik koordinat ini akan dipakai untuk membentuk model 3D dari denah yang telah diberikan. Setelah model 3D dibuat, maka akan digambar di atas denah di dalam video real-time dari kamera komputer, laptop, atau *smartphone*.

Diharapkan aplikasi yang dibuat pada tugas akhir ini dapat membantu banyak orang, terutama yang memiliki latar belakang arsitektur seperti mahasiswa Departemen Arsitektur atau Teknik

Sipil. Karena menggunakan pengenalan objek secara otomatis, maka pembuatan dan pengembangan denah juga lebih cepat.

1.2 Batasan Permasalahan

Batasan masalah pada tugas akhir ini antara lain:

1. Menggunakan *smartphone* Android yang mendukung *augmented reality*.
2. Denah yang dipakai adalah denah bangunan satu lantai.
3. Model yang dihasilkan tidak memiliki detail tambahan seperti tekstur dan furnitur.
4. Model yang dihasilkan tidak memiliki informasi semantik seperti nama ruangan.
5. Bentuk ruang yang bisa dideteksi adalah ruangan dengan bentuk segiempat, ruangan dengan bentuk lain tidak terdeteksi dan tidak akan digambar dalam bentuk model 3D.

1.3 Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana mengenali titik sudut dari citra denah secara otomatis?
2. Bagaimana membangun model 3 dimensi dari titik sudut yang telah dikenali dan ditampilkan ke pengguna?
3. Bagaimana evaluasi performa aplikasi dalam mengenali citra denah dan membangun model 3 dimensi?

1.4 Tujuan

Tujuan dari pembuatan tugas akhir ini antara lain:

1. Dapat meng-*generate* model 3 dimensi yang sesuai dengan gambar denah 2 dimensi.
2. Dapat menampilkan model 3 dimensi pada *smartphone* menggunakan teknologi *augmented reality*.

1.5 Manfaat

Tugas akhir ini diharapkan mampu membantu mempermudah pengenalan dan visualisasi denah kepada masyarakat umum, utamanya yang memiliki profesi berkaitan dengan arsitektur atau mahasiswa yang sedang mempelajari arsitektur.

1.6 Metodologi

Langkah-langkah yang ditempuh dalam pengerjaan tugas akhir ini adalah sebagai berikut:

1.6.1 Penyusunan Proposal Tugas Akhir

Proposal tugas akhir ini berisi tentang deskripsi pendahuluan dari tugas akhir yang akan dibuat. Pendahuluan ini terdiri atas hal yang menjadi latar belakang diajukannya usulan tugas akhir, rumusan masalah yang diangkat, batasan masalah untuk tugas akhir, tujuan dari pembuatan tugas akhir, dan manfaat dari hasil pembuatan tugas akhir. Selain itu dijabarkan pula tinjauan pustaka yang digunakan sebagai referensi pendukung pembuatan tugas akhir. Sub bab metodologi berisi penjelasan mengenai tahapan penyusunan tugas akhir mulai dari penyusunan proposal hingga penyusunan buku tugas akhir. Terdapat pula sub bab jadwal kegiatan yang menjelaskan jadwal pengerjaan tugas akhir.

1.6.2 Studi Literatur

Pada studi literatur ini akan dipelajari sejumlah referensi yang relevan terhadap tugas akhir yang akan dikerjakan. Antara lain seperti mempelajari *paper* yang diadaptasi oleh tugas akhir ini, menggunakan Python dan PyTorch untuk mengekstrak fitur, Unity 3D untuk pembuatan model, dan Vuforia untuk menampilkan *augmented reality* di komputer dan *smartphone*. Studi literatur yang didapatkan untuk tugas akhir ini berasal dari buku, internet, dan materi-materi perkuliahan yang berhubungan dengan aplikasi yang akan dibangun.

1.6.3 Analisis dan Desain Sistem

Pada tahap ini dilakukan analisis dan pendefinisian kebutuhan sistem untuk masalah yang dihadapi, terutama analisis terkait bagaimana skenario latihan berbicara di depan umum yang akan diterapkan pada sistem. Selanjutnya, dilakukan perancangan sistem dengan beberapa tahap sebagai berikut:

- a. Analisis aktor yang terlibat di dalam sistem.
- b. Pengenalan denah oleh mesin dan pembuatan model 3 dimensi dari hasil pengenalan denah.
- c. Perancangan aplikasi *augmented reality* pada *desktop* dan *smartphone*.
- d. Perancangan lingkungan dan antarmuka aplikasi.
- e. Perancangan diagram kasus penggunaan sistem.

1.6.4 Implementasi Perangkat Lunak

Aplikasi ini akan dibangun dengan bahasa pemrograman Python dan Javascript. Aplikasi ini akan dibangun menggunakan Integrated Development Environment (IDE) Visual Studio Code dan Anaconda.

1.6.5 Pengujian dan Evaluasi

Pada tahap ini dilakukan pengujian sistem kepada pengguna secara langsung. Pengujian dan evaluasi sistem dilakukan untuk mengevaluasi hasil analisis program. Tahapan-tahapan dari pengujian adalah sebagai berikut:

1. Pengujian usabilitas

Pengujian usabilitas dilakukan dengan cara memberikan demonstrasi program dan melakukan survei ke pengguna yaitu mahasiswa departemen arsitektur ITS. Survei digunakan untuk mengukur tingkat kegunaan dari aplikasi yang dibuat dalam membantu pengguna.

2. Pengujian terhadap fitur-fitur yang terdapat di aplikasi.

1.6.6 Penyusunan Buku

Pada tahap ini dilakukan penyusunan buku yang menjelaskan seluruh konsep, teori dasar dari metode yang digunakan, implementasi, serta hasil yang telah dikerjakan sebagai dokumentasi dari pelaksanaan Tugas Akhir.

1.7 Sistematika Penulisan

Buku Tugas Akhir ini bertujuan untuk mendapatkan gambaran dari pengerjaan Tugas Akhir ini. Selain itu, diharapkan dapat berguna untuk pembaca yang tertarik untuk melakukan pengembangan lebih lanjut. Secara garis besar, buku Tugas Akhir terdiri atas beberapa bagian seperti berikut ini.

Bab I Pendahuluan

Bab ini berisi latar belakang masalah, tujuan dan manfaat pembuatan Tugas Akhir, permasalahan, batasan masalah, metodologi yang digunakan, dan sistematika penyusunan Tugas Akhir.

Bab II Dasar Teori

Bab ini membahas beberapa teori penunjang yang berhubungan dengan pokok pembahasan dan mendasari pembuatan Tugas Akhir ini.

Bab III Analisis dan Perancangan Sistem

Bab ini membahas mengenai perancangan perangkat lunak. Perancangan perangkat lunak meliputi perancangan data, arsitektur, proses dan perancangan antarmuka pada aplikasi.

Bab IV Implementasi

Bab ini berisi implementasi dari perancangan perangkat lunak.

Bab V Pengujian dan Evaluasi

Bab ini membahas pengujian dari aplikasi yang dibuat dengan melihat keluaran yang dihasilkan oleh aplikasi dan evaluasi untuk mengetahui kemampuan aplikasi serta mengetahui penilaian aspek kegunaan (*usability*) dari perangkat lunak.

Bab VI Kesimpulan dan Saran

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan. Bab ini membahas saran-saran untuk pengembangan sistem lebih lanjut.

Daftar Pustaka

Merupakan daftar referensi yang digunakan untuk mengembangkan Tugas Akhir.

Lampiran

Merupakan bab tambahan yang berisi daftar istilah yang penting pada aplikasi .

BAB II

DASAR TEORI

Bab ini berisi pembahasan mengenai teori-teori dasar yang digunakan dalam Tugas Akhir.

2.1 Visual Studio

Microsoft Visual Studio merupakan sebuah aplikasi yang terdiri dari *compiler*, SDK, *Integrated Development Environment* (IDE), dan dokumentasi yang biasa digunakan untuk mengembangkan aplikasi personal, aplikasi web, aplikasi *mobile*, aplikasi konsol maupun komponen aplikasi pada Microsoft Visual Studio. Banyak pengembang aplikasi yang mengembangkan aplikasinya menggunakan Microsoft Visual Studio karena banyak fungsi yang telah disediakan oleh Microsoft Visual Studio, sehingga pengembang lebih mudah dan sangat terbantu dalam mengembangkan maupun membuat aplikasi.[2]

2.2 Augmented Reality

Augmented reality atau realitas tertambah adalah teknologi yang menggabungkan benda maya dua dimensi dan ataupun tiga dimensi ke dalam sebuah lingkungan nyata tiga dimensi lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata (*real time*). *Augmented reality* bersifat sekedar menambahkan atau melengkapi kenyataan.

Benda-benda maya menampilkan informasi yang tidak dapat diterima oleh pengguna dengan indranya sendiri. Hal ini membuat realitas tertambah sesuai sebagai alat untuk membantu persepsi dan interaksi pengguna dengan dunia nyata. Informasi yang ditampilkan oleh benda maya membantu pengguna melaksanakan kegiatan-kegiatan dalam dunia nyata. Contoh aplikasi *augmented reality* bisa dilihat pada gambar 2.1.



Gambar 2-1 Contoh aplikasi *Augmented Reality*

2.3 Bahasa Pemrograman Python

Bahasa Python adalah bahasa yang interpretatif, berorientasi objek, dan tingkat tinggi dengan semantik yang dinamis. Sintaksis Python yang sederhana dan mudah dipelajari menekankan keterbacaan dan akhirnya mengurangi biaya pemeliharaan program.

Python mendukung modul dan paket, yang mendorong modularitas program dan penggunaan kembali kode. Interpreter Python dan library yang tersedia dalam bentuk sumber atau biner tanpa biaya untuk semua platform utama, dan dapat didistribusikan secara bebas.

Karena tidak ada langkah kompilasi, siklus edit-uji-debug sangat cepat. Debugging program Python mudah: bug atau input yang buruk tidak akan pernah menyebabkan kesalahan segmentasi. Sebaliknya, ketika penerjemah menemukan kesalahan, itu menimbulkan pengecualian. Ketika program tidak menangkap pengecualian, penerjemah mencetak jejak *stack*.^[3]

2.4 Unity 3D

Unity 3D adalah sebuah aplikasi yang digunakan untuk mengembangkan *game* berbasis *multi-platform*. Unity dapat digunakan untuk membuat sebuah *game* yang bisa digunakan pada perangkat komputer, ponsel pintar android, iPhone, PS3, dan bahkan Xbox.

Unity merupakan sebuah *tool* yang terintegrasi, untuk membuat *game*, arsitektur bangunan dan simulasi. Unity juga bisa digunakan untuk permainan PC dan permainan *online*. Untuk permainan *online* diperlukan sebuah *plugin*, yaitu Unity Web Player, sama halnya dengan Flash Player pada browser.

Unity tidak dirancang untuk proses desain atau *modelling*, dikarenakan Unity bukanlah sebuah *tools* untuk mendesain, Unity hanyalah sebuah *game engine* 2D atau 3D. Banyak hal yang bisa dilakukan dengan Unity dengan berbagai fitur yang dimilikinya, seperti adanya fitur *audio reverb zone*, *particle effect*, dan *sky box* untuk menambahkan langit. Fitur *scripting* yang disediakan mendukung 3 bahasa pemrograman, JavaScript, C#, dan Boo.[4]

2.5 Pemodelan 3 Dimensi

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 bisa diperlihatkan secara 3 dimensi, sehingga banyak yang menyebut hasil ini sebagai pemodelan 3 dimensi.

Pemodelan 3 dimensi memiliki beberapa aspek yang harus diperhatikan, yaitu pendeskripsian objek, tujuan dari model, tingkat kerumitan, kesuaian dan kenyamanan, serta kemudahan manipulasi objek. Pemodelan 3 dimensi dapat dilakukan dengan

bantuan aplikasi-aplikasi, seperti Blender, K-3D, Google SketchUp, dan masih banyak lainnya.

2.6 Flask

Flask adalah kerangka kerja aplikasi web WSGI yang ringan. Dirancang untuk dimulai dengan cepat dan mudah, dengan kemampuan meningkatkan aplikasi yang kompleks. Program ini dimulai sebagai *wrapper* sederhana untuk *Werkzeug* dan *Jinja* dan telah menjadi salah satu kerangka kerja aplikasi web Python paling populer.

Flask bisa menawarkan saran dan sugesti mengenai program yang dikerjakan, tetapi tidak memberlakukan dependensi atau tata letak proyek. Terserah pada pengembang untuk memilih alat dan perpustakaan yang ingin mereka gunakan. Ada banyak ekstensi yang disediakan oleh komunitas yang membuat penambahan fungsionalitas baru menjadi mudah.[5]

2.7 *Raster-to-Vector Floorplan Transformation*

Raster-to-Vector Floorplan Transformation adalah sebuah algoritma yang digunakan untuk mengubah sebuah gambar raster menjadi sebuah gambar vector melalui serangkaian pengenalan dan fungsi matematika. Algoritma yang dipakai pada buku ini adalah algoritma yang dikembangkan oleh Liu *et al* [1] pada tahun 2017.

Algoritma ini tersusun atas 2 bagian, yang dibagi berdasarkan proses yang digunakan dan hasil yang dikeluarkan. Bagian pertama algoritma menggunakan *deep network* untuk mengekstrak informasi geometri dan semantik tingkat rendah ke dalam serangkaian persimpangan (pada dasarnya titik-titik yang diberi tipe) sebagai representasi pertama. Bagian kedua, integer programming (IP) digunakan untuk mengagregasi persimpangan ke dalam kumpulan primitif sederhana (misal, dinding sebagai

garis dan ikon sebagai kotak). IP memberlakukan batasan tingkat yang lebih tinggi di antara primitif, misalnya, ruang sebagai rantai primitif dinding yang membentuk loop tertutup. Ini memastikan bahwa langkah *post-processing* yang sederhana dapat menghasilkan representasi vektor yang lengkap.

Pada tugas akhir ini, algoritma ini akan digunakan untuk mengubah gambar denah yang telah dikirimkan oleh pengguna menggunakan kamera laptop atau *smartphone* menjadi serangkaian titik dan garis yang kemudian bisa diubah secara otomatis menjadi objek 3 dimensi oleh *three.js*. Objek inilah yang akan ditampilkan di dalam *augmented reality*.

2.8 Vuforia

Vuforia merupakan platform yang digunakan untuk membangun aplikasi augmented reality (AR), virtual reality (VR), dan mixed reality (MR), dengan menggunakan pelacakan dan kinerja yang sangat baik pada berbagai perangkat keras (termasuk didalamnya perangkat selular seperti smartphone Android maupun iOS dan Head Mounted Display untuk mixed reality seperti Microsoft HoloLens). Integrasi antara Unity 3D dan Vuforia sangat memudahkan dalam pengembangan aplikasi dan game untuk Android dan iOS, hanya menggunakan proses drag and drop. Sampel dari Vuforia pun bisa diperoleh di Unity Asset Store dengan beberapa contoh yang berguna dalam mendemonstrasikan fitur-fitur terpentingnya.

Vuforia mendukung sangat banyak perangkat pihak ketiga, seperti kacamata AR maupun MR dan perangkat Virtual Reality (VR) dengan kamera seperti GearVR. Keunggulan dari Vuforia adalah dapat menggunakan kamera untuk menguji game AR atau MR serta aplikasi-aplikasi lainnya yang dibangun di Unity dengan menggunakan Vuforia.[6]

Halaman ini sengaja dikosongkan

BAB III

PERANCANGAN SISTEM

Bab ini membahas tentang desain dan perancangan aplikasi visualisasi denah rumah menggunakan *augmented reality* pada *smartphone* dan laptop. Pembahasan yang dilakukan meliputi analisis sistem, perancangan sistem, skenario simulasi, dan perancangan antar muka sistem.

3.1 Analisis Sistem

Sub bab ini akan membahas tentang analisis kebutuhan sistem, meliputi spesifikasi kebutuhan sistem, baik itu kebutuhan fungsional sistem maupun kebutuhan non-fungsional sistem, dan identifikasi pengguna sistem.

3.1.1 Spesifikasi Kebutuhan Sistem

Pada sistem ini terdapat beberapa kebutuhan fungsional dan kebutuhan non-fungsional yang mendukung berjalannya sistem. Kebutuhan fungsional sistem dapat dilihat pada Tabel 3.1, sedangkan kebutuhan non-fungsional sistem dapat dilihat pada Tabel 3.2.

Tabel 3-1 Kebutuhan fungsional sistem

Kode	Deskripsi
F1	Perangkat <i>smartphone</i> dan laptop menangkap gambar/ objek target.
F2	Perangkat <i>smartphone</i> dan laptop dapat menampilkan objek 3 dimensi dengan baik.
F3	Sistem dapat mengenali gambar yang ditangkap secara langsung.

F4	Sistem dapat membuat model 3D dari hasil pengenalan gambar dengan akurat.
----	---

Tabel 3-2 Kebutuhan non-fungsional sistem

Kode	Deskripsi
NF1	Sistem dapat dijalankan di komputer dan <i>smartphone</i> .
NF2	Sistem berjalan dengan lancar di <i>smartphone</i> beresolusi layar 1920x1080 piksel.
NF3	Sistem dapat membuka kamera dan memuat objek 3 dimensi kurang dari 30 detik.
NF4	Sistem memiliki tampilan antar muka yang mudah dipahami.
NF5	Sistem dapat digunakan dengan mudah.

3.1.2 Identifikasi Pengguna

Pengguna dari aplikasi ini adalah masyarakat umum dan lebih dikhususkan pada mahasiswa dan pekerjaan yang berkaitan erat dengan arsitektur. Namun, semua pengguna berhak menggunakan seluruh fungsionalitas yang terdapat pada sistem.

3.2 Perancangan Sistem

Sub bab ini membahas tentang bagaimana sistem ini dirancang, meliputi deskripsi umum sistem dan arsitektur sistem.

3.2.1 Deskripsi Umum Sistem

Aplikasi yang akan dibangun pada tugas akhir ini merupakan sistem aplikasi *augmented reality* yang berguna sebagai media penunjang dalam membaca atau memahami denah rumah dalam bentuk 3D untuk mendapatkan sudut pandang lebih. Denah yang bisa dipakai untuk tugas akhir ini terbatas pada denah yang tidak memiliki sudut selain sudut 90 derajat, dan hanya memiliki satu lantai. Pengguna awalnya akan mengambil gambar denah, dan setelah denah dipotret gambar dikirim ke server yang akan mengenali denah yang telah diambil. Setelah server selesai mengenali denah, pengguna akan di *redirect* ke halaman baru dimana sistem *augmented reality* ditampilkan.

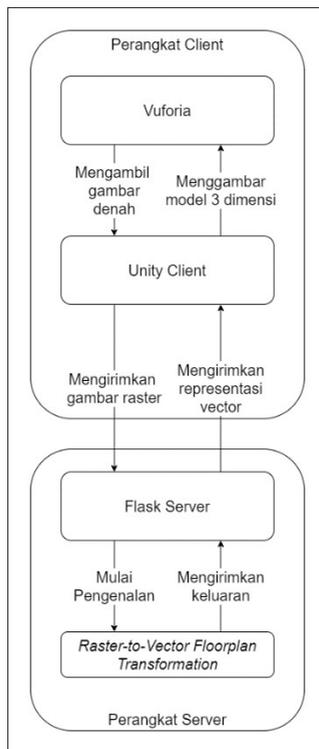
Sistem ini terdiri dari dua bagian utama, yaitu aplikasi client di *smartphone* atau laptop yang menampilkan model 3D dan server yang digunakan untuk mengenali denah dan pembuatan model 3D.

Pembangunan sistem ini dimulai dengan membangun algoritma yang akan digunakan untuk mengenali denah. Algoritma yang digunakan dikembangkan oleh Liu *et al* [1] dan kemudian dimodifikasi untuk bisa digunakan pada tugas akhir ini. Algoritma ini memiliki masukan sebuah gambar atau sebuah list berisi gambar-gambar yang akan dikenali dan memberi keluaran berupa titik-titik koordinat yang diperlukan untuk membangun dinding pada model 3D. Titik-titik koordinat yang telah dikeluarkan kemudian dikirim ke halaman web untuk kemudian dibentuk model 3D nya.

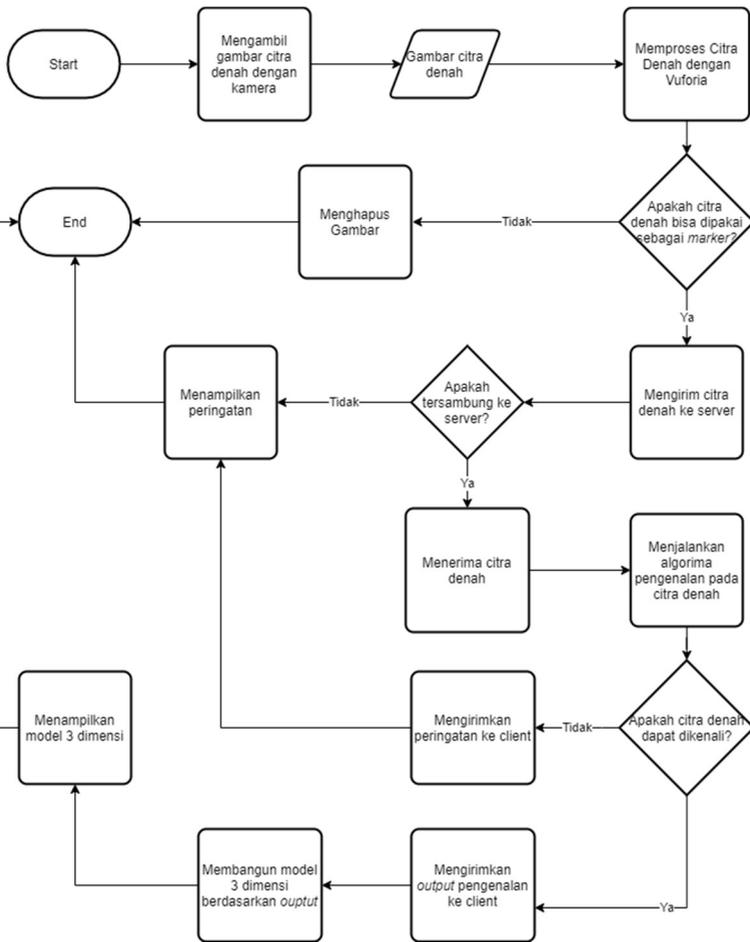
Pembangunan sistem terakhir adalah pembangunan server yang akan digunakan untuk komunikasi antara client yang menampilkan *augmented reality* dengan server yang akan mengenali denah. Server dibangun menggunakan Flask, sebuah kerangka kerja berbasis Python, sehingga mampu menjalankan algoritma pengenalan denah dan kemudian mengirimkan keluaran algoritma langsung ke client.

3.2.2 Arsitektur Sistem

Dalam pengerjaan sistem aplikasi pada tugas akhir ini, peralatan yang digunakan berupa *Personal Computer* untuk mengembangkan aplikasi, *smartphone* dan laptop untuk menggunakan serta menguji aplikasi, dan gambar denah untuk dikenali dan divisualisasikan.



Gambar 3-1 Rancangan arsitektur aplikasi



Gambar 3-2 Alur kerja sistem aplikasi

Alur dari aplikasi ini adalah mula-mula pengguna meletakkan denah yang akan dikenali di depan kamera. Kamera kemudian mengambil gambar denah yang kemudian dikirim ke server yang menjalankan proses pengenalan dan pembuatan model 3D.

Setelah server menerima gambar dari client, server menjalankan algoritma pengenalan denah dan mempersiapkan sistem *tracking* dari gambar yang dikirim. Pengenalan denah sendiri dilakukan dengan menjalankan algoritma pengenalan yang kemudian diberi masukan berupa gambar yang telah didapat dari server.

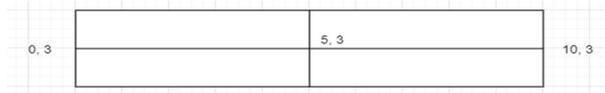
Server mengenali denah menggunakan algoritma yang dikembangkan oleh Liu *et al*[1] untuk mengubah gambar raster denah menjadi representasi vektor yang bisa dibaca oleh computer. Algoritma ini bekerja dengan pertama menggunakan *deep network* untuk mengekstrak informasi geometrik yang akan menjadi masukan untuk *integer programming* (IP) yang mengubah informasi ini menjadi representasi vektor yang lengkap. Representasi vektor inilah yang akan dikirim kembali ke client untuk diubah menjadi model 3 dimensi.

Setelah server selesai mengenali denah dan sistem *tracking* telah siap dipakai, server mengirimkan hasil keluaran algoritma pengenalan dan sistem *tracking* ke client. Apabila sebuah gambar denah tidak dapat dikenali, maka server akan mengirimkan array yang kosong, dan komputer atau *smartphone* akan menampilkan peringatan bahwa denah tidak dapat dikenali. Komputer atau *smartphone* yang digunakan sebagai client kemudian dapat menggambar model yang diterima diatas denah yang telah disediakan. Setelah client menerima informasi yang dibutuhkan dari server, client akan membangun model 3 dimensi menggunakan titik-titik koordinat dari informasi yang diberikan. Titik koordinat ini diberikan dalam bentuk nested array, dimana array paling dalam berisi 4 titik, dan titik ini adalah titik ujung dari sebuah dinding. Contohnya dapat dilihat pada gambar 3.1, dengan contoh titik [0 , 3, 10, 3].



Gambar 3-3 Contoh dinding yang dibuat menggunakan koordinat dilihat dari atas

Client akan membangun semua dinding yang ada di dalam Unity 3D, dimana pertama harus dibuat sebuah geometri yang akan dipakai untuk semua dinding. Geometri yang dipakai dari Unity 3D adalah Cube, untuk membuat objek berbentuk balok. Setelah itu dibuat material yang akan dipakai oleh objek, disini digunakan material basic dengan warna tekstur matte, sehingga lebih mudah dilihat meski dalam kondisi pencahayaan yang terang. Setiap dinding dibuat dulu objeknya menggunakan geometri dan material yang telah ditentukan, kemudian diatur posisinya sesuai titik koordinat yang telah didapat. Karena dalam Unity 3D posisi sebuah objek itu berada pada titik tengahnya, maka untuk posisi dinding dihitung nilai tengah dari keempat koordinat setiap dinding. Dapat dilihat pada gambar 3.2 tentang posisi dinding.



Gambar 3-4 Contoh posisi dinding yang dibuat menggunakan nilai tengah koordinat

Dapat dilihat pada gambar 3.2, dengan contoh koordinat [0, 3, 10, 3], nilai tengah yang dipakai untuk posisi dinding adalah 5,3.

Penghitungannya adalah $x = \frac{\text{koordinat}[0] + \text{koordinat}[2]}{2}$ dan $y = \frac{\text{koordinat}[1] + \text{koordinat}[3]}{2}$.

Setelah model selesai dibuat, maka model dimasukkan ke dalam sistem *augmented reality* menggunakan Vuforia SDK, dimana sistem *tracking* yang telah dipersiapkan digunakan untuk mengenali denah sebagai *anchor point* sehingga model dapat digambar posisi persis diatas gambar denah di dalam kamera.

Vuforia SDK menggunakan mode *Used Defined Target*, dimana target yang akan digunakan sebagai *anchor point* ditentukan oleh pengguna. Vuforia memindai gambar dari kamera *smartphone* secara *real-time*, sehingga ketika pengguna ingin menggunakan gambar tertentu yang diambil dari kamera Vuforia dapat langsung menambah gambar tersebut sebagai *anchor point*. Akan tetapi, Vuforia hanya akan menambahkan sebuah gambar menjadi *anchor point* apabila gambar tersebut memiliki *feature point* yang cukup untuk mendapatkan *anchor point* yang stabil. Apabila Vuforia menilai sebuah gambar denah tidak bisa dijadikan *anchor point* yang stabil, maka aplikasi tidak akan mengirim gambar ke server dan menampilkan peringatan bahwa gambar denah tidak bisa digunakan.

Unity kemudian menggambar model 3 dimensi diatas *anchor point* yang telah disediakan oleh Vuforia SDK, dan pengguna bisa melihat representasi 3 dimensi dari denah yang telah mereka persiapkan.

3.3 Perancangan Skenario Aplikasi

Pada subbab ini menjelaskan skenario dari aplikasi untuk menentukan kondisi sukses atau tidak. Selain itu dibahas juga aturan dari aplikasi agar kondisi berakhir sukses dalam pengenalan dan pemunculan objek.

3.3.1 Alur Aplikasi

Pada tugas akhir ini, sistem aplikasi memiliki alur sebagai berikut :

1. Saat aplikasi dimulai, pengguna akan dihadapkan dengan dengan halaman menu utama.
2. Pengguna menekan tombol ambil gambar dan selanjutnya mengambil gambar denah yang ingin dikenali.
3. Setelah gambar diambil, pengguna menekan tombol submit untuk mengirimkan gambar ke server.
4. Pengguna bisa memindahkan denah, dan juga memutar posisi objek 3 dimensi dengan menggunakan gerakan tangan ke layar *smartphone*.

3.3.2 Aturan Aplikasi

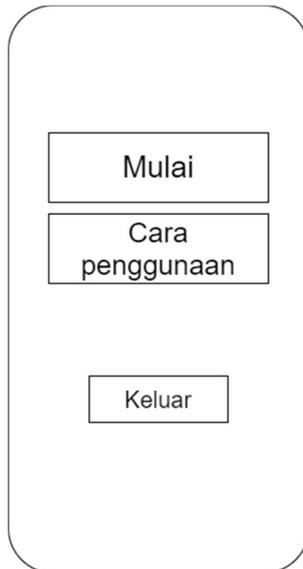
Dalam menggunakan aplikasi ini, terdapat beberapa aturan sebagai berikut :

1. Pengguna harus memastikan denah yang akan dikenali memenuhi batasan yang telah ditentukan.
2. Pengguna mengambil gambar denah dari atas dan sejajar dengan denah.
3. Apabila pengguna menekan tombol Submit tanpa mengambil gambar, maka aplikasi tidak akan mengirimkan data ke server dan hanya menampilkan peringatan.
4. Hasil model hanya bisa diakses melalui aplikasi dan tidak bisa dilihat di galeri *smartphone* ataupun di-*export* ke ekstensi lain.

3.4 Perancangan Tampilan Antarmuka

Pada sub bab ini membahas bagaimana rancangan antarmuka pengguna yang akan digunakan dalam tugas akhir. Rancangan antarmuka yang terdapat pada tugas akhir ini meliputi tampilan menu, tampilan pemilihan gambar, tampilan informasi pengguna, tampilan kamera AR, dan tampilan galeri.

3.4.1 Tampilan Menu



Gambar 3-5 Rancangan antarmuka halaman menu

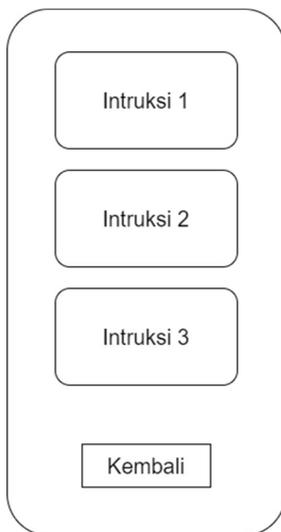
Tampilan menu merupakan tampilan awal yang pertama kali muncul saat membuka aplikasi. Pada tampilan ini terdapat 3 tombol yaitu tombol Ambil gambar, Submit, dan Cara penggunaan.

Tampilan dapat dilihat di Gambar 3.5 dan terlihat bahwa :

1. Tombol Mulai, berfungsi untuk memulai sistem.
2. Tombol Cara penggunaan, berfungsi untuk melihat cara penggunaan aplikasi.
3. Tombol Keluar, berfungsi untuk keluar dari aplikasi.

3.4.2 Tampilan Cara Penggunaan

Tampilan cara penggunaan merupakan halaman yang digunakan oleh pengguna untuk melihat bagaimana cara menggunakan aplikasi. Tampilan halaman cara penggunaan bisa dilihat di Gambar 3.6 dan disitu terdapat 3 instruksi berupa gambar dan sebuah tombol Kembali untuk kembali ke halaman utama.

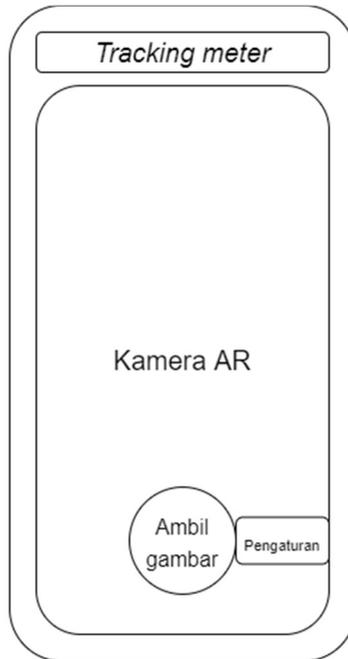


Gambar 3-6 Rancangan antarmuka halaman Pemilihan Gambar

3.4.3 Tampilan Kamera *Augmented Reality*

Tampilan Kamera *Augmented Reality* merupakan halaman yang digunakan oleh pengguna untuk melihat sistem *augmented reality* menampilkan model 3 dimensi. Tampilan halaman cara

penggunaan bisa dilihat di Gambar 3.7 dan disitu terdapat canvas yang menampilkan sistem *augmented reality*, sebuah teks di ujung kiri bawah yang menampilkan status tracking dan sebuah tombol Kembali untuk kembali ke halaman utama.



Gambar 3-7 Rancangan antarmuka halaman Kamera AR

BAB IV IMPLEMENTASI

Bab ini membahas implementasi dari perancangan sistem sesuai dengan perancangan yang telah dibuat. Sistem pengenalan denah yang digunakan dalam implementasi sistem aplikasi ini dibuat dengan PyTorch dan menggunakan bahasa pemrograman Python. Objek 3 dimensi yang digunakan dalam implementasi sistem aplikasi ini dibuat menggunakan three.js. Aplikasi *augmented reality* pada sistem ini dibuat menggunakan Unity 3D dan Vuforia SDK dan bahasa pemrograman yang digunakan untuk implementasi sistem adalah bahasa pemrograman C#.

4.1 Lingkungan Implementasi Perangkat Lunak

Lingkungan implementasi sistem yang digunakan untuk mengembangkan tugas akhir ini dibagi menjadi dua, yaitu perangkat komputer dan perangkat ponsel. Kedua perangkat tersebut memiliki spesifikasi perangkat keras dan perangkat lunak yang ditunjukkan oleh Tabel 4.1 dan Tabel 4.2.

Tabel 4-1 Spesifikasi perangkat komputer

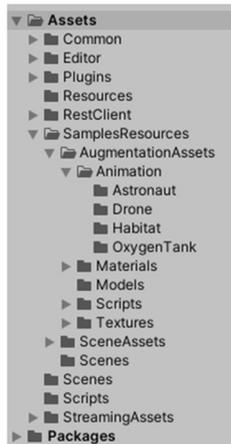
Perangkat	Spesifikasi
Perangkat Keras	<ul style="list-style-type: none">• Prosesor: Intel® Core™ i5-4300M CPU @ 2.60 GHz (4 CPUs), ~2.6GHz• Memori: 12288 MB
Perangkat Lunak	<ul style="list-style-type: none">• Sistem Operasi Microsoft Windows 10 64-bit• Perangkat Pengembang Visual Studio Code• Perangkat Pembantu Paint.net, draw.io, Microsoft Word 365, Snip & Sketch

Tabel 4-2 Spesifikasi perangkat *smartphone*

Perangkat	Spesifikasi
Perangkat Keras	<ul style="list-style-type: none">• Prosesor: Qualcomm Snapdragon 636 1.80 GHz• Memori: 2745B
Perangkat Lunak	<ul style="list-style-type: none">• Sistem Operasi Android 10

4.2 Implementasi *Import Asset dan Package*

Aplikasi AR *Floorplan Visualization* menggunakan teknologi *Augmented Reality*. Oleh karena itu, penulis menggunakan package dari Vuforia yang sudah mendukung *augmented reality*. Database dari Vuforia juga dibuat menjadi support dengan Unity Package sehingga mudah untuk dimasukkan ke dalam proyek pembangunan aplikasi ini. Selain itu, ada banyak asset yang dimasukkan ke dalam aplikasi seperti objek 3 dimensi, asset untuk antarmuka dan lainnya. Implementasi asept dan package dapat dilihat pada gambar 4.1

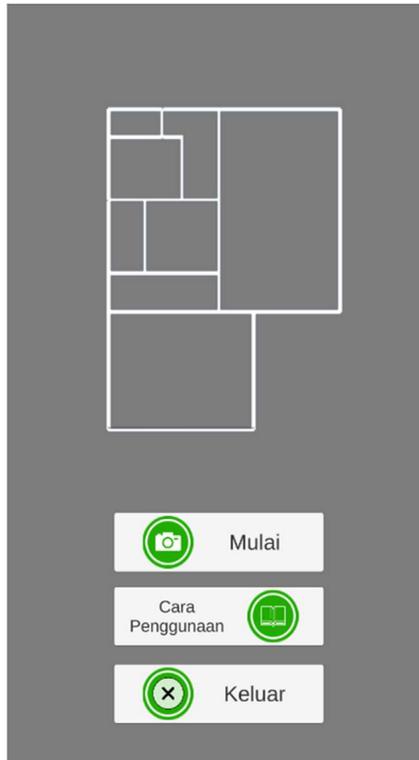


Gambar 4-1 Implementasi *import asset dan package*

4.3 Implementasi pada Aplikasi

Implementasi dari masing-masing fungsi utama dalam aplikasi pada tugas akhir ini dituliskan menggunakan *code* berbahasa *C#*. Implementasi fungsi diurutkan berdasarkan rancangan-rancangan tampilan pada sub bab 3.4.

4.3.1 Implementasi Halaman Menu



Gambar 4-2 Implementasi antarmuka Menu

Tampilan awal permainan dapat dilihat pada Gambar 4.2 terdapat tombol sebagai berikut :

1. Mulai untuk memulai aplikasi.
2. Cara penggunaan untuk melihat cara menggunakan aplikasi.
3. Keluar untuk keluar dari aplikasi.

Pada Kode Sumber 4.1, terdapat fungsi untuk menjalankan tombol-tombol pada halaman menu, diantaranya untuk memulai aplikasi dan untuk melihat galeri. Kode Sumber 4.2 digunakan untuk keluar dari aplikasi.

```
1. using System.Collections;
2. using System.Collections.Generic;
3. using UnityEngine;
4. using UnityEngine.SceneManagement
5.
6. public class MenuButton : MonoBehaviour {
7.     public void NewButtonMenu (string SceneName
8.     {
9.         SceneManager.LoadScene (SceneName);
10.    }
11. }
```

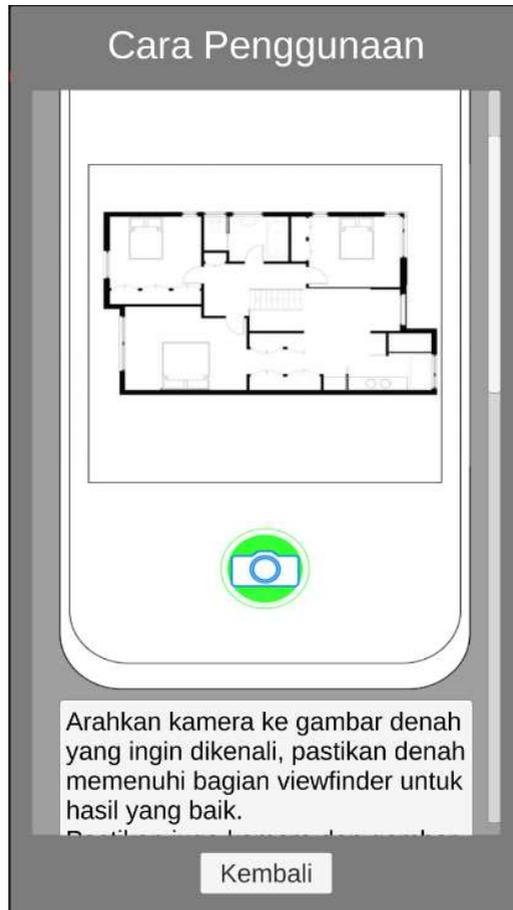
Kode Sumber 4.1 Fungsi tombol untuk pindah *scene*

```
1. using UnityEngine;
2. using System.Collections;
3. using System.Collections.Generic;
4.
5. public class ExitButton : MonoBehaviour {
6.
7.     public void menuexit()
8.
9.     {
10.         Application.Quit ();
11.    }
```

Kode Sumber 4.2 Fungsi untuk tombol keluar

4.3.2 Implementasi Halaman Cara Penggunaan

Implementasi halaman pilih gambar bisa dilihat pada Gambar 4.3. Pada halaman ini berisi 2 gambar instruksi dan satu tombol kembali. Implementasi tombol kembali yang digunakan sama dengan Kode Sumber 4.1 .



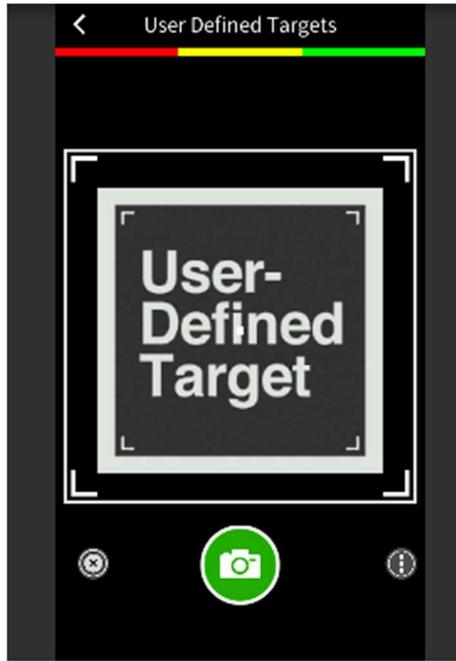
Gambar 4-3 Implementasi antarmuka Cara Penggunaan

4.3.3 Implementasi Kamera *Augmented Reality*

Pada implementasi halaman Kamera *Augmented Reality* adalah implementasi utama aplikasi AR *Floorplan Visualization* untuk tugas akhir ini. Pada halaman ini dilakukan proses pengambilan gambar melalui ARCamera Vuforia dan pencocokan gambar dengan database untuk memunculkan objek 3 dimensi. Pada halaman ini juga dilakukan proses pengambilan gambar yang kemudian akan dijadikan tekstur pada objek 3 dimensi yang dimunculkan. Selain itu, fitur untuk pengambilan gambar pada layar untuk disimpan ke dalam galeri juga dilakukan di halaman ini. Implementasi halaman Kamera *Augmented Reality* bisa dilihat pada Gambar 4.7. Terdapat 3 tombol pada halaman ini, yaitu :

Tombol Kembali untuk kembali ke halaman Pilih Gambar.

1. Tombol Informasi Pengguna untuk menampilkan dan menyembunyikan halaman Informasi Pengguna.
2. Tombol Kamera untuk mengambil gambar dan kemudian disimpan dan ditampilkan pada halaman Galeri.



Gambar 4-4 Implementasi antarmuka Kamera AR

Implementasi pada tombol kembali bisa dilihat pada Sumber 4.1.

Implementasi fungsi pengambilan gambar sebagai tekstur objek 3 dimensi dapat dilihat pada Sumber 4-3 dibawah ini.

```
1. IEnumerator CaptureTextureAsJPG()  
2. {  
3.     commonCanvas.SetActive(false);  
4.     builderCanvas.SetActive(false);  
5.  
6.     yield return new WaitForEndOfFrame();  
7. }
```

```

8. Texture2D _TextureFromCamera = new Texture2D(detectorCamera.targetTexture.width, detectorCamera.targetTexture.height, TextureFormat.RGB24, false);
9. _TextureFromCamera.ReadPixels(new Rect(0, 0, detectorCamera.targetTexture.width, detectorCamera.targetTexture.height), 0, 0);
10. _TextureFromCamera.Apply();
11.
12. ImageCropper.Instance.Show(_TextureFromCamera, (bool result, Texture2D originalImage, Texture2D croppedImage) =>
13. {
14.     commonCanvas.SetActive(true);
15.     builderCanvas.SetActive(true);
16.
17.     // If screenshot was cropped successfully
18.     if (result)
19.     {
20.         StartCoroutine(SendImage(croppedImage));
21.     }
22.     else
23.     {
24.
25.     }
26. },
27. settings: new ImageCropper.Settings()
28. {
29.     ovalSelection = false,
30.     autoZoomEnabled = false,
31.     imageBackground = Color.clear, // transparent background
32.     selectionMinAspectRatio = minAspectRatio,
33.     selectionMaxAspectRatio = maxAspectRatio,
34.     markTextureNonReadable = false
35.
36. });
37. }

```

Kode Sumber 4.3 Fungsi pengambilan tekstur melalui kamera

Implementasi pengiriman gambar yang sedang ditampilkan pada layar *smartphone* dapat dilihat pada Sumber 4.4 dibawah ini.

```

1. private IEnumerator SendImage(Texture2D toBeSent)
2. {
3.     byte[] imageBytes = toBeSent.EncodeToJPG();
4.     string filePath = "SavedScreen1.jpg";
5.
6.     commonCanvas.SetActive(true);
7.     builderCanvas.SetActive(true);
8.
9.     WWWForm formData = new WWWForm();
10.    formData.AddBinaryData("file", imageBytes, "image.jpg");
11.
12.    currentRequest = new RequestHelper
13.    {
14.        Uri = basePath + "/send",
15.        FormData = formData,
16.        EnableDebug = false,
17.        ParseResponseBody = true
18.    };
19.    RestClient.Post(currentRequest)
20.    .Then(res =>
21.    {
22.        RestClient.ClearDefaultParams();
23.
24.        string[] wallCoordinates = res.Text.ToString().Split(' ');
25.        foreach (string coordinates in wallCoordinates)
26.        {
27.            string[] points = coordinates.Split('|');
28.            newFloorplan.Add(new Wall(new Vector2(float.Parse(points[
0]) / 10, float.Parse(points[1]) / 10), new Vector2(float.Parse(points[2
]) / 10, float.Parse(points[3]) / 10)));
29.        }
30.    })
31.    .Catch(err => this.LogMessage("Error", err.Message));
32.
33.    File.WriteAllBytes(filePath, imageBytes);
34.
35.    yield return null;
36. }

```

Kode Sumber 4.4 Fungsi pengiriman gambar pada layar

Implementasi pembuatan model berdasarkan data yang dikirim oleh server dan ditampilkan pada layar *smartphone* dapat dilihat pada Sumber 4.5 dibawah ini.

```
1. private List<Wall> wallPlan = new List<Wall>();
2.
3. public void BuildWall()
4. {
5.     float top = Mathf.NegativeInfinity;
6.     float bottom = Mathf.Infinity;
7.     float right = Mathf.NegativeInfinity;
8.     float left = Mathf.Infinity;
9.     for (int i = 0; i < wallPlan.Count; i++)
10.    {
11.        walls[i].transform.localPosition = new Vector3((wallPlan[i].Coordinates[0].x + wallPlan[i].Coordinates[1].x)/2, 0, -(wallPlan[i].Coordinates[0].y + wallPlan[i].Coordinates[1].y) / 2);
12.        walls[i].transform.localScale = new Vector3(Mathf.Abs(wallPlan[i].Coordinates[1].x - wallPlan[i].Coordinates[0].x) + 0.1f, 1, Mathf.Abs(wallPlan[i].Coordinates[1].y - wallPlan[i].Coordinates[0].y) + 0.1f);
13.        walls[i].SetActive(true);
14.        if (walls[i].transform.localPosition.x > right) right = walls[i].transform.localPosition.x;
15.        if (walls[i].transform.localPosition.x < left) left = walls[i].transform.localPosition.x;
16.
17.        if (walls[i].transform.localPosition.z > top) top = walls[i].transform.localPosition.z;
18.        if (walls[i].transform.localPosition.z < bottom) bottom = walls[i].transform.localPosition.z;
19.    }
20.    gameObject.transform.position = new Vector3((gameObject.transform.position.x - ((right + left)/2))/40, 0, (gameObject.transform.position.z - ((top + bottom) / 2))/40);
21. }
```

Kode Sumber 4.5 Fungsi pembuatan model 3 dimensi

BAB V UJI COBA DAN EVALUASI

Pada bab ini akan dijelaskan mengenai rangkaian uji coba dan evaluasi yang dilakukan pada aplikasi AR *Floorplan Visualization*.

5.1 Lingkungan Uji Coba

Pada proses pengujian perangkat lunak, dibutuhkan suatu lingkungan pengujian yang sesuai dengan standar kebutuhan. Lingkungan pengujian sistem pada pengerjaan Tugas Akhir disesuaikan dengan kebutuhan dari aplikasi. Spesifikasi lingkungan pengujian dijabarkan pada Tabel 5.1.

Tabel 5-1 Spesifikasi Lingkungan Uji Coba

Perangkat	Spesifikasi
Perangkat Keras	✓ <i>Smartphone</i> Asus Max M1 Pro
	✓ Qualcomm SDM636 Snapdragon 636 (14 nm)
	✓ RAM 3 GB
	✓ IPS LCD capacitive touchscreen, 16M colors
	✓ 13 MP, f/2.2, 25mm (wide), 1.12µm, PDAF ✓ 5 MP, f/2.4, (depth)
Perangkat Lunak	✓ Sistem operasi Android 10

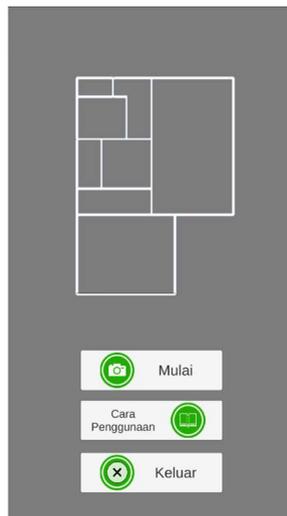
5.2 Pengujian Fungsionalitas

Pengujian fungsionalitas sistem dilakukan dengan menyiapkan sejumlah skenario sebagai tolok ukur keberhasilan pengujian. Pengujian fungsionalitas yang terdapat pada aplikasi dijabarkan sebagai berikut:

5.2.1 Uji Coba Halaman Awal

Pada sub bab ini dijelaskan secara detil mengenai skenario yang dilakukan dan hasil yang didapatkan dari pengujian fungsionalitas perangkat lunak yang dibangun pada halaman awal.

Pada halaman awal yang akan diuji adalah fungsionalitas tombol yang terdapat di halaman awal, yaitu tombol Mulai, Galeri, dan tombol Keluar. Tampilan halaman awal adalah dapat dilihat pada Gambar 5.1.



Gambar 5-1 Implementasi antarmuka Menu

Hasil uji coba pada halaman menu dapat dilihat pada Tabel 5.2.

Tabel 5-2 Hasil uji coba halaman Menu

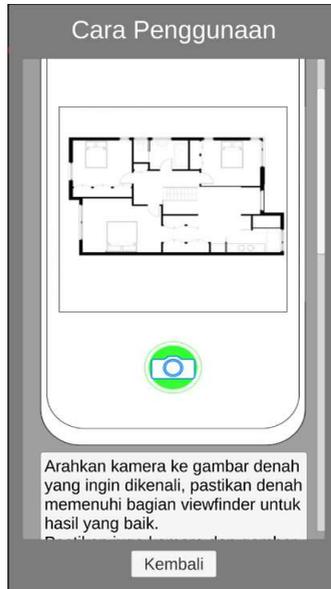
ID	UF-001
Nama	Uji Coba Pada Halaman Menu
Tujuan uji coba	Pengguna mengetahui fungsionalitas tombol yang ada pada halaman awal aplikasi
Kondisi awal	Pengguna berada pada halaman awal permainan
<i>Skenario 1</i>	<i>Pengguna memilih tombol Mulai</i>
Masukan	Klik tombol Mulai pada layar
Keluaran yang diharapkan	Pengguna berpindah ke halaman Kamera AR
Hasil uji coba	Berhasil
Kondisi Akhir	Pengguna berada pada halaman Kamera AR
<i>Skenario 2</i>	<i>Pengguna memilih tombol Cara Penggunaan</i>
Masukan	Klik tombol Cara Penggunaan pada layar
Keluaran yang diharapkan	Pengguna berpindah ke Halaman Cara Penggunaan
Hasil uji coba	Berhasil
Kondisi Akhir	Pengguna berpindah ke Halaman Cara Penggunaan

ID	UF-001
<i>Skenario 3</i>	<i>Pengguna memilih tombol Keluar</i>
Masukan	Klik tombol Keluar pada layar
Keluaran yang diharapkan	Pengguna berpindah keluar dari aplikasi
Hasil uji coba	Berhasil
Kondisi Akhir	Pengguna berpindah keluar dari aplikasi

5.2.2 Uji Coba Halaman Cara Penggunaan

Pada sub bab ini dijelaskan secara detil mengenai skenario yang dilakukan dan hasil yang didapatkan dari pengujian fungsionalitas perangkat lunak yang dibangun pada halaman Cara Penggunaan. Penjelasan disajikan dengan menampilkan kondisi awal, masukan, keluaran, hasil yang dicapai, dan kondisi akhir.

Pada halaman Pemilihan gambar yang akan diuji adalah fungsionalitas tombol yang terdapat di halaman ini, yaitu tombol kembali dan juga gambar. Tampilan informasi dapat dilihat pada Gambar 5.2.



Gambar 5-2 Implementasi antarmuka Cara Penggunaan

Tabel 5-3 Hasil uji coba halaman Pemilihan Gambar

ID	UF-002
Nama	Uji Coba Pada Halaman Cara Penggunaan
Tujuan uji coba	Pengguna mengetahui fungsionalitas tombol yang ada pada halaman Cara Penggunaan
Kondisi awal	Pengguna berada pada halaman Cara Penggunaan
<i>Skenario 1</i>	<i>Pengguna memilih tombol level Kembali</i>

ID	UF-002
Masukan	Klik tombol kembali pada layar
Keluaran yang diharapkan	Pengguna berpindah ke halaman Menu utama
Hasil uji coba	Berhasil
Kondisi Akhir	Pengguna berada pada halaman Menu Utama

5.2.3 Uji Coba Halaman Kamera *Augmented Reality*

Pada sub bab ini dijelaskan mengenai skenario yang dilakukan dan hasil yang didapatkan dari pengujian fungsionalitas halaman Kamera *Augmented Reality*. Halaman ini berisi halaman utama dari aplikasi. Terdapat 3 tombol yang bisa dilihat pada Gambar 5.4, yaitu tombol Kembali dan Tombol Kamera.



Gambar 5-3 Implementasi antarmuka Kamera AR

Hasil pengujian dapat dilihat pada Tabel 5.5

Tabel 5-4 Hasil uji coba halaman Kamera AR

ID	UF-003
Nama	Uji Coba Pada Halaman Kamera <i>Augmented Reality</i>
Tujuan uji coba	Pengguna mengetahui fungsionalitas tombol dan interaksi yang ada pada halaman Kamera <i>Augmented Reality</i>
Kondisi awal	Pengguna berada pada halaman Kamera <i>Augmented Reality</i> dan layar menampilkan gambar dari kamera
<i>Skenario 1</i>	<i>Pengguna memilih tombol Kamera</i>
Masukan	Klik tombol Kamera pada layar
Keluaran yang diharapkan	Gambar pada layar berhasil diambil dan dikirim ke server
Hasil uji coba	Berhasil
Kondisi Akhir	Gambar pada layar berhasil diambil dan model 3 dimensi digambar diatas gambar denah
<i>Skenario 2</i>	<i>Pengguna memilih tombol Kembali</i>
Masukan	Klik tombol level kembali pada layar
Keluaran yang diharapkan	Pengguna berpindah ke halaman Menu utama

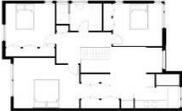
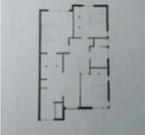
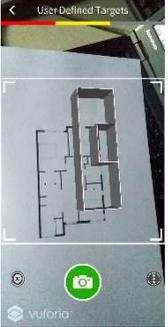
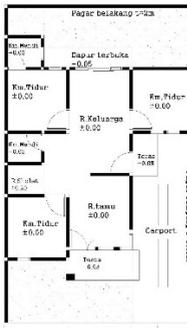
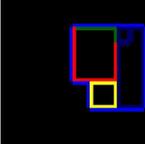
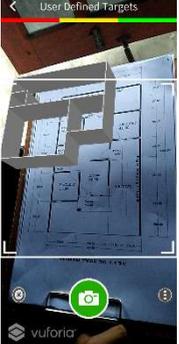
ID	UF-003
Hasil uji coba	Berhasil
Kondisi Akhir	Pengguna berpindah ke halaman Menu utama

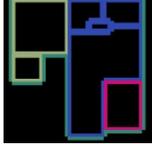
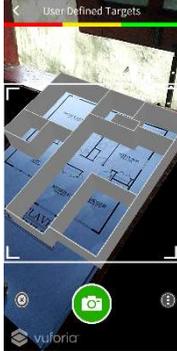
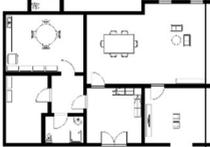
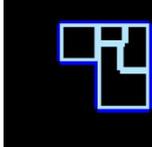
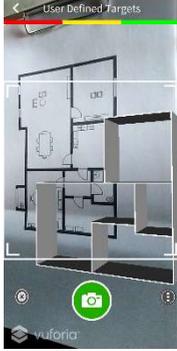
5.2.4 Uji Coba Pengenalan Denah

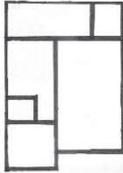
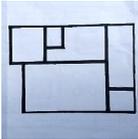
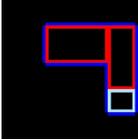
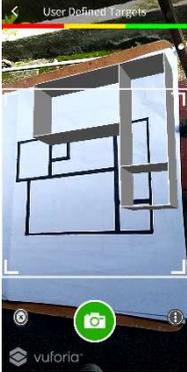
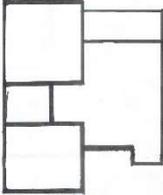
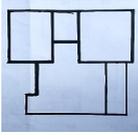
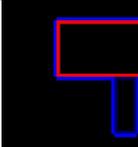
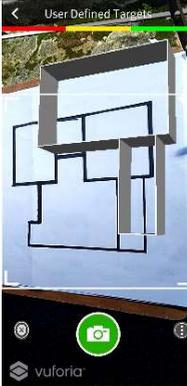
Pada sub bab ini dilakukan pengujian terhadap gambar denah untuk mengetahui seberapa keberhasilan aplikasi membuat objek 3 dimensi sesuai dengan gambar yang diberikan.

Uji coba dilakukan dengan mengambil 4 gambar denah dari Google Images secara acak dan membuat 2 gambar denah di kertas. Hasil akurasi dihitung berdasarkan jumlah dinding yang berhasil dibuat oleh model dan berada pada tempat yang sesuai pada lokasi dinding yang ada di citra denah dibagi dengan jumlah dinding pada gambar asli. Satu dinding didefinisikan dengan garis antara dua sudut atau persimpangan, sehingga apabila ada satu garis yang melewati lebih dari dua persimpangan, maka garis tersebut akan dibagi berdasarkan jumlah persimpangan yang dilewati. Hasil Uji coba bisa dilihat pada Tabel 5.5.

Tabel 5-5 Hasil uji coba pengenalan citra denah

Gambar denah	Raster masukan	Vektor keluaran	Hasil akhir
 <p>Jumlah dinding: 22</p>			 <p>Jumlah dinding: 9 Akurasi: 40%</p>
 <p>Jumlah dinding: 34</p>			 <p>Jumlah dinding: 15 Akurasi: 44%</p>

Gambar denah	Raster masukan	Vektor keluaran	Hasil akhir
 <p data-bbox="213 520 351 587">Jumlah dinding: 36</p>			 <p data-bbox="762 667 958 735">Jumlah dinding: 28</p> <p data-bbox="762 762 930 794">Akurasi: 77%</p>
 <p data-bbox="213 999 351 1066">Jumlah dinding: 28</p>			 <p data-bbox="762 1201 958 1270">Jumlah dinding: 15</p> <p data-bbox="762 1297 930 1329">Akurasi: 53%</p>

Gambar denah	Raster masukan	Vektor keluaran	Hasil akhir
 <p data-bbox="210 533 348 600">Jumlah dinding: 21</p>			 <p data-bbox="766 687 956 754">Jumlah dinding: 11</p> <p data-bbox="766 783 930 815">Akurasi: 52%</p>
 <p data-bbox="210 1091 351 1158">Jumlah dinding: 22</p>			 <p data-bbox="766 1251 956 1318">Jumlah dinding: 9</p> <p data-bbox="766 1347 930 1378">Akurasi: 41%</p>

Berdasarkan hasil evaluasi pada tabel 5.5 pengenalan dapat berjalan lancar dengan akurasi rata-rata 51%, namun saat pengujian, ada 3 dari 6 percobaan, akurasi berada dibawah 50%, hal ini dapat terjadi karena beberapa faktor yaitu:

1. Gambar raster yang menjadi masukan kurang detail, sehingga algoritma kesulitan melakukan pengenalan.
2. Cahaya yang diterima gambar sedikit dan menjadikan gambar terlalu gelap dan sulit dipindai oleh kamera.
3. Rating feature pada gambar terlalu sedikit.

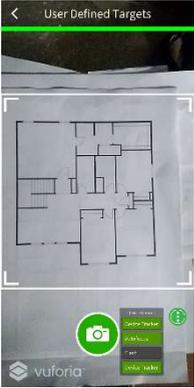
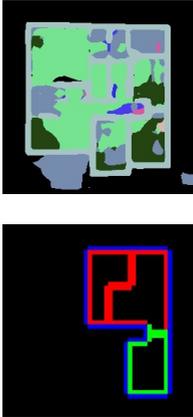
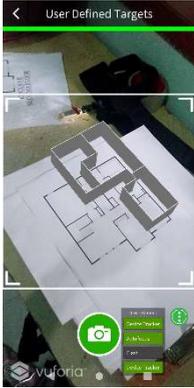
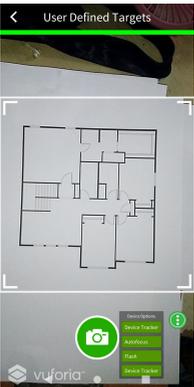
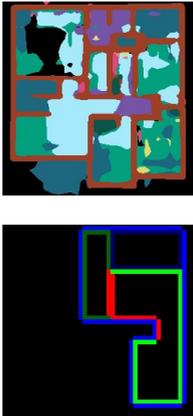
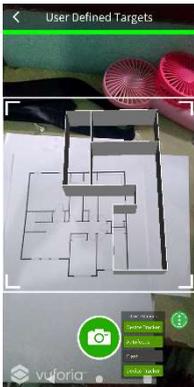
5.2.5 Uji Coba Akurasi Pengenalan Denah

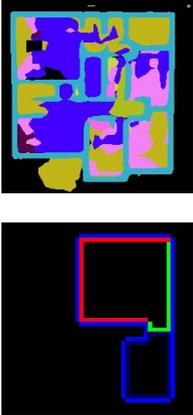
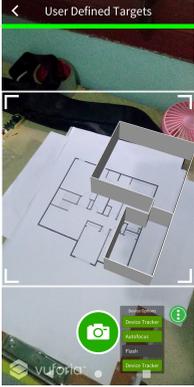
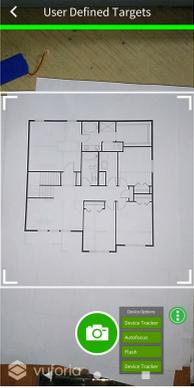
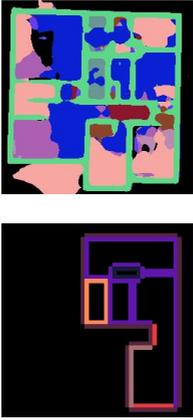
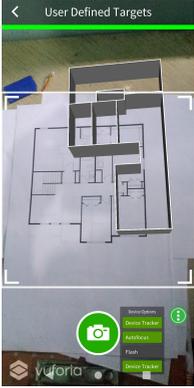
Pada sub bab ini dilakukan pengujian terhadap gambar denah untuk mengetahui tingkat akurasi sistem aplikasi membuat objek 3 dimensi sesuai dengan gambar yang diberikan. Uji coba ini dilakukan untuk menilai apakah detail teks dan detail furnitur mempengaruhi akurasi algoritma dalam mengenali citra denah.

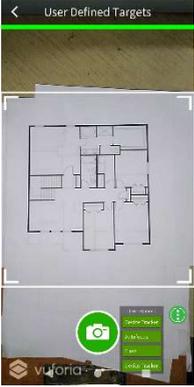
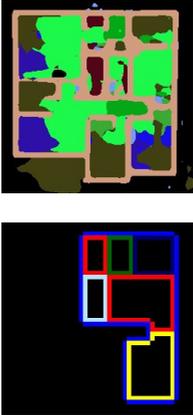
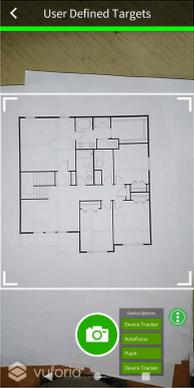
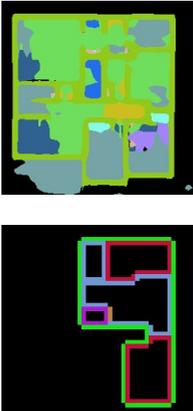
Uji coba dilakukan dengan mengambil satu gambar citra denah dan kemudian disunting menjadi 4 gambar. Satu gambar berupa citra denah tanpa adanya detail teks dan furnitur, satu gambar berupa citra denah yang hanya memiliki detail teks, satu gambar berupa citra denah yang hanya memiliki detail furnitur, dan satu gambar lagi berupa citra denah asli yang memiliki detail teks dan furnitur. Hasil akurasi mengikuti perhitungan yang telah dijelaskan pada subbab 5.2.4, dimana jumlah dinding yang berhasil dibuat oleh model dan berada pada tempat yang sesuai pada lokasi dinding yang ada di citra denah dibagi dengan jumlah dinding pada gambar asli kemudian diubah menjadi persentase. Untuk bisa mendapatkan nilai akurasi yang lebih tepat, maka pengujian tiap gambar dilakukan tiga kali, dan rata-rata dari setiap pengujian itulah yang ditetapkan sebagai nilai akurasi yang didapat oleh aplikasi terhadap gambar tersebut. Hasil Uji coba

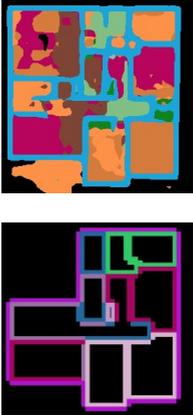
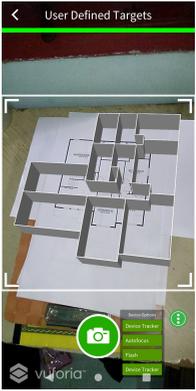
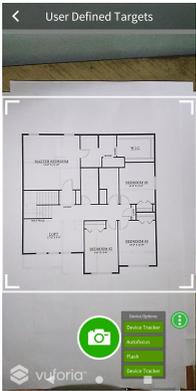
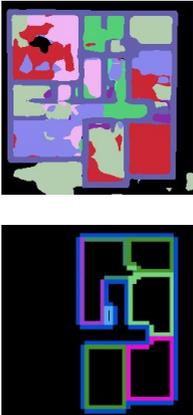
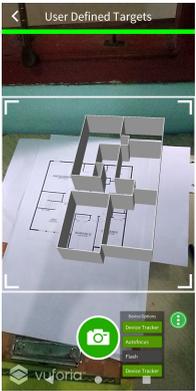
bisa dilihat pada Tabel 5.6 dan hasil penghitungan akurasi bisa dilihat pada Tabel 5.7.

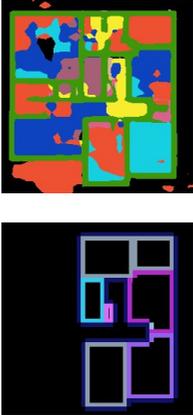
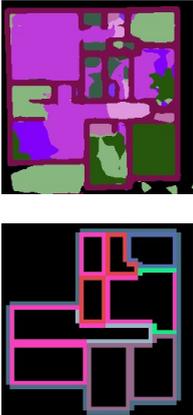
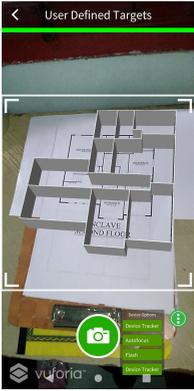
Tabel 5-6 Hasil uji coba akurasi pengenalan citra denah

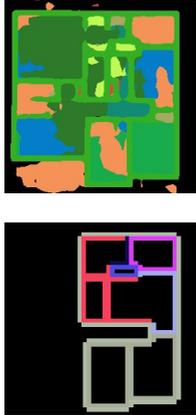
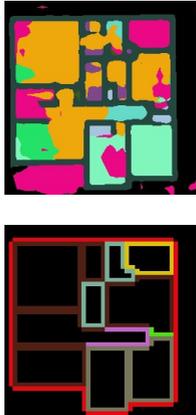
Input dari Unity	Hasil Pengenalan	Hasil Akhir
Skenario 1: Citra denah tidak memiliki detail teks dan furnitur		
		
		

Input dari Unity	Hasil Pengenalan	Hasil Akhir
		
<p>Skenario 2: Citra denah tidak memiliki detail teks, citra denah memiliki detail furnitur</p>		
		

Input dari Unity	Hasil Pengenalan	Hasil Akhir
		
		
<p>Skenario 3: Citra denah memiliki detail teks, citra denah tidak memiliki detail furnitur</p>		

Input dari Unity	Hasil Pengenalan	Hasil Akhir
		
		

Input dari Unity	Hasil Pengenalan	Hasil Akhir
		
<p>Skenario 4: Citra denah memiliki detail teks, citra denah memiliki detail furnitur</p>		
		

Input dari Unity	Hasil Pengenalan	Hasil Akhir
		
		

Tabel 5-7 Hasil pengukuran akurasi pada hasil akhir pengenalan citra denah

	Jumlah dinding		Nilai akurasi
	Citra denah	Hasil akhir	
Skenario 1: Citra denah tidak memiliki detail teks dan furnitur			
Citra pertama	36	17	47%
Citra kedua	36	18	50%
Citra ketiga	36	16	44%
Akurasi rata-rata			47%
Skenario 2: Citra denah memiliki detail furnitur, tetapi tidak memiliki detail teks			
Citra pertama	36	22	61%
Citra kedua	36	19	53%
Citra ketiga	36	19	53%
Akurasi rata-rata			56%
Skenario 3: Citra denah memiliki detail teks, tetapi tidak memiliki detail furnitur			
Citra pertama	36	28	78%
Citra kedua	36	24	67%
Citra ketiga	36	23	64%
Akurasi rata-rata			70%
Skenario 4: Citra denah memiliki detail teks dan furnitur			
Citra pertama	36	28	78%
Citra kedua	36	23	64%
Citra ketiga	36	28	78%
Akurasi rata-rata			73%

Berdasarkan hasil pengukuran pada table 5.7, algoritma pengenalan memiliki akurasi rata-rata 73% pada citra denah yang memiliki detail teks dan furnitur, dan akurasi rata-rata 47% pada citra denah yang tidak memiliki detail teks atau furnitur. Akurasi meningkat pada denah yang hanya memiliki citra detail furnitur dengan nilai akurasi rata-rata 56% dan pada citra denah yang hanya memiliki detail teks dengan nilai akurasi rata-rata 70%.

Algoritma pengenalan mendapatkan peningkatan pada akurasi rata-rata pada citra denah yang memiliki detail furnitur sebanyak 9% dibandingkan dengan denah yang tidak memiliki detail sama sekali. Algoritma pengenalan juga mendapatkan peningkatan akurasi rata-rata sebanyak 23% dibandingkan dengan denah yang tidak memiliki detail sama sekali.

5.2.6 Pengujian Pengguna

Pengujian pada perangkat lunak yang dibangun tidak hanya dilakukan pada fungsionalitas yang dimiliki, tetapi juga ditujukan kepada pengguna untuk melihat aplikasi bekerja. Pengujian ini berfungsi sebagai pengujian subjektif yang bertujuan untuk mengetahui tingkat keberhasilan aplikasi yang dibangun dari sisi pengguna. Hal ini dapat dicapai dengan meminta penilaian dan tanggapan dari pengguna terhadap sejumlah aspek perangkat lunak yang ada.

5.2.7 Skenario Pengujian Pengguna

Dalam melakukan pengujian perangkat lunak, pengguna diminta mencoba menggunakan perangkat lunak untuk mencoba semua fungsionalitas dan fitur yang ada. Pengujian aplikasi oleh pengguna dilakukan dengan sebelumnya memberikan informasi seputar aplikasi, kegunaan, dan fitur-fitur yang dimiliki. Setelah informasi tersampaikan, pengguna kemudian diarahkan untuk menonton sebuah video yang memperlihatkan aplikasi bekerja

dengan spesifikasi lingkungan yang sama dengan yang telah diuraikan pada Lingkungan Uji Coba.

Jumlah pengguna yang terlibat dalam pengujian perangkat lunak sebanyak 7 orang. Dalam pengujian pengguna diberi pertanyaan tentang perangkat lunak ini memiliki beberapa aspek penilaian seputar desain antarmuka, *immersivity*, dan tingkat kenyamanan aplikasi. Nilai yang diberikan memiliki rentang nilai 1 hingga 4 dengan rincian pada Tabel 5.6. Pada bagian akhir terdapat saran untuk perbaikan fitur. Detail pertanyaan kepada pengguna dapat dilihat pada Tabel 5.7.

Tabel 5-8 Rentang nilai

No	Keterangan	Nilai
1	Tidak Setuju (TS)	1
2	Cukup Setuju (CS)	2
3	Setuju (S)	3
4	Sangat Setuju (SS)	4

Tabel 5-9 Kuesioner Pengguna

No	Pernyataan	TS	CS	S	SS
1	Antarmuka aplikasi mudah dimengerti				
2	Aplikasi memiliki tata letak tombol yang mudah dilihat/dikenali				
3	Aplikasi dapat mengenali denah dengan akurat				
4	Saya merasa akan terbantu dengan adanya aplikasi ini				
5	Saya merasa tertarik untuk menggunakan aplikasi ini				

5.2.8 Daftar Penguji Perangkat Lunak

Pada sub bab ini ditunjukkan daftar pengguna yang bertindak sebagai penguji coba aplikasi yang dibangun. Dalam pengujian ini, pengguna sudah ditentukan yaitu mahasiswa ITS

dari departemen Arsitektur dan Teknik Sipil. Daftar nama pengujian aplikasi ini dapat dilihat pada Tabel 5.8.

Tabel 5-10 Daftar Pengujian Perangkat Lunak

No	Nama	Profesi	Institusi
1	Raihan Erviansyah Ekaputra	Mahasiswa, Game Developer	ITS, Crescent Moon Games
2	Naufal Fadhil		
3	Brigita L	Mahasiswa	ITS
4	Farhan Firmansyah	Mahasiswa	ITS
5	Alya Putri Nabila	Mahasiswa Arsitektur	ITS
6	Tiara Shafira	Mahasiswa	ITS
7	Widya Indriyani M	Mahasiswa	ITS

5.2.9 Hasil Pengujian Pengguna

Uji coba yang dilakukan terhadap beberapa pengguna memiliki beberapa aspek yang dipisahkan berdasarkan antarmuka, *immersivity*, dan tingkat kenyamanan. Sistem penilaian didasarkan pada skala penghitungan satu sampai empat dimana skala satu menunjukkan nilai terendah dan skala empat menunjukkan skala tertinggi. Penilaian akhir kemudian dilakukan dengan menghitung berapa banyak pengujian yang memilih suatu skala tertentu dan kemudian dicari nilai rata-ratanya. Hasil uji coba dipaparkan secara lengkap dengan disertai tabel yang dapat dilihat pada Tabel 5.9.

Tabel 5-11 Hasil Akhir Pengujian Pengguna

No	Pernyataan	Penilaian				Rata-Rata
		1	2	3	4	

1	Antarmuka aplikasi mudah dimengerti	0	3	2	2	2,9
2	Aplikasi memiliki tata letak tombol yang mudah dilihat/dikenali	0	1	5	1	3,0
3	Aplikasi dapat mengenali denah dengan akurat	2	2	2	1	2,3
4	Saya merasa akan terbantu dengan adanya aplikasi ini	0	1	3	3	3,3
5	Saya merasa tertarik untuk menggunakan aplikasi ini	0	1	4	2	3,1

5.2.10 Kritik dan Saran Pengguna

Dalam memberikan penilaian dan tanggapan, pengguna diberikan kuesioner pengujian perangkat lunak. Kuesioner pengujian perangkat lunak ini terdapat bagian kritik dan saran untuk perbaikan fitur kedepannya. Kritik dan saran pengguna dapat dilihat pada Tabel 5.10.

Tabel 5-12 Kritik dan Saran Pengguna

No	Nama	Kritik dan Saran
1	Raihan Erviansyah Ekaputra	Menggunakan ARCore atau ARKit untuk stabilitas dan presisi <i>augmented reality</i> , menggunakan <i>visual cues</i> yang ditambahkan ke denah sehingga lebih akurat, desain ulang UI sehingga lebih menarik, dan menambahkan interaktivitas pengguna dengan model 3 dimensi.
2	Farhan Firmansyah	Desain UI pada aplikasi bisa lebih bagus dan tingkat akurasi semakin dipertajam.
3	Alya Putri Nabila	Meningkatkan akurasi Image Recognition dan menambahkan

		furnitur untuk menarik lebih banyak pengguna.
4	Widya Indriyani M	Memberikan penjelasan mengenai batasan denah yang bisa dipakai di dalam aplikasi.

5.3 Evaluasi Pengujian

Sub bab ini membahas mengenai evaluasi terhadap pengujian-pengujian yang telah dilakukan. Dalam hal ini, evaluasi menunjukkan data rekapitulasi dari hasil pengujian fungsionalitas. Rekapitulasi disusun dalam bentuk tabel yang dapat dilihat pada Tabel 5.5. Dari data yang terdapat pada tabel tersebut, diketahui bahwa aplikasi yang dibuat telah berjalan sesuai dengan skenario yang diharapkan pada awal perancangan dan mencapai akurasi rata-rata 51% . Hasil kuesioner pada Tabel 5.10 menyatakan bahwa aplikasi sudah memiliki antarmuka yang baik, dan masih memiliki tingkat akurasi yang bisa diterima. Kebanyakan pengguna menyarankan untuk meningkatkan akurasi algoritma pengenalan denah, dan menambah desain antarmuka sehingga menjadi lebih menarik.

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini akan diberikan kesimpulan yang diambil selama pengerjaan Tugas Akhir serta saran-saran tentang pengembangan yang dapat dilakukan terhadap Tugas Akhir ini di masa yang akan datang.

6.1 Kesimpulan

Dari hasil selama proses perancangan, implementasi, serta pengujian dapat diambil kesimpulan sebagai berikut:

1. Sistem aplikasi dapat mengenali titik sudut dari suatu citra denah menggunakan algoritma *raster-to-vector floorplan transformation*, yang memanfaatkan algoritma *deep network* dan *integer programming*.
2. Sistem aplikasi berhasil membangun model 3 dimensi berdasarkan keluaran dalam bentuk titik-titik vektor dari algoritma *raster-to-vector floorplan transformation* dan menampilkan model 3 dimensi yang selesai dibangun dengan Vuforia.
3. Sistem aplikasi dapat mengenali citra denah yang diberikan dengan akurasi rata-rata 51%. Tampilan antar muka aplikasi sudah cukup bagus dan aplikasi mudah digunakan. Akurasi pengenalan denah juga dinilai cukup baik oleh sebagian besar pengguna.

6.2 Saran

Berikut saran-saran untuk pengembangan dan perbaikan sistem di masa yang akan datang. Diantaranya adalah sebagai berikut:

1. Menambahkan furnitur dan detail ke denah yang sudah pernah dibuat model 3 dimensinya.
2. Memastikan gambar denah tidak mengalami distorsi akibat posisi kamera yang tidak sejajar dengan denah.

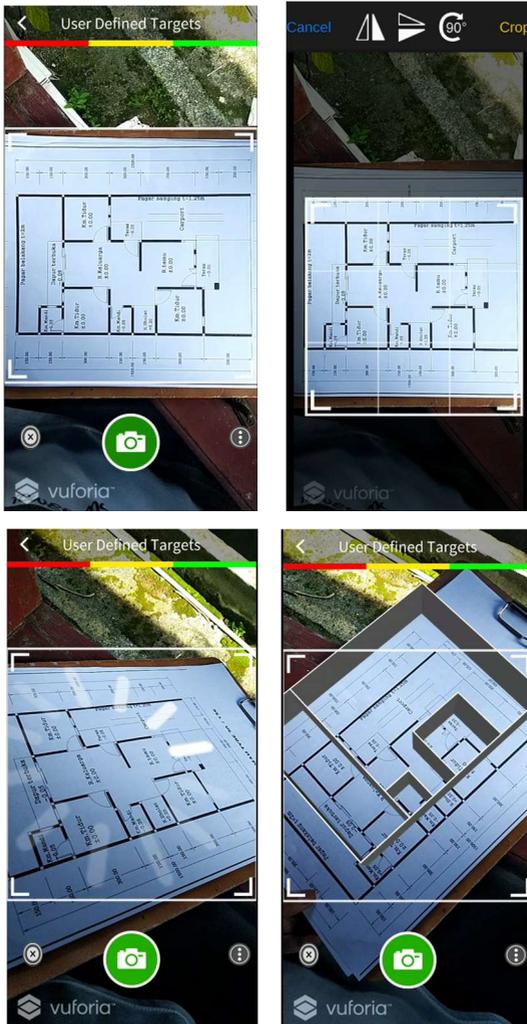
3. Meningkatkan akurasi *tracking* untuk *augmented reality* dengan menyematkan *visual cues* pada denah dan menggunakan ARCore dan ARKit.

DAFTAR PUSTAKA

- [1] C. Liu, J. Wu, P. Kohli, and Y. Furukawa, “Raster-to-Vector: Revisiting Floorplan Transformation,” in *Proceedings of the IEEE International Conference on Computer Vision*, Dec. 2017, vol. 2017-Octob, pp. 2214–2222, doi: 10.1109/ICCV.2017.241.
- [2] “Documentation for Visual Studio Code.” <https://code.visualstudio.com/docs> (accessed Jun. 22, 2020).
- [3] “What is Python? Executive Summary | Python.org.” <https://www.python.org/doc/essays/blurb/> (accessed Jun. 23, 2020).
- [4] “Unity - Manual: Unity User Manual (2019.4 LTS).” <https://docs.unity3d.com/Manual/UnityManual.html> (accessed Jun. 25, 2020).
- [5] “Flask | The Pallets Projects.” <https://palletsprojects.com/p/flask/> (accessed Jun. 23, 2020).
- [6] “Vuforia Developer Portal |.” <https://developer.vuforia.com/> (accessed Jun. 25, 2020).

LAMPIRAN

Uji coba



Gambar 0-1 Pengambilan gambar denah

Hasil kuesioner

Widya Indriyani M Widya mahasiswa

Submission Date
July 1, 2020 19:14

Nama	Widya Indriyani M Widya
Profesi	mahasiswa
Institusi (jika ada)	pelajar
Antarmuka aplikasi mudah dimengerti	Cukup setuju
Aplikasi memiliki tata letak tombol yang mudah dilihat/dikenali	Setuju
Aplikasi dapat mengenali denah dengan akurat	Cukup setuju
Saya merasa akan terbantu dengan adanya aplikasi ini	Setuju
Saya merasa tertarik untuk menggunakan aplikasi ini	Setuju
Saran:	apakah contoh modelnya bisa dari aplikasi autocad? Apa harus dari kertas? Mungkin bisa dijelaskan disitu

Gambar 0-4 Kuesioner Responden Pertama

Tiara Shafira Mahasiswa

Submission Date
July 2, 2020 08:42

Nama	Tiara Shafira
Profesi	Mahasiswa
Institusi (jika ada)	ITS
Antarmuka aplikasi mudah dimengerti	Sangat setuju
Aplikasi memiliki tata letak tombol yang mudah dilihat/dikenali	Sangat setuju
Aplikasi dapat mengenali denah dengan akurat	Sangat setuju
Saya merasa akan terbantu dengan adanya aplikasi ini	Sangat setuju
Saya merasa tertarik untuk menggunakan aplikasi ini	Sangat setuju

Gambar 0-5 Kuesioner Responden Kedua

Alya Putri Nabila

Mahasiswa Arsitektur

Submission Date
July 2, 2020 08:48

Nama	Alya Putri Nabila
Profesi	Mahasiswa Arsitektur
Institusi (jika ada)	ITS
Antarmuka aplikasi mudah dimengerti	Setuju
Aplikasi memiliki tata letak tombol yang mudah dilihat/dikenali	Setuju
Aplikasi dapat mengenali denah dengan akurat	Setuju
Saya merasa akan terbantu dengan adanya aplikasi ini	Sangat setuju
Saya merasa tertarik untuk menggunakan aplikasi ini	Setuju
Saran:	sarannya adalah image recognitionnya ditingkatkan akurasi, penambahan furniture dalam aplikasi bisa semakin menarik user karena all-in-one application ketika ingin melihat bagaimana bangunan ketika sudah jadi dalam bentuk 3 dimensi

Gambar 0-6 Kuesioner Responden Ketiga

Farhan Firmansyah

Mahasiswa

Submission Date
July 2, 2020 09:00

Nama	Farhan Firmansyah
Profesi	Mahasiswa
Institusi (jika ada)	ITS
Antarmuka aplikasi mudah dimengerti	Cukup setuju
Aplikasi memiliki tata letak tombol yang mudah dilihat/dikenali	Setuju
Aplikasi dapat mengenali denah dengan akurat	Cukup setuju
Saya merasa akan terbantu dengan adanya aplikasi ini	Setuju
Saya merasa tertarik untuk menggunakan aplikasi ini	Cukup setuju
Saran:	Desain UI pada aplikasi bisa lebih bagus dan tingkat akurasi semakin dipertajam

Gambar 0-7 Kuesioner Responden Keempat

Brigita L Mahasiswa

Submission Date
July 2, 2020 09:04

Nama	Brigita L
Profesi	Mahasiswa
Institusi (jika ada)	ITS
Antarmuka aplikasi mudah dimengerti	Setuju
Aplikasi memiliki tata letak tombol yang mudah dilihat/dikenali	Setuju
Aplikasi dapat mengenali denah dengan akurat	Tidak setuju
Saya merasa akan terbantu dengan adanya aplikasi ini	Cukup setuju
Saya merasa tertarik untuk menggunakan aplikasi ini	Setuju
Saran:	-

Gambar 0-8 Kuesioner Responden Kelima

Naufal Fadhil

Submission Date
July 2, 2020 09:09

Nama	Naufal Fadhil
Antarmuka aplikasi mudah dimengerti	Sangat setuju
Aplikasi memiliki tata letak tombol yang mudah dilihat/dikenali	Setuju
Aplikasi dapat mengenali denah dengan akurat	Setuju
Saya merasa akan terbantu dengan adanya aplikasi ini	Setuju
Saya merasa tertarik untuk menggunakan aplikasi ini	Sangat setuju

Gambar 0-9 Kuesioner Responden Keenam

RAIHAN ERVIANSYAH EKAPUTRA
MAHASISWA, GAME DEVELOPER

Submission Date
July 2, 2020 09:20

Nama	RAIHAN ERVIANSYAH EKAPUTRA
Profesi	MAHASISWA, GAME DEVELOPER
Institusi (jika ada)	ITS, CRESCENT MOON GAMES
Antarmuka aplikasi mudah dimengerti	Cukup setuju
Aplikasi memiliki tata letak tombol yang mudah dilihat/dikenali	Cukup setuju
Aplikasi dapat mengenali denah dengan akurat	Tidak setuju
Saya merasa akan terbantu dengan adanya aplikasi ini	Sangat setuju
Saya merasa tertarik untuk menggunakan aplikasi ini	Setuju
Saran:	Implementasi ARCore dan ARKit untuk membantu stabilitas 3D Object tracking, Precise detection agar 3D model dapat muncul secara presisi diatas denah, jangan gunakan seluruh denah sebagai marker, tetapi gunakan visual cues yang lebih close-up disematkan ke denah sehingga tracking tidak terlalu jauh. Redesign UI agar lebih visually appealing. Saran saya adalah penambahan opsi fitur Instant 3D AR, dimana user bisa memunculkan AR dari 3D model yang diupload tanpa memerlukan tracker. Interaktivitas juga dapat membantu visualisasi, seperti day and night cycle, section cut dan memunculkan informasi BIM dalam mode AR.

Gambar 0-10 Kuesioner Responden Ketujuh

BIODATA PENULIS



Muhammad Auliaramadani, lahir di Tanjung Batu pada tanggal 24 Desember 1997. Penulis merupakan anak ke pertama dari 4 bersaudara dari pasangan Bapak Budiono dan Ibu Syamsiah. Penulis lulus dari SMAN 1 Tanjungpinang pada tahun 2016 dan melanjutkan studinya di Departemen Informatika Institut Teknologi Sepuluh Nopember Surabaya.

Selama menempuh pendidikan di Departemen Informatika ITS, penulis aktif mengikuti organisasi antara lain staf Departemen Hubungan Luar

Himpunan Mahasiswa Teknik Computer-Informatika (HMTC) 2017/2018 dan menjadi staf Perlengkapan Schematics ITS 2018 dan 2019.

Dalam menyelesaikan Pendidikan sarjana, penulis mengambil bidang minat Interaksi, Grafika dan Seni (IGS). Penulis dapat dihubungi melalui alamat *e-mail*: muhammadauliaramdan@gmail.com.