APLIKASI *AUGMENTED REALITY* SEBAGAI MEDIA SIMULASI IKATAN KIMIA BERBASIS ANDROID MENGGUNAKAN METODE *FAST CORNER DETECTION*

SKRIPSI

Oleh :

MOHAMMAD SYAHROFI IRSYAD
NIM. 09650086

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

2016

HALAMAN PENGAJUAN

APLIKASI *AUGMENTED REALITY* SEBAGAI MEDIA SIMULASI IKATAN KIMIA BERBASIS ANDROID MENGGUNAKAN METODE *FAST CORNER DETECTION*

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:
MOHAMMAD SYAHROFI IRSYAD
NIM. 09650086

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

HALAMAN PERSETUJUAN

APLIKASI *AUGMENTED REALITY* SEBAGAI MEDIA SIMULASI IKATAN KIMIA BERBASIS ANDROID MENGGUNAKAN *METODE FAST CORNER DETECTION*

SKRIPSI

Oleh:

MOHAMMAD SYAHROFI IRSYAD
NIM. 09650086

Telah Disetujui pada tanggal …….. 20…

Dosen Pembimbing I,	Dosen Pembimbing II,
Dr. Cahyo Crysdian	**Ainatul Mardhiyah, S.Kom, M.Cs**
NIP. 19740424 200901 1 008	

Mengetahui:
Ketua Jurusan Teknik Informatika,

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

LEMBAR PENGESAHAN

APLIKASI *AUGMENTED REALITY* SEBAGAI MEDIA SIMULASI IKATAN KIMIA BERBASIS ANDROID MENGGUNAKAN METODE *FAST CORNER DETECTION*

SKRIPSI

Oleh:

MOHAMMAD SYAHROFI IRSYAD
NIM. 09650086

Telah Dipertahankan di Depan Dewan Penguji
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
pada Tanggal ……. 20...

Susunan Dewan Penguji			Tanda Tangan
1. Penguji Utama	:	**A'la Syauqi, M.Kom** **NIP. 19771201 200801 1 007**	( )
2. Ketua	:	**Dr. M. Amin Hariyadi, M.T** **NIP. 19670118 200501 1 001**	( )
3. Sekretaris	:	**Dr. Cahyo Crysdian** **NIP. 19740424 200901 1 008**	( )
4. Anggota	:	**Ainatul Mardhiyah, S.Kom, M.Cs**	( )

Disahkan Oleh:
Ketua Jurusan Teknik Informatika,

Dr. Cahyo Crysdian
NIP. 19740424 200901 1 008

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : Mohammad Syahrofi Irsyad

NIM : 09650086

Fakultas : Sains dan Teknologi

Jurusan : Teknik Informatika

menyatakan bahwa "**Skripsi**" yang saya buat untuk memenuhi persyaratan kelulusan pada Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang, dengan judul:

APLIKASI AUGMENTED REALITY SEBAGAI MEDIA SIMULASI IKATAN KIMIA BERBASIS ANDROID MENGGUNAKAN METODE FAST CORNER DETECTION

adalah hasil karya saya sendiri, bukan "**duplikasi**" dari karya orang lain.

Selanjutnya apabila di kemudian hari ada "**klaim**" dari pihak lain, bukan menjadi tanggung jawab Dosen Pembimbing dan atau pihak Fakultas Sains dan Teknologi, tetapi menjadi tanggung jawab saya sendiri.

Demikian surat pernyataan ini saya buat dengan sebenarnya dan tanpa paksaan dari siapapun.

Malang, 27 Juni 2016
Hormat saya,

M. SYAHROFI IRSYAD
NIM. 09650086

HALAMAN MOTTO

إِنَّ مَعَ ٱلْعُسْرِ يُسْرًا ۝٦ فَإِذَا فَرَغْتَ فَٱنصَبْ ۝٧ وَإِلَىٰ رَبِّكَ فَٱرْغَب ۝٨

"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)

"意思あるところに道あり"

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM PUSAT PERPUSTAKAAN

HALAMAN PERSEMBAHAN

ٱلْحَمْدُ لِلَّهِ رَبِّ ٱلْعَٰلَمِينَ

Segala puji bagi Allah Tuhan semesta alam

Aku persembahkan karya ini untuk orang-orang yang terkasih:

Ayah dan Bunda tercinta,

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

Seluruh keluargaku, atas semua doa dan dukungannya.

Teman-teman almamater,

terima kasih telah berjuang bersama-sama hingga ahir.

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM PUSAT PERPUSTAKAAN

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT, karena atas rahmat dan karunia-Nya penyusunan skripsi yang berjudul "***Aplikasi Augmented Reality Sebagai Media Simulasi Ikatan Kimia Berbasis Android Menggunakan Metode Fast Corner Detection***" dapat diselesaikan dengan baik. Skripsi ini merupakan prasyarat untuk memenuhi syarat utama kelulusan program pendidikan Strata 1 pada Jurusan Teknik Informatika, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Penulis menyadari bahwa dalam proses penulisan skripsi ini banyak mengalami kendala, namun berkat bantuan, bimbingan, kerjasama dari berbagai pihak dan berkah dari Allah SWT sehingga kendala-kendala yang dihadapi tersebut dapat diatasi.

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

1. Dr. Cahyo Crysdian selaku Pembimbing I dan Ketua Jurusan Teknik Informatika UIN Maliki yang telah mendukung semua proses penelitian penulis.
2. Ainatul Mardhiyah, S.Kom, M.Cs. Selaku dosen pembimbing II yang telah memberikan masukan dan bimbingannya dalam proses penyusunan laporan skripsi ini.
3. Totok Chamidy, 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. Teman-teman 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.

Akhirnya, dengan segala kerendahan hati penulis menyadari masih banyak terdapat kekurangan-kekurangan, sehingga penulis mengharapkan adanya saran dan kritik yang bersifat membangun demi kesempurnaan skripsi ini. Penulis berharap semoga laporan skripsi ini dapat bermanfaat bagi seluruh pihak yang membutuhkan.

Malang, 27 Juni 2016

Penulis

DAFTAR ISI

HALAMAN JUDUL
HALAMAN PENGAJUAN i
HALAMAN PERSETUJUAN ii
LEMBAR PENGESAHAN iii
SURAT PERNYATAAN iv
HALAMAN MOTTO v
HALAMAN PERSEMBAHAN vi
KATA PENGANTAR vii
DAFTAR ISI ix
DAFTAR GAMBAR xii
DAFTAR TABEL xiv
DAFTAR LAMPIRAN xv
ABSTRAK xvi
ABSTRACT xvii
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Rumusan Masalah 3
1.3 Tujuan 3
1.4 Manfaat 3
1.5 Batasan Masalah 3
BAB II KAJIAN PUSTAKA 4
2.1 Augmented Reality 4
2.1.1 Sejarah Augmented Reality 5
2.1.2 Proses Augmented Reality 7
2.1.3 Penerapan Augmented Reality 7
2.1.4 Natural Feature Tracking FAST Corner Detection 8
2.2 Unity 3D 9
2.3 Vuforia 10
2.3.1 Vuforia API Reference 11
2.3.2 Marker 12
2.3.3 Deteksi Marker 13
2.4 Ilmu Kimia 13
2.4.1 Teori Atom 14

2.4.2 Simbol Titik Lewis 16
2.4.3 Ikatan Kimia 17
2.4.4 Teori VSEPR 19
BAB III ANALISIS DAN PERANCANGAN SISTEM 21
3.1 Analisis Sistem 21
3.2 Arsitektur Sistem 21
3.3 Alur Sistem 23
3.4 Perancangan sistem 24
3.4.1 Flowchart 24
3.4.2 Algoritma FAST Corner detection 26
3.4.3 Pembuatan Marker 29
3.5 Usecase Diagram 35
3.5.1 Definisi Usecase 35
3.5.2 Skenario Usecase 36
3.5.3 Activity Diagram 40
3.5.4 Sequence Diagram 45
3.5.5 Class Diagram 49
3.5.6 User Interface 51
3.5.7 Desain Marker 52
BAB IV IMPLEMENTASI DAN PENGUJIAN SISTEM 56
4.1 Implementasi Sistem 56
4.2 Implementasi Interface 56
4.2.1 Tampilan Pembuka 56
4.2.2 Tampilan Menu Utama 57
4.2.3 Tampilan Help 57
4.2.4 Tampilan About 57
4.2.5 Tampilan Exit 58
4.2.6 Tampilan Deteksi Marker 58
4.3 Pengujian Sistem 59
4.3.1 Pengujian Proses 59
4.3.2 Pengujian Perangkat Android 65
4.4 Pengujian Tracking Marker 67
4.4.1 Pengujian Intensitas Cahaya 67
4.4.2 Pengujian Oklusi 68
4.4.3 Pengujian Akurasi 69
4.5 Integrasi Nilai Islam 71

BAB V PENUTUP 73
5.1 Kesimpulan 73
5.2 Saran 74
DAFTAR PUSTAKA 75

UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM
PUSAT PERPUSTAKAAN

DAFTAR GAMBAR

Gambar 2.1 Alur aplikasi augmented reality ()..... 7
Gambar 2.2 User interface Unity3D 10
Gambar 2.3 Sistem high level vuforia (http://developer.vuforia.com). 11
Gambar 2.4 Contoh marker..... 12
Gambar 2.5 Diagram block deteksi marker (http://developer.vuforia.com/library/reference)..... 13
Gambar 2.6 Lewis dot symbol (Raymon, 2010)..... 16
Gambar 2.7 Ikatan ion NaCl (Zumdahl, 2014) 18
Gambar 2.8 Ikatan kovalen H2O yang di gambarkan dengan simbol titik lewis (Zumdahl, 2014)..... 19
Gambar 2.9 Geometri tolakan pasangan antar elektron (Raymon, 2010)..... 20
Gambar 3.1 Diagram blok arsitektur sistem 22
Gambar 3.2 Alur sistem 23
Gambar 3.3 Blok diagram aplikasi AR..... 24
Gambar 3.4 Alur proses pendeteksian marker 25
Gambar 3.5 Flowchart inisialisasi marker 26
Gambar 3.6 Titik awal p..... 27
Gambar 3.7 Titik *p* pada koordinat $n = 1$, $n = 2$, $n = 3$,dan $n = 4$ 28
Gambar 3.8 Perbandingan intensitas pada seluruh titik..... 28
Gambar 3.9 Workflow pembuatan marker developer vuforia (http://developer.vuforia.com). 30
Gambar 3.10 Flowchart alur proses pembuatan marker 30
Gambar 3.11 Psudocode resize gambar 31
Gambar 3.12 Psudocode algoritma convert grayscale 31
Gambar 3.13 Psudocode algoritma convert histogram 32
Gambar 3.14 Psudocode algoritma convert histogram 32
Gambar 3.15 Flowchart FAST corner detection..... 33
Gambar 3.16 Psudocode Fast Corner Detection 34
Gambar 3.17 Contoh hasil penentuan titik point 35
Gambar 3.18 Usecase diagaram Apliaksi 35
Gambar 3.19 Activity diagram start..... 41

Gambar 3.20 Activity diagram tracking marker 42
Gambar 3.21 Activity diagram tampilkan objek 43
Gambar 3.22 Activity diagram bonding atom 44
Gambar 3.23 Activity diagram dari tampilkan informasi 45
Gambar 3.24 Sequence diagram start 46
Gambar 3.25 Sequence diagram tracking marker 46
Gambar 3.26 Sequence diagram tampilkan object 47
Gambar 3.27 Sequence diagram bonding 48
Gambar 3.28 Sequence diagram tampilkan informasi senyawa 49
Gambar 3.29 Class diagram aplikasi 50
Gambar 4.1 Tampilan pembuka splash screen 56
Gambar 4.2 Tampilan menu utama 57
Gambar 4.3 Tampilan help 57
Gambar 4.4 Tampilan about 58
Gambar 4.5 Tampilan exit 58
Gambar 4.6 Tampilan Deteksi Marker 59
Gambar 4.7 Kode program menu utama 59
Gambar 4.8 Kamera siap untuk mendeteksi marker 60
Gambar 4.9 Pendeteksian marker 60
Gambar 4.10 Center atom ditambahkan pada marker bond 61
Gambar 4.11 Atom ikatan ditambahkan pada marker bond 61
Gambar 4.12 Ikatan atom terbentuk 61
Gambar 4.13 Kode program ikatan (bonding) atom menambah atom 62
Gambar 4.14 Kode program ikatan (bonding) atom menggambar garis ikatan 62
Gambar 4.15 Informasi senyawa pada marker hint 63
Gambar 4.16 Kode program Informasi senyawa pada marker hint 63
Gambar 4.17 Help pada aplikasi AR 64
Gambar 4.18 About pada aplikasi 64
Gambar 4.19 Notifikasi keluar aplikasi 65

DAFTAR TABEL

Tabel 3.1 Deskripsi Usecase 36
Tabel 3.2 Skenario Start 36
Tabel 3.3 Skenario tracking marker 37
Tabel 3.4 Skenario tampilkan object 3d 38
Tabel 3.5 Skenario bonding atom 39
Tabel 3.6 Tampilkan informasi senyawa 39
Tabel 3.7 User Interface 51
Tabel 3.8 Disain marker 53
Tabel 4.2 Spesifikasi perangkat android 65
Tabel 4.3 Hasil pengujian pada perangkat android 66
Tabel 4.4 Hasil pengujian intensitas cahaya menggunakan perangkat android sony experia m 67
Tabel 4.5 Pengujian oklusi mengunakan perangkat android sony experia m 68
Tabel 4.6 Pengujian Akurasi 70

DAFTAR LAMPIRAN

Lampiran 1. Printable Marker.....77

UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM
PUSAT PERPUSTAKAAN

ABSTRAK

Irsyad, Mohammad Syahrofi. 2016. ***Aplikasi Augmented Reality Sebagai Media Simulasi Ikatan Kimia Berbasis Android Menggunakan Metode FAST Corner Detection***. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Cahyo Crysdian (II) Ainatul Mardhiyah, S.Kom, M.Cs

Kata kunci: augmented reality, fast corner detection, unity, ikatan kimia, vsepr, android.

Teknologi augmented reality adalah teknologi yang menggabungkan benda maya 2D atau 3D ke dalam sebuah lingkungan nyata. Aplikasi augmented reality ini dibuat menggunakan software game engine unity dan vuforia dengan menerapkan metode FAST corner detection. Aplikasi augmented reality ini merupakan media yang digunakan untuk menvisualisasikan ikatan kimia dari beberapa senyawa dalam bentuk tiga dimensi dan mensimulasikan pembentukan ikatan kimia berdasarkan teori VSEPR secara interaktif, sehingga dapat membantu dalam mempelajari ilmu kimia khususnya dalam hal ikatan kimia dan menimbulkan efek menarik bagi pengguna. Aplikasi augmented reality ini telah melalui pengujian proses, pengujian sistem pada perangkat android berbeda, dan pengujian pendeteksian marker. Hasil pengujian proses menunjukkan bahwa setiap proses pada aplikasi ini berjalan sesuai dengan rancangan. Hasil pengujian sistem menunjukkan perbedaan spesifikasi setiap perangkat android mempengaruhi cepat lambatnya proses loading aplikasi. Hasil pengujian pendeteksian marker menunjukkan bahwa intensitas cahaya, oklusi, jarak dan sudut pendeteksian mempengaruhi proses pendeteksian marker. Ketika intensitas cahaya rendah, proses pendeteksian marker semakin melambat. Aplikasi tetap dapat mendeteksi marker dengan baik hingga marker tertutup 35%. Ketika marker tertutup 50%, pendeteksian marker mulai melambat, dan ketika marker tertutup 75%, aplikasi tidak dapat mendeteksi marker dan menampilkan objek 3D. Hasil dari pengujian akurasi menunjukkan bahwa pedeteksian marker sangat baik ketika jarak marker ke kamera yaitu 20cm – 60cm dan pada sudut 30º, 45º, 60º, 75º, 90º.

ABSTRACT

Irsyad, Mohammad Syahrofi. 2016. ***Android-Based Augmented Reality App For Chemical Bonds Simulation Media Using FAST Corner Detection Method***. Thesis. Informatics Department, Science and Technology Faculty, State Islamic University of Maulana Malik Ibrahim Malang. Preceptor: (I) Dr. Cahyo Crysdian (II) Ainatul Mardhiyah, Kom, M.Cs

---

Keywords: augmented reality, fast corner detection, unity, chemical bonding, vsepr, android.

Augmented reality technology is a technology that combines 2D or 3D virtual object into a real environment. This augmented reality application created using Unity game engine software and vuforia by applying the FAST corner detection method. Augmented reality application is a media that is used to visualize the chemical bonds of several compounds in the form of three-dimensional and simulates the formation of chemical bonds by VSEPR theory interactively, so it can help in studying chemistry, especially in terms of chemical bonds and the effect of interest to the user. Augmented reality applications have been through the testing process, testing of the system at different android devices, and testing marker detection. The test results show that the process of every process on the app works as designed. System test results show differences in the specifications of each device android affect how quickly the application loading process. The test results show that the marker detection light intensity, occlusion, distance and angle detection marker affect the detection process. When the light intensity is low, the marker detection process slowed. Applications can still detect a good marker to marker covered with 35%. When the marker is covered 50%, the detection marker starts to slow down, and when the marker is covered 75%, the application can not detect a marker and display 3D objects. Results of the test showed that the detection accuracy is very good marker when the distance marker to the camera that is 20cm - 60cm and at an angle of 30º, 45º, 60º, 75º, 90º.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Seiring perkembangan ilmu pengetahuan dan teknologi (IPTEK) yang begitu cepat, setiap informasi dan pengetahuan dapat disampaikan dalam bentuk visual. Visualisasi sebagai salah satu bagian penting dalam pengembangan sebuah aplikasi yang bersifat interaktif kepada penggunanya haruslah dapat menggambarkan keadaan yang sebenarnya. Hal ini tentunya dapat dilakukan menggunakan teknologi yang bersifat interaktif nyata (augmented reality).

Augmented reality (AR) merupakan sistem yang menggabungkan objek virtual dengan objek nyata. Menurut Ronald Azuma pada tahun 1997, augmented reality adalah menggabungkan dunia nyata dan virtual, bersifat interaktif secara real time, dan merupakan animasi 3D. Sistem ini lebih menekankan pada unsur nyata (real) dari pada unsur virtual, lain halnya dengan virtual reality (VR) yang sepenuhnya merupakan virtual environment. AR mengijinkan penggunanya untuk berinteraksi secara real time dengan sistem. AR merupakan konsep perpaduan antara virtual reality dengan world reality. Sehingga objek virtual 3D seolah-olah tampak nyata dan menyatu dengan dunia nyata.

Ilmu kimia merupakan bidang ilmu pengetahuan alam yang mempelajari tentang materi yang meliputi struktur, susunan, sifat, dan perubahan materi serta energi yang meyertainya. Kimia berhubungan dengan interaksi materi yang dapat melibatkan dua zat. Pada kimia tradisional terjadi interaksi antara zat kimia dalam reaksi kimia, yang mengubah satu atau lebih zat menjadi satu atau lebih zat lain. Ikatan kimia adalah gaya tarik menarik antara atom-atom sehingga atom-atom

tersebut tetap berada bersama-sama dan terkombinasi dalam senyawa. Ikatan ion adalah ikatan yang terbentuk sebagai akibat adanya gaya tarik-menarik antara ion positif dan ion negatif. Ikatan kovalen adalah ikatan yang terjadi bila terdapat pemakaian bersama sepasang atau lebih elektron yang menyebabkan atom-atom yang berikatan memperoleh susunan oktet.

Dalam surat Yaasiin ayat 36, Allah SWT berfirman:

سُبْحَانَ الَّذِي خَلَقَ الْأَزْوَاجَ كُلَّهَا مِمَّا تُنْبِتُ الْأَرْضُ وَمِنْ أَنْفُسِهِمْ وَمِمَّا لَا يَعْلَمُونَ

"Maha suci Tuhan yang telah menciptakan pasangan-pasangan semuanya, baik dari apa yang ditumbuhkan oleh bumi dan dari diri mereka maupun dari apa yang tidak mereka ketahui". (Q.S Yaasiin: 36)

Ayat di atas menjelaskan bahwa Allah telah menciptakan semuaya dalam pasangan-pasangan hal ini sesuai dengan konsep ikatan kimia bahwa atom akan berpasangan untuk mencapai suatu kestabilan.

Kesulitan mempelajari ilmu kimia ini terkait dengan ciri ilmu kimia itu sendiri, yaitu karena ilmu kimia bersifat abstrak, memiliki sifat yang berurutan dan memiliki perkembangan materi yang cukup pesat, serta juga merupakan penyederhanaan dari berbagai materi ilmu lainnya (Sirhan, 2007). Untuk itu diperlukan sebuah media yang interaktif agar dapat mempermudah dalam mempelajari ilmu kimia khususnya dalam hal ikatan kimia.

Bermula dari permasalahan diatas menjadi dasar penulis menerapkan augmented reality (AR) untuk membantu dalam mempelajari ilmu kimia khususnya dalam memahami ikatan-ikatan kimia dan juga akan menimbulkan efek menarik dan menyenangkan dalam mempelajarinya.

1.2 Rumusan Masalah

Seberapa besar keakuratan FAST corner detection pada augmented reality aplikasi media simulasi ikatan kimia berbasis android?

1.3 Tujuan

Adapun tujuan aplikasi simulasi ini adalah menvisualisikan dan mensimulasikan ikatan-ikatan kimia dari beberapa senyawa dalam bentuk tiga dimensi (3D) yang interaktif, sehingga dapat membantu dalam mempelajari ilmu kimia khususnya dalam hal ikatan kimia dan menimbulkan efek menarik bagi pengguna.

1.4 Manfaat

Manfaat yang diharapakan dari penelitian ini sebagai berikut:

1. Mengembangkan teknologi augmented reality dalam simulasi ikatan kimia
2. Menjadi media belajar interaktif ilmu kimia khususnya dalam ikatan-ikatan kimia.

1.5 Batasan Masalah

Batasan masalah diasumsikan sebagai berikut:

1. Aplikasi ini dibagun menggunakan unity3D dan vuforia
2. Bentuk ikatan senyawa yang ditampilkan pada aplikasi ini adalah: $BeCl_2$, BF_3, SO_2, NH_3, CH_4, NH_4, H_2O, PCl_5, SF_4, BrF_3, SF_6, XeF_4.
3. Aplikasi dijalankan pada perangkat mobile berbasis Android dengan versi 4.1.x keatas.

BAB II

KAJIAN PUSTAKA

2.1 Augmented Reality

Augmented Reality (AR) adalah sebuah istilah untuk lingkungan yang menggabungkan dunia nyata dan dunia virtual yang dibuat oleh komputer sehingga batas antara keduanya menjadi sangat tipis. Sistem ini lebih dekat kepada lingkungan nyata (real). Karena itu, reality lebih diutamakan pada sistem ini (Kurniawan, 2011).

Ronald Azuma pada tahun 1997 mendefinisikan augmented reality sebagai sistem yang memiliki karakteristik yaitu: (1) Menggunakan lingkungan nyata dan virtual, (2) Berjalan secara interaktif dalam waktu nyata (real-time), (3) Integrasi dalam tiga dimensi (3D).

Lebih lanjut, Azuma menuliskan bahwa secara sederhana AR bisa didefinisikan sebagai lingkungan nyata yang ditambahkan objek virtual. Penggabungan objek nyata dan virtual dimungkinkan dengan teknologi display yang sesuai, interaktivitas dimungkinkan melalui perangkat-perangkat input tertentu.

Menurut Stephen Cawood dan Mark Fiala bahwa augmented reality merupakan cara alami untuk mengeksplorasi objek 3D dan data, AR merupakan suatu konsep perpaduan antara virtual reality dan world reality. Sehingga objek-objek virtual 2D dan 3D seolah-olah terlihat nyata dan menyatu dengan dunia nyata. Pada teknologi AR, pengguna dapat melihat dunia nyata yang ada di sekelilingnya dengan penambahan objek virtual yang dihasilkan oleh komputer. (Cawood, 2008)

AR merupakan variasi dari Virtual Environments (VE), atau yang lebih dikenal dengan istilah Virtual Reality (VR). Teknologi VR membuat pengguna tergabung dalam sebuah lingkungan virtual secara keseluruhan. Ketika tergabung dalam lingkungan tersebut, pengguna tidak bisa melihat lingkungan nyata di sekitarnya. Sebaliknya, AR memungkinkan pengguna untuk melihat lingkungan nyata, dengan objek virtual yang ditambahkan atau tergabung dengan lingkungan nyata. Tidak seperti VR yang sepenuhnya menggantikan lingkungan nyata, AR sekedar menambahkan atau melengkapi lingkungan nyata (Azuma, 1997).

Dengan bantuan teknologi augmented reality, lingkungan nyata di sekitar kita akan dapat berinteraksi dalam bentuk digital (virtual). Informasi-informasi tentang obyek dan lingkungan disekitar kita dapat ditambahkan ke dalam sistem augmented reality yang kemudian informasi tersebut ditampilkan diatas layer dunia nyata secara real-time seolah-olah informasi tersebut adalah nyata.

Augmented reality merupakan salah satu cabang di bidang teknologi yang belum terlalu lama, namun memiliki perkembangan yang sangat cepat. Perkembangan augmented reality pada industri mobile phone juga mempunyai perkembangan yang paling cepat.

2.1.1 Sejarah Augmented Reality

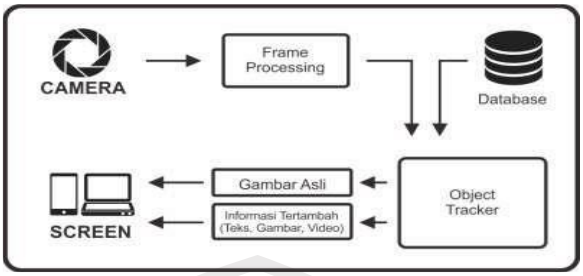
Sejarah tentang augmented reality dimulai dari tahun 1957-1962, ketika seorang penemu yang bernama Morton Heilig, seorang cinematografer, menciptakan dan memapatenkan sebuah simulator yang disebut sensorama dengan visual, getaran dan bau. Pada tahun 1966, Ivan Sutherland menemukan head-mounted display yang diklaim sebagai jendela ke dunia virtual.

Tahun 1975 seorang ilmuwan bernama Myron Krueger menemukan Videoplace yang memungkinkan pengguna, dapat berinteraksi dengan objek virtual untuk pertama kalinya. Tahun 1989, Jaron Lanier memeperkenalkan virtual reality dan menciptakan bisnis komersial pertama kali di dunia maya, Tahun 1992 mengembangkan augmented reality untuk melakukan perbaikan pada pesawat boeing, dan pada tahun yang sama, LB Rosenberg mengembangkan salah satu fungsi sistem AR, yang disebut Virtual Fixtures, yang digunakan di Angkatan Udara AS Armstrong Labs, dan menunjukan manfaatnya pada manusia, dan pada tahun 1992 juga, Steven Feiner, Blair Maclntyre dan dorée Seligmann, memperkenalkan untuk pertama kalinya Major Paper untuk perkembangan Prototype AR (Sudareswaran, 2013).

Pada tahun 1999, Hirokazu Kato, mengembangkan ArToolkit di HITLab dan didemonstrasikan di SIGGRAPH, pada tahun 2000, Bruce. H. Thomas, mengembangkan ARQuake, sebuah Mobile Game AR yang ditunjukan di International Symposium on Wearable Computers.

Pada tahun 2008, Wikitude AR Travel Guide, memperkenalkan Android G1 Telephone yang berteknologi AR. Tahun 2009, Saqoosha memperkenalkan FLARToolkit yang merupakan perkembangan dari ArToolkit. FLARToolkit memungkinkan kita memasang teknologi AR pada sebuah website, karena output yang dihasilkan oleh FLARToolkit berbentuk Flash. Ditahun yang sama, Wikitude Drive meluncurkan sistem navigasi berteknologi AR di Platform Android. Tahun 2010, Acrossair menggunakan teknologi AR pada iPhone 3Gs.

2.1.2 Proses Augmented Reality



Gambar 2.1 Alur aplikasi augmented reality (Kurniawan, 2011)

Pada Gambar 2.1 diatas merupakan gambaran sederhana dari alur aplikasi augmented reality. Proses dimulai dari pengambilan gambar marker dengan webcam. Marker tersebut berdasarkan feature yang dimiliki, kemudian masuk ke dalam object tracker yang disediakan oleh sdk (softawre development kit). Selain itu, marker tersebut telah didaftarkan dan disimpan dalam database agar dapat menampilkan informasi yang sesuai. Hasil keluaran pelacakan marker ditampilkan pada layar komputer atau smartphone.

2.1.3 Penerapan Augmented Reality

Bidang-bidang yang pernah menerapkan teknologi augmented reality adalah:

1. Kedokteran (Medical)

   Bidang kedokteran menerapkan AR pada visualisasi penelitian mereka misal untuk simulasi operasi, simulasi pembuatan vaksin virus, dll.

2. Hiburan (Entertainment)

   Dalam dunia hiburan biasanya augmented reality dipakai untuk efek perfilman, permainan di smartphone, majalah, dll.

3. Latihan Militer (Militer Training)

Militer telah menerapkan augmented reality pada latihan tempur mereka. Sebagai contoh, militer menggunakan augmented reality untuk membuat sebuah permainan perang, dimana prajurit masuk ke dalam dunia game tersebut dan seolah-olah seperti melakukan perang sungguhan.

4. Enginering

   Augmented reality digunakan untuk latihan para engineer untuk bereksperimen. Misalnya engineer mesin, menggunakan augmented reality untuk memperbaiki mobil yang rusak.

5. Robotics dan Telerobotics

   Dalam dunia robotik, seorang operator robot menggunakan visual dalam mengendalikan robot itu. Jadi, penerapan augmented reality dibutuhkan di dunia robot.

6. Consumer Design

   Augmented reality telah digunakan dalam mempromosikan produk. Sebagai contoh, seorang pengembang perumahan menggunakan augmented reality untuk memberikan informasi tentang perumahan secara 3D. Ataupun memberikan informasi tentang mobil yang dikembangkan. (Anggi Andriyadi, 2011)

2.1.4 Natural Feature Tracking FAST Corner Detection

Qualcomm Augmented Reality (QCAR) merupakan salah satu SDK untuk merancang aplikasi AR pada Vuforia. QCAR menerapkan konsep natural feature tracking untuk mendeteksi dan mengenali image target. QCAR juga menerapkan metode FAST (Feature from Accelerate Segmen Test).

FAST adalah suatu algoritma yang dikembangkan oleh Edward Rosten, Reid Porter, and Tom Drummond. FAST corner detection ini dibuat dengan tujuan mempercepat waktu komputasi secara real-time dengan konsekuensi menurunkan tingkat akurasi pendeteksian sudut. (Harris, 1988)

2.2 Unity 3D

Unity 3D merupakan alat yang digunakan untuk membangun games 3 dimensi yang telah terintergrasi untuk menghasilkan suatu animasi 3 dimensi secara real time. Game developer tool ini dilengkapi dengan GUI yang memudahkan untuk membuat, mengedit dan membuat script untuk menciptakan sebuah game 3D.

Unity 3D tidak hanya dapat digunakan untuk membangun game PC saja, tetapi juga dapat digunakan untuk membangun game console seperti Nintendo Wii, PS3, Xbox 360, juga Iphone, dan Android (Hernowo, 2014).

Unity 3D ini dikembangkan oleh Unity Technologies yang dibangun di tahun 2004 oleh David Helgason, Nicholas Francis dan Joachim Ante. Game engine ini dibangun atas kepedulian mereka terhadap indie developer yang tidak bisa membeli game engine karena terlalu mahal. Fokus perusahaan ini adalah membuat sebuah perangkat lunak yang bisa digunakan oleh semua orang, khususnya untuk membangun sebuah game. Pada tahun 2009, Unity diluncurkan secara gratis dan pada April 2012 Unity mencapai popularitas tertinggi dengan lebih dari satu juta developer terdaftar di seluruh dunia. (Roedavan, 2014)

Unity juga memiliki IDE (Integrated Development Environment) yaitu MonoDevelop yang bertujuan untuk mengintegrasikan semua script yang dibuat kedalam Unity, sehingga dapat langsung diproses. Game engine unity

dikembangkan dengan menggunakan bahasa pemrograman C/C++ dan juga mendukung berbagai macam bahasa pemrograman yang lainnya seperti Javasript.

Fitur scripting yang disediakan, mendukung 3 bahasa pemrograman yaitu JavaScript, C#, dan Boo. Flexible, easy moving, rotating, dan scaling objects hanya perlu sebaris kode, begitu juga dengan duplicating, removing, dan changing properties. Visual Properties Variables yang di definisikan dengan scripts ditampilkan pada Editor. Bisa digeser, di drag and drop, bisa memilih warna dengan color picker. Berbasis .NET. Artinya running program dilakukan dengan Open Source .NET platform, Mono (www.unity3d.com).

Gambar 2.2 User interface Unity3D

2.3 Vuforia

Vuforia adalah (Software Development Kit) SDK yang disediakan oleh Qualcomm untuk membantu para developer membuat aplikasi-aplikasi Augmented Reality (AR) di mobile phones (iOS, Android). Dulunya lebih dikenal sebagai QCAR (Qualcomm Company Augmented Reality) yang menggunakan teknologi computer vision untuk mengenali dan melacak gambar planar (image target) dan objek 3D sederhana secara realtime.

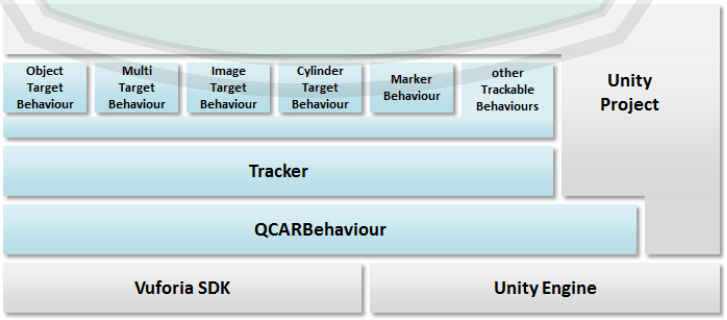
Kemampuan registrasi citra memungkinkan pengembang untuk mengatur posisi dan virtual orientasi objek, seperti model tiga dimensi dan media lainnya, dalam kaitannya dengan gambar nyata ketika dilihat melalui kamera perangkat mobile.

Vuforia merupakan library yang digunakan sebagai pendukung adanya Augmented reality pada Android dan iOS. Vuforia menganalisa gambar dengan menggunakan pendeteksi marker dan menghasilkan informasi 3D dari marker yang sudah dideteksi via API. Programmer juga dapat menggunakannya untuk membangun objek 3D virtual pada kamera.

Vuforia mendukung berbagai jenis target gambar 2D dan 3D marker termasuk target gambar markerless. Fiture tambahan dari SDK ini yaitu deteksi oklusi lokal menggunakan tombol virtual. Vuforia meyediakan API (Aplication programing interface) pada lingkungan C++, java dan objective C (http://developer.vuforia.com).

2.3.1 Vuforia API Reference

Vuforia API reference berisi informasi tentang hirarki kelas dan fungsi member dari QCAR SDK. Sistem QCAR SDK digambarkan sebagai berikut:



Gambar 2.3 Sistem high level vuforia (http://developer.vuforia.com).

Pengembang diijinkan untuk melakukan koneksi antara aplikasi yang sudah dibuat dengan library static i.e libQCAR.a pada iOS atau libQCAR.so pada Android.

Library libQCAR.so inilah yang menjadi sistem aplikasi sehingga pelacakan target bisa dilakukan dengan mudah (http://developer.vuforia.com).

2.3.2 Marker

Marker merupakan sebuah penanda khusus yang memiliki pola tertentu sehingga saat kamera mendeteksi marker, objek 3 dimensi dapat ditampilkan. Augmented reality saat ini melakukan perkembangan besar-besaran, salah satunya pada bagian marker. Marker pertama adalah marker based tracking. Marker Based Tracking ini sudah lama dikembangkan sejak 1980-an dan pada awal 1990-an mulai dikembangkan untuk penggunaan Augmented Reality.

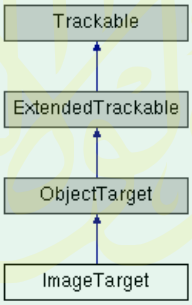
Markerless, perkembangan terbaru marker ini merupakan salah satu metode Augmented Reality tanpa menggunakan frame marker sebagai obyek yang dideteksi. Dengan adanya Markerless Augmented Reality, maka, penggunaan marker sebagai tracking object yang selama ini menghabiskan ruang, akan digantikan dengan gambar, atau permukaan apapun yang berisi dengan tulisan, logo, atau gambar sebagai tracking object (obyek yang dilacak) agar dapat langsung melibatkan obyek yang dilacak tersebut sehingga dapat terlihat hidup dan interaktif (http://developer.vuforia.com/library/articles/training).

Gambar 2.4 Contoh marker

2.3.3 Deteksi Marker

Qualcom sebagai salah satu pengembang Aungmentd reality melakukan proses pendeteksian marker menggunakan pengenalan pola gambar. Metode yang digunakan dalam QCAR adalah Natural Features Traking dengan metode FAST Corner Detection yaitu pendeteksian dengan mencari titik-titik (interst point) atau sudut-sudut pada suatu gambar.

Proses pelacakan (registrasi marker) proses pelacakan adalah beberapa objek yang dapat dilacak dan diregistrasi oleh QCAR SDK. Dalam proses pelacakan ada beberapa parameter untuk menentukan objek yang akan dilacak. Adapun parameter tersebut adalah nama, ID, status dan posisi yang disimpan dalam state object. (http://developer.vuforia.com/library/articles/training).



Gambar 2.5 Diagram block deteksi marker (http://developer.vuforia.com/library/reference).

2.4 Ilmu Kimia

Kimia merupakan salah satu cabang dari ilmu sains. Sebenarnya pelajaran kimia sudah didapat dari sekolah dasar namun tergabung dalam pembelajaran sains (IPA). Kimia sering kali dianggap sebagai mata pelajaran yang sulit karena kebanyakan materi kimia bersifat abstrak sehingga sulit dipelajari. Hal ini sesuai dengan pendapat Zoller, “These abstract concepts are important because further

chemistry/science concepts or theories cannot be easily understood if these underpinning concepts are not sufficiently grasped by the Student" (Sirhan, 2007).

Ilmu kimia secara garis besar mencakup dua bagian, yakni kimia sebagai proses dan kimia sebagai produk. Kimia sebagai produk meliputi sekumpulan pengetahuan yang terdiri atas fakta-fakta, konsep-konsep, dan prinsip-prinsip ilmu kimia. Sedangkan kimia sebagai proses meliputi keterampilan-keterampilan dan sikap yang dimiliki oleh para ilmuwan untuk memperoleh dan mengembangkan produk kimia (BSNP, 2006).

2.4.1 Teori Atom

Konsep atom pertama kali dikemukakan oleh Democritus. Atom berasal dari kata atomos (dalam bahasa Yunani a = tidak, tomos = dibagi), jadi atom merupakan partikel yang sudah tidak dapat dibagi lagi. Menurut Dalton konsep atom Democritus ini tidak bertentangan dengan Hukum Kekekalan Massa dan Hukum Kekekalan Energi, sehingga Dalton membuat teori tentang atom yang salah satunya adalah materi tersusun atas partikel-partikel terkecil yang tidak dapat dibagi lagi. Tetapi konsep atom Dalton belum memuaskan para ilmuwan pada masa itu. Ditemukannya elektron, proton, neutron, dan radioaktivitas menyebabkan timbulnya teori baru tentang atom. Mulai dari teori atom Thomson, Rutherford, Bohr, dan Mekanika Kuantum.

Menurut Dalton, reaksi kimia dapat terjadi karena penggabungan atom-atom atau pemisahan gabungan atom. Misalnya, logam natrium bersifat netral dan reaktif dengan air dan dapat menimbulkan ledakan. Jika logam natrium direaksikan dengan gas klorin yang bersifat racun dan berbau tajam akan dihasilkan.

Menurut teori Thomson, atom berbentuk bulat di mana muatan listrik positif yang tersebar merata dalam atom dinetralkan oleh elektron-elektron yang berada di antara muatan positif. Elektron-elektron dalam atom diumpamakan seperti butiran kismis dalam roti, maka Teori Atom Thomson juga sering dikenal Teori Atom Roti Kismis.

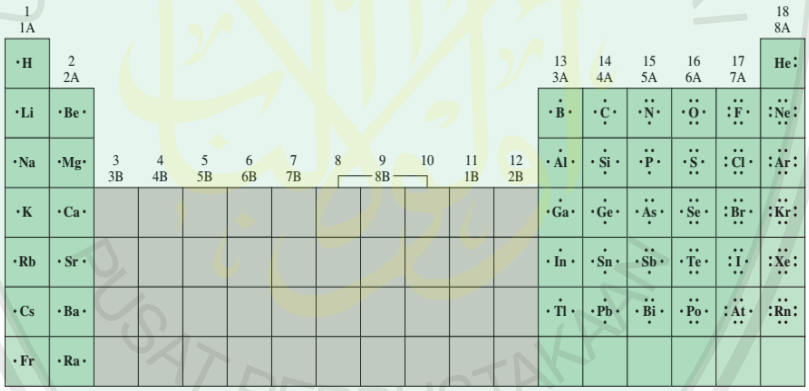
Rutherford mengajukan teori atom sebagai berikut: atom tersusun atas inti atom yang bermuatan positif sebagai pusat massa dan dikelilingi elektron-elektron yang bermuatan negatif. Massa atom berpusat pada inti dan sebagian besar volume atom merupakan ruang hampa. Atom bersifat netral, karena itu jumlah muatan positif dalam atom (proton) harus sama dengan jumlah elektron. Diameter inti atom berkisar 10–15m, sedangkan diameter atom berkisar 10-10m.

Menurut pengamatan Niels Bohr terhadap spektrum atom, adanya spectrum garis menunjukkan bahwa elektron hanya beredar pada lintasan-lintasan dengan energi tertentu. Dengan teori Mekanika Kuantum Planck, Bohr tahun 1913 menyampaikan 2 postulat untuk menjelaskan kestabilan atom.

Dua Postulat Bohr: (1) Elektron mengelilingi inti atom pada lintasan tertentu yang stasioner yang disebut orbit/kulit. Walaupun elektron bergerak cepat tetapi elektron tidak memancarkan atau menyerap energi sehingga energi elektron konstan. Hal ini berarti elektron yang berputar mengelilingi inti atom mempunyai lintasan tetap sehingga elektron tidak jatuh ke inti, (2) Elektron dapat berpindah dari kulit yang satu ke kulit yang lain dengan memancarkan atau menyerap energi. Energi yang dipancarkan atau diserap ketika elektron berpindah-pindah kulit disebut foton (Raymon, 2010).

2.4.2 Simbol Titik Lewis

Gilbert Lewis merumuskan bahwa atom bergabung untuk mencapai konfigurasi elektron yang lebih stabil. Stabilitas maksimum terjadi ketika sebuah atom adalah isoelektrik dengan gas mulia. Ketika atom berinteraksi untuk membentuk suatu ikatan kimia, hanya daerah luar atom yang bersentuhan. Hal ini terjadi karena elektron valensi dari atom. Untuk melacak elektron valensi dalam reaksi kimia dan untuk memastikan bahwa jumlah total elektron tidak berubah, ahli kimia menggunakan sistem titik yang dirancang oleh Lewis disebut simbol titik lewis (Lewis dot symbols). Sebuah simbol titik lewis terdiri dari simbol elemen dan satu titik untuk setiap elektron valensi dalam atom dari elemen. Berikut adalah gambar simbol titik lewis:



Gambar 2.6 Lewis dot symbol (Raymon, 2010)

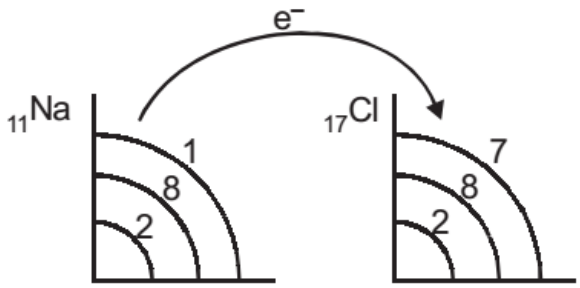
Gambar 2.5 menunjukkan simbol titik lewis untuk unsur-unsur perwakilan dan gas mulia. Kecuali untuk helium, jumlah elektron valensi setiap atom memiliki electron valensi sama dengan jumlah kelompok elemen. Misalnya, Li adalah elemen grup 1A dan memiliki satu titik untuk satu elektron valensi, elemen grup 2A, memiliki dua elektron valensi (dua titik) dan seterusnya. Elemen dalam kelompok yang sama memiliki sejenis elektron terluar gurations kerahasiaan dan karenanya setara Lewis dot simbol (Raymon, 2010).

2.4.3 Ikatan Kimia

Ikatan kimia adalah ikatan yang terjadi antar atom atau antar molekul. Terdapat beberapa jenis ikatan kimia dalam ilmu kimia. Kimiawan Jerman Albrecht Kossel (1853-1927) menganggap kestabilan gas mulia disebabkan konfigurasi elektronnya yang penuh (konfigurasi elektron di kulit terluarnya, kulit valensi terisi penuh). Atom selain gas mulia cenderung mendapatkan muatan listrik (elektron) dari luar atau memberikan muatan listrik ke luar, bergantung apakah jumlah elektron di kulit terluarnya lebih sedikit atau lebih banyak dari atom gas mulia yang terdekat dengannya. Bila suatu atom kehilangan elektron, atom tersebut akan menjadi kation yang memiliki jumlah elektron yang sama dengan gas mulia terdekat, sementara bila atom mendapatkan elektron, atom tersebut akan menjadi anion yang memiliki jumlah elektron yang sama dengan atom gas mulia terdekatnya. Ia menyimpulkan bahwa gaya dorong pembentukan ikatan kimia adalah gaya elektrostatik antara kation dan anion. Ikatan kimia yang dibentuk disebut dengan ikatan ionik (Yashito, 2006).

Ikatan ionik terbentuk akibat adanya melepas atau menerima elektron oleh atom-atom yang berikatan. Atom-atom yang melepas elektron menjadi ion positif (kation) sedang atom-atom yang menerima elektron menjadi ion negatif (anion). Ikatan ion biasanya disebut ikatan elektrovalen. Senyawa yang memiliki ikatan ion disebut senyawa ionik.

Senyawa ionik biasanya terbentuk antara atom-atom unsur logam dan nonlogam. Atom unsur logam cenderung melepas elektron membentuk ion positif, dan atom unsur nonlogam cenderung menangkap elektron membentuk ion negatif. Contoh: NaCl, MgO, CaF2, Li2O.

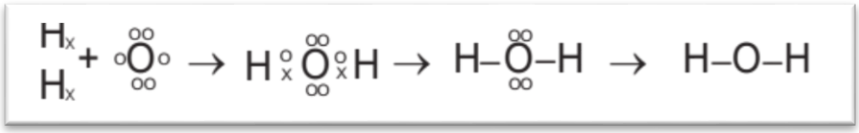


Gambar 2.7 Ikatan ion NaCl (Zumdahl, 2014)

Sifat-sifat fisika senyawa ionik pada umumnya: (1) pada suhu kamar berwujud padat, (2) struktur kristalnya keras tapi rapuh, (3) mempunyai titik didih dan titik leleh tinggi, (4) larut dalam pelarut air tetapi tidak larut dalam pelarut organik, (5) tidak menghantarkan listrik pada fase padat, tetapi pada fase cair (lelehan) dan larutannya menghantarkan listrik.

Ikatan kovalen adalah ikatan yang terjadi karena pemakaian pasangan elektron secara bersama oleh dua atom yang berikatan. Ikatan kovalen terbentuk dari atom-atom unsur yang memiliki afinitas elektron tinggi serta beda keelektronegatifannya lebih kecil dibandingkan ikatan ion.

Ikatan kovalen terjadi karena pemakaian bersama pasangan elektron oleh atom-atom yang berikatan. Pasangan elektron yang dipakai bersama disebut pasangan electron ikatan dan pasangan elektron valensi yang tidak terlibat dalam pembentukan ikatan kovalen disebut pasangan elektron bebas. Ikatan kovalen umumnya terjadi antara atom-atom unsur nonlogam, bisa sejenis (contoh: H2, N2, O2, Cl2, F2, Br2, I2) dan berbeda jenis (contoh: H2O, CO2, dan lain-lain). Senyawa yang hanya mengandung ikatan kovalen disebut senyawa kovalen (Zumdahl, 2014).



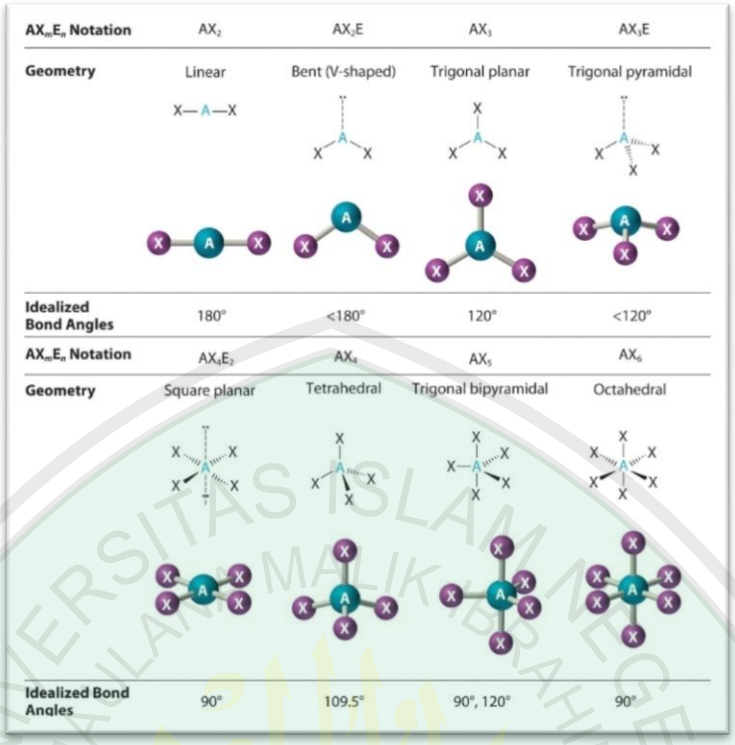
Gambar 2.8 Ikatan kovalen H2O yang di gambarkan dengan simbol titik lewis (Zumdahl, 2014)

2.4.4 Teori VSEPR

Teori VSEPR (Valence shell electron pair repulsion) merupakan penjabaran sederhana dari rumusan Lewis yang berguna untuk memprediksikan bentuk molekul poliatom berdasarkan struktur Lewis-nya. Teori VSEPR pertama kali dikembangkan oleh Nevil Sidgwick dan Herbet Powel pada tahun 1940, dan dikembangkan lebih lanjut oleh Ronald Gillespie dan Ronald Nyholm.

Ide dasar teori VSEPR adalah adanya tolakan antara pasangan elektron sehingga pasangan elektron tersebut akan menempatkan diri pada posisi sejauh mungkin dari pasangan elektron lainnya. Posisi pasangan elektron satu dengan yang lain yang semakin berjauhan akan menyebabkan tolakan antar mereka menjadi semakin kecil. Pada posisi yang paling jauh yang dapat dicapai, tolakan antar pasangan elektron menjadi minimal.

Tolakan antar pasangan elektron terjadi antara pasangan elektron non-ikat yang terlokalisasi pada atom pusat dan elektron ikat secara ikatan koordinasi. Pasangan elektron non-ikat suatu atom tidak digunakan untuk berikatan dengan atom lain, sedangkan pasangan elektron ikat digunakan untuk berikatan dengan atom lain dengan cara pemakaian elektron secara bersama-sama. Teori VSEPR mengasumsikan bahwa masing-masing molekul akan mencapai geometri tertentu sehingga tolakan pasangan antar elektron di kulit valensi menjadi minimal. Geometri tolakan pasangan antar electron sebagai berikut:



Gambar 2.9 Geometri tolakan pasangan antar elektron (Raymon, 2010)

Karena ikatan kovalen terbentuk dari pemakaian pasangan elektron secara bersama oleh dua atom yang berikatan, perubahan sudut ikat menyebabkan perubahan posisi relatif pasangan elektron di sekitar atom pusat. Bila dua elektron saling mendekat, maka akan terjadi gaya tolak menolak di antara kedua elektron tersebut. Konsekuensinya, dalam terminologi energi, kedua elektron akan saling menjauhi (Raymon, 2010).

BAB III

ANALISIS DAN PERANCANGAN SISTEM

3.1 Analisis Sistem

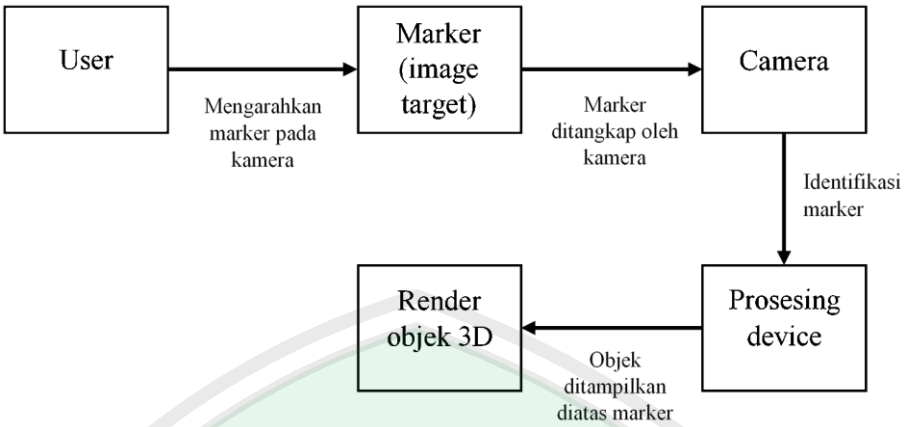
Sistem yang dibuat merupakan media simulasi ikatan kimia berbasis augmented reality dimana objek simbol atom yang berupa gambar sebagai markernya yang kemudian discan dengan aplikasi AR yang dibuat sehingga akan muncul objek atom dalam bentuk 3D.

Aplikasi ini dibuat dengan mengambil latar dilingkungan nyata yang kemudian gabungkan dengan obyek-obyek 3D melalui kamera. Orientasi dan posisi marker akan dideteksi dan ditangkap oleh kamera. Setelah marker terdeteksi oleh kamera, maka akan didapatkan matriks transformasi yang dapat digunakan untuk transformasi seluruh obyek yang ada dalam aplikasi.

Aplikasi Augmented Reality ini menerapkan teori VSEPR yang digunakan untuk memprediksikan bentuk molekul poliatom berdasarkan struktur Lewis atom tersebut. Aplikasi ini menerapkan Natural Features Tracking dengan metode FAST Corner Detection.

3.2 Arsitektur Sistem

Arsitektur sistem yang akan dibangun terdiri dari beberapa komponen yaitu: user, marker, kamera. User adalah pengguna yang akan menggunakan aplikasi, user mengarahkan marker sehingga marker dapat ditangkap oleh kamera. Kemudian dari kamera akan melakukan tracking pada marker untuk mengidentifikasi marker yang digunakan oleh user. Sistem melakukan render objek 3D di atas marker yang telah teridntifikasi. Diagram blok alur sistem seperti pada Gambar 3.1.



Gambar 3.1 Diagram blok arsitektur sistem

Menurut Professor Michael R. Lyu komponen inti dari QCAR Lib adalah sebagai berikut:

1. Camera

   Spesifikasi kamera cukup dengan menggunakankamera tunggal. Kamera mengambil gambar untuk melacak marker dan kemudian melakukan registrasi marker. Pengembang dapat mengaturnya ketika memulai danmenghentikan pengambilan gambar.

2. Image converter

   Gambar akan dikenversi dari format YUV 12 keformat RGB565 untuk OpenGL ES kemudian mengatur pencahayaan untuk pelacakan marker.

3. Tracker

   Menggunakan algoritma computer vision untuk mendeteksi dan melakukan pelacakan objek nyata yang diambil dari kamera. Objek tersebut dievaluasi dan hasilnya akan disimpan yang kemudian akan diakses oleh aplikasi.

4. Renderer

Digunakan untuk melakukan rendering hasil objek yang ditangkap oleh kamera ke video yang dimaksudkan untuk optimasi device.

5. Application code

   Application code melibatkan inisialisasi dari semua komponen diatas. Selama objek yang dikehendaki diubah prosesnya, maka application code harus diubah berdasarkan lokasi objek virtual.

6. Target resources

   Target resource dihasilkan dari target management system. Output yang dihasilkan dari sistem berupa file binary yang menyimpan pola marker dan file konfigurasi XML. Dan semuanya digabung dalam sebuah aplikasi.

3.3 Alur Sistem

Alur sistem mendeskripsikan bagaimana proses augmented reality dari awal inisialisasi, tracking marker, sampai dengan proses rendering objek 3D dengan metode marker yang diterapkan oleh Vuforia. Dalam perancangan aplikasi dengan teknologi AR ini seolah-olah menggabungkan objek virtual dengan objek nyata, dalam hal ini objek virtual berupa objek 3D dan objek nyatanya berupa gambar dengan pola tertentu (marker).

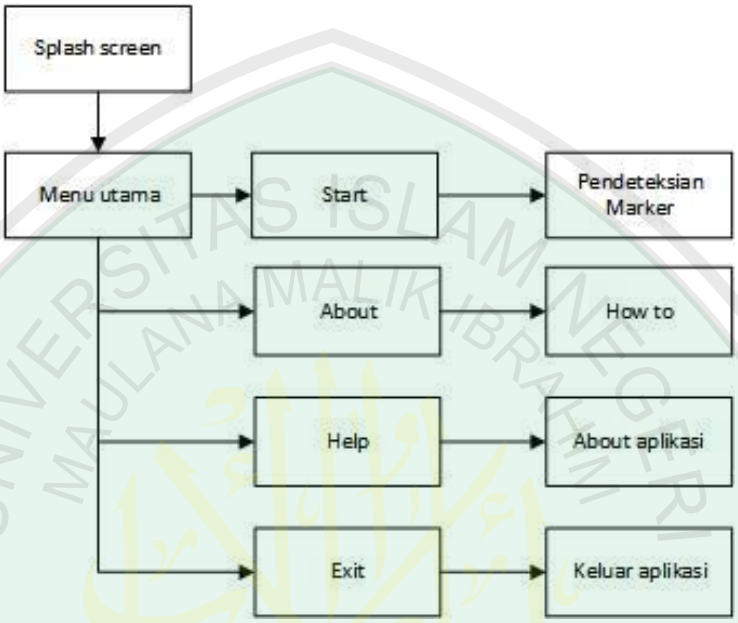
Secara garis besar, dalam perancangan aplikasi AR kimia ini terdapat tiga bagian utama yaitu traking marker, inisialisasi dan rendering objek. Diagram blok alur sistem dapat dilihat pada Gambar 3.2.

Gambar 3.2 Alur sistem

3.4 Perancangan sistem

3.4.1 Flowchart

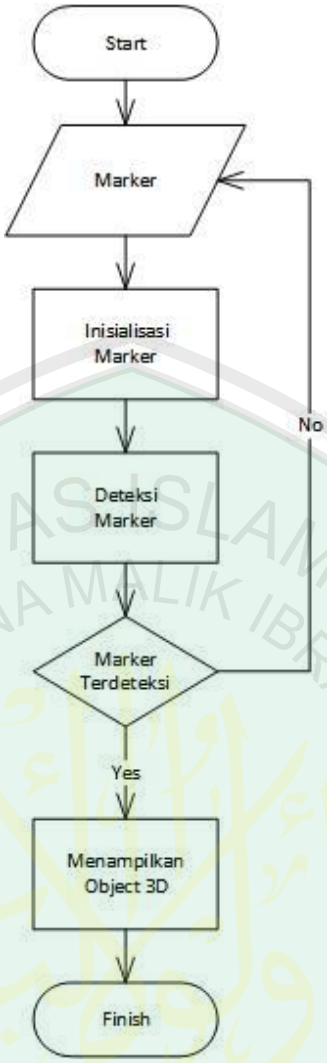
Alur kerja aplikasi augmented reality yang akan dibagun secara umum ditunjukkan dalam Gambar 3.2.



Gambar 3.3 Blok diagram aplikasi AR

Pada Gambar 3.3, setelah tampilan splash screen, aplikasi akan menampilkan menu utama. Pada menu utama tersebut terdapat pilihan menu Start, menu Help, menu About dan Exit. Ketika memilih Start, maka aplikasi akan mengaktifkan kamera untuk memulai proses pendeteksian marker. Pada menu Help, ketika user memilih menu Help maka aplikasi akan menampilkan informasi cara penggunaan aplikasi. Pada menu About, ketika user memilih menu About maka aplikasi akan menampilkan informasi yang berkaitan dengan aplikasi.

Alur proses pendeteksian marker pada aplikasi AR ini dapat dilihat pada Gambar 3.4.

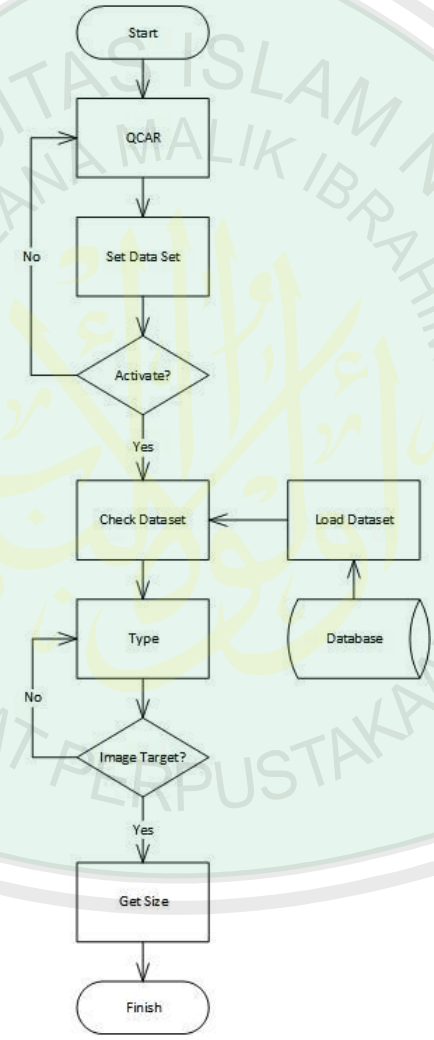


Gambar 3.4 Alur proses pendeteksian marker

Proses pendeteksian marker dimulai dengan user mengarahkan marker pada kamera. Kemudian kamera akan mendeteksi marker tersebut. Pendeteksian marker bergantung pada beberapa hal, yaitu intensitas cahaya, jarak marker dengan kamera, oklusi (pendeteksian marker terhalang sesuatu), dan resolusi kamera. Jika marker tidak terdeteksi, maka user harus mengatur marker dan mengarahkah kembali pada kamera. Jika marker telah terdeteksi, maka aplikasi akan menampilkan objek 3D yang sesuai dengan marker.

Aplikasi AR ini memerlukan marker untuk menampilkan objek 3D yang telah dibuat. Tahap pertama dalam inisialisasi marker adalah mengupload marker

pada developer vuforia. Marker yang akan diupload harus diatur ukurannya sesuai dengan ketentuan dari vuforia. File marker harus bertipe .jpg atau .png. Setelah marker berhasil diupload, proses selanjutnya adalah mendownload dataset dari marker tersebut. Dataset ini nantinya akan diimport ke dalam unity, untuk dipasangkan dengan objek 3D yang telah dibuat sebelumnya. Proses dapat dilihat pada Gambar 3.5.



Gambar 3.5 Flowchart inisialisasi marker

3.4.2 Algoritma FAST Corner detection

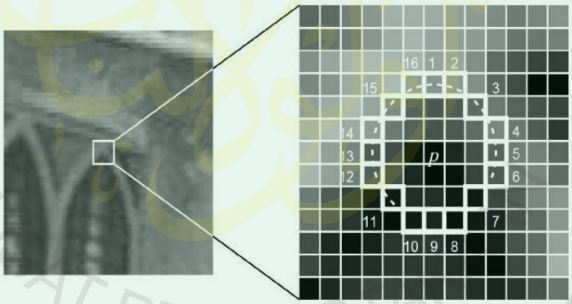
FAST corner detection dimulai dengan menentukan suatu titik p pada koordinat (x_p , y_p) pada citra dan membandingkan intensitas titik p dengan 4 titik di

sekitarnya. Titik pertama terletak pada koordinat (x , y_{p-3}), titik kedua terletak pada koordinat (x_{p+3} , y), titik ketiga terletak pada koordinat (x , y_{p+3}), dan titik keempat terletak pada koordinat (x_{p-3}, y).

Jika nilai intensitas di titik *p* bernilai lebih besar atau lebih kecil daripada intensitas sedikitnya tiga titik di sekitarnya ditambah dengan suatu intensitas batas ambang (Threshold), maka dapat dikatakan bahwa titik *p* adalah suatu sudut. Setelah itu titik *p* akan digeser ke posisi (x_{p+1}, y_p) dan melakukan intensitas keempat titik disekitarnya lagi. Iterasi ini terus dilakukan sampai semua titik pada citra sudah dibandingkan.

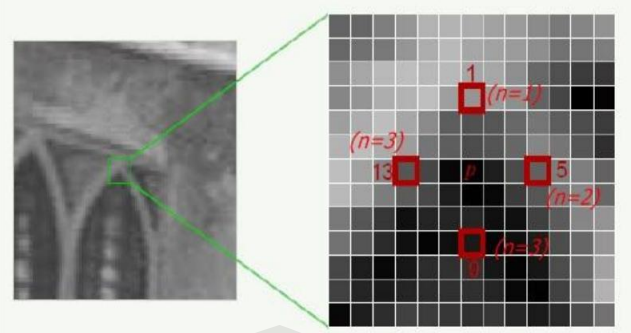
Algoritma FAST corner detectesion bekerja pada suatu citra sebagai berikut:

1. Tentukan sebuah titik p pada citra dengan posisi awal (x_p,y_p)



Gambar 3.6 Titik awal p

2. Tentukan keempat titik. Titik pertama (*n* = 1) terletak pada koordinat (x_p,y_{p+3}), titik kedua (*n* = 2) terletak pada koordinat (x_{p+3}, y_p), titik ketiga terletak pada koordinat (*n* = 3) terletak pada koordinat (x_p, y_{p-3}), titik keempat (*n* = 4) terletak pada koordinat (x_{p-3},y_p).



Gambar 3.7 Titik *p* pada koordinat *n* = 1, *n* = 2, *n* = 3,dan *n* = 4

3. Bandingkan intensitas titik pusat p dengan keempat titik disekitar. Jika terdapat paling sedikit 3 titik yang memenuhi syarat berikut, maka titik pusat p adalah titik sudut.

$$C_p = \begin{cases} 1, & I_n < I_p - t \text{ or } I_n > I_p + t \\ 0, & otherwise \end{cases}$$

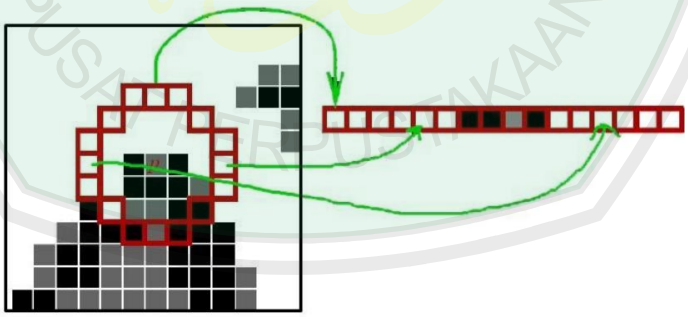
Dengan,

C_p : Keputusan titik p sebagai sudut, nilai 1 menunjukan bahwa titik merupakan suatu sudut, dan nilai 0 menunjukan bahwa titik bukanlah sudut

I_n : Nilai intensitas piksel ke-*n*

I_p : Nilai intensitas titik *p*

t : batas ambang nilai intensitas yang ditoleransi



Gambar 3.8 Perbandingan intensitas pada seluruh titik

4. Ulangi proses sampai seluruh titik pada citra sudah dibandingkan intensitasnya.

Vuforia menggunakan algoritma FAST Corner detection untuk mendefinisikan seberapa baik gambar dapat dideteksi dan dilacak menggunakan

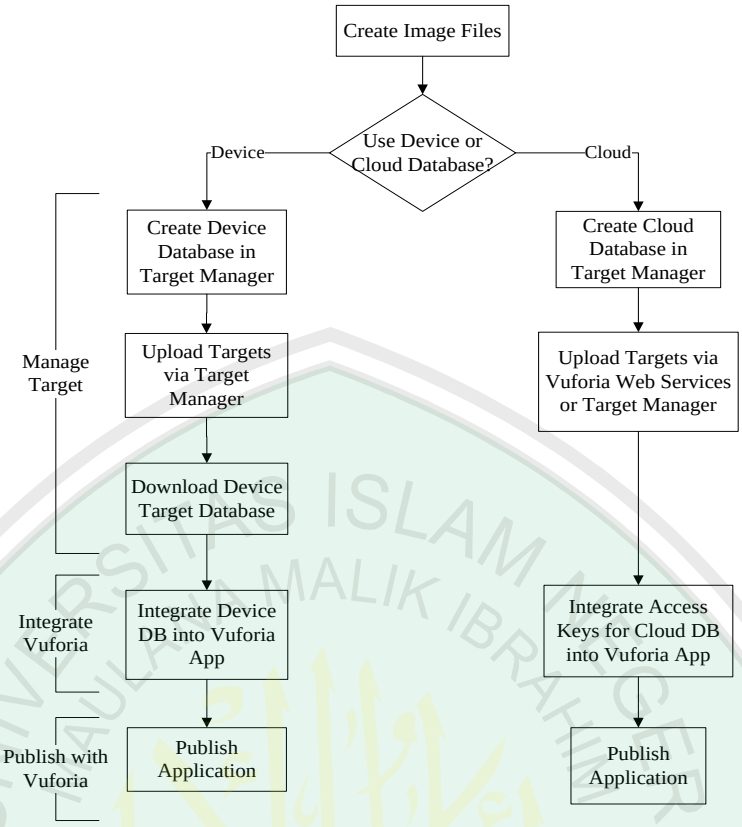
Vuforia SDK. Peringkat ini ditampilkan dalam Target Manager untuk setiap target yang diupload melalui web API. Rating augmentable dapat berkisar dari 0 sampai 5 untuk setiap gambar yang diberikan. Semakin tinggi rating augmentable dari target gambar, semakin kuat kemampuan deteksi dan pelacakan yang dikandungnya. Sebuah rating dari nol menunjukkan bahwa target tidak dilacak sama sekali oleh sistem Augmented Reality, sedangkan rating bintang 5 menunjukkan bahwa sebuah gambar dengan mudah dilacak oleh sistem Augmented Reality.

Baik atau buruknya kontras dapat mempengaruhi terdeteksi fitur, dengan meningkatkan kontras gambar secara umum atau memilih gambar dengan detail rincian bulat, kabur dan gambar yang dikompresi berlebihan maka tidak akan memberikan kekayaan yang cukup rinci untuk dideteksi dan dilacak dengan benar.(http://developer.vuforia.com/library/articles/training)

3.4.3 Pembuatan Marker

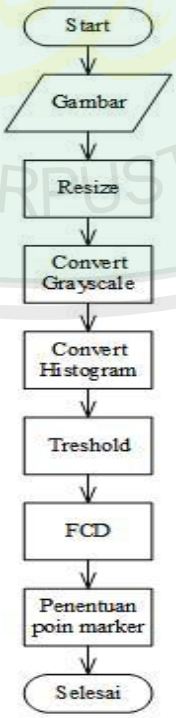
Marker yang digunakan pada aplikasi ini dibuat menggunakan Adobe Photoshop. Proses pembuatan gambar menjadi marker dilakukan oleh Vuforia, yaitu dengan cara mengupload gambar marker pada laman web developer Vuforia. Setelah gambar di convert menghasilkan file dengan format .unitypackage. File tersebut kemudian dijadikan masukan pada Engin Unity3D untuk mendeteksi gambar yang dijadikan marker. Format gambar yang dapat dijadikan marker adalah harus berformat .jpg atau .png, ukuran gambar tidak lebih dari 2 MB, warnanya harus RGB atau grayscale (bukan CMYK), serta gambar harus memiliki feature. Feature dalam gambar artinya adalah gambar tersebut memiliki sudut.

Gambar 3.9 merupakan workflow dari developer Vuforia dalam proses pembuatan marker.



Gambar 3.9 Workflow pembuatan marker developer vuforia (http://developer.vuforia.com).

Flow chart alur proses pembuatan marker ditunjukan pada Gambar 3.10.



Gambar 3.10 Flowchart alur proses pembuatan marker

Pada Gambar 3.10, terdapat 5 proses yang dilakukan untuk membuat marker sebagai berikut:

1. Resize

Tahap pertama yang dilakukan adalah merubah ukuran gambar yang akan dijadikan marker. Gambar akan diperkecil ke ukuran 320 x 320 pixel. Psudocode algoritma resize ditunjukkan pada Gambar 3.11.

```
image,imgx,imgy
Max_x <- 320 /*maksimal nilai lebar gambar*/
Max_y <- 320 /*maksimal nilai panjang gambar*/

if(imgx >= Max_x)
    Scale = Max_x/imgx
    new_imgx = imgx * scale
    new_imgy = imgy * scale
endif

if(new_imgy > Max_y)
    Scale = Max_y/new_imgy
    new_imgx = new_imgx * scale
    new_imgy = new_imgy*scale
endif

if(imgx < Max_x)
    new_imgx = imgx
endif

if(imgy < Max_y)
    new_imgy = imgy
endif
```

Gambar 3.11 Psudocode resize gambar

2. Convert grayscale

Convert grayscale merupakan proses mengubah gambar menjadi monokrom dengan nilai intensitas 0 untuk hitam, 255 untuk putih, dan abu-abu untuk nilai yang berada diantara nilai 0 sampai 255. Secara garis besar dapat dirumuskan *new pixel = (Red + Green +Blue)/3*. Psudocode algoritma convert grayscale ditunjukkan pada Gambar 3.12.

```
image,x,y

for(i=0 ; i<x; i++)
    for(j=0; j<y; j++)
        Getpixel(i,j)
        int grayscale = (int)((originalcolor.R)+(originalcollor.G)+(originalcolor.B)/3)
        Newimage(i,j,grayscale)
    endfor
endfor
```

Gambar 3.12 Psudocode algoritma convert grayscale

3. Convert histogram

Convert histogram merupakan proses pemerataan derajat keabuan pada gambar. Psudocode algoritma convert histogram ditunjukkan pada Gambar 3.13.

```
A,int N,int M /*A=array gambar, N= Panjang gambar, M=Lebar gambar*/

/*inisialisasi Hist[0,255] dengan 0*/
for(i=0;i<=255;i++)
    Hist[i]=0
endfor

/*mencacah frekuensi kemunculan*/
for(i=0;i<=255;i++)
    for (j=0;j<=N-1;j++)
        for(k=0,k<=M-1;k++)
            if(A[j,i]==j-1)
                Hist[j]=Hist[j]+1
            endif
        endfor
    endfor
endfor
```

Gambar 3.13 Psudocode algoritma convert histogram

4. Threshold

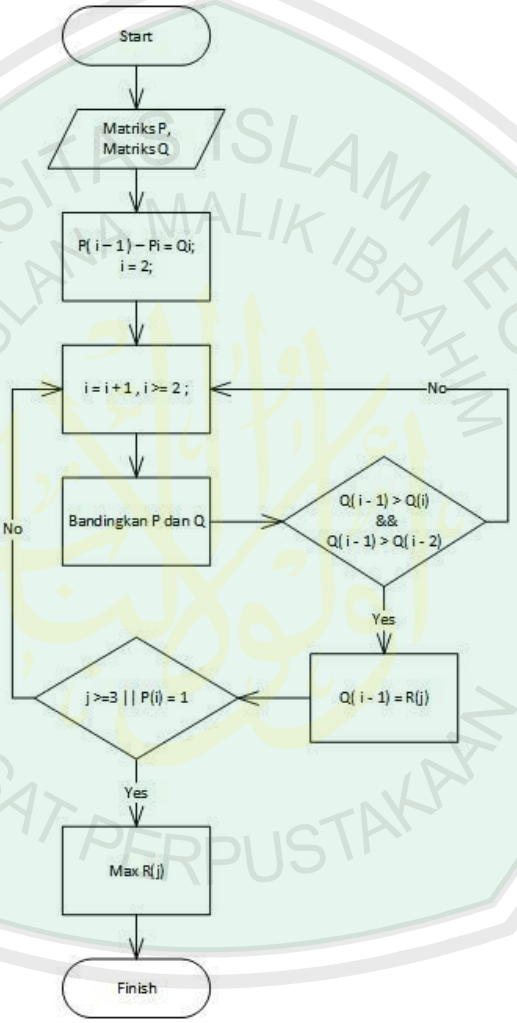
Dalam Library Vuforia sebelum objek diconvert kedalam point-point marker yang dapat menghasilkan titik sudut marker, marker diubah kedalam bentuk threshold. Proses threshold ini yang menjadi acuan untuk proses selanjutnya, yaitu proses penentu titik marker. Psudocode algoritma convert histogram ditunjukkan pada Gambar 3.14.

```
Tmean = 0 /*nilai awal t*/

for(i =0;i<level;i++)
    for(j=0;j<panjang_pixel-1;j++)
        for(k=0;k<lebar_pixel-1;k++)
            N= image[i],[y]
            if(n<=T)
                new_x=0
                new_y=0
                t=k;
            else
                new_x=255
                new_y=255
            endif
        endfor
    endfor
endfor
```

Gambar 3.14 Psudocode algoritma convert histogram

5. FAST corner detection

Gambar hasil proses threshold, kemudian diproses untuk menentukan letak titik point pada marker dan pencocokan pola menggunakan metode Fast Corner Detection. Proses ini bertujuan untuk menetukan banyaknya jumlah dari titik point. Flowchart FAST corner detection ditunjukkan pada Gambar 3.15.



Gambar 3.15 Flowchart FAST corner detection

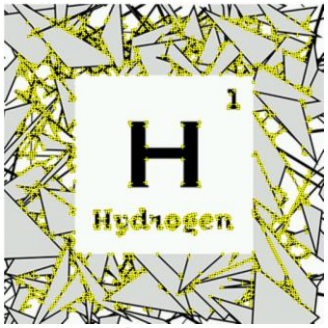
Proses corner detection dimulai dengan menentukan suatu titik p pada koordinat (x_p , y_p) pada citra dan membandingkan intensitas titik p dengan 4 titik di sekitarnya. Kemudian dilakukan perbandingan nilai intensitas di titik *p* bernilai lebih besar atau lebih kecil daripada intensitas sedikitnya tiga titik di sekitarnya

ditambah dengan suatu intensitas batas ambang (Threshold), maka dapat dikatakan bahwa titik *p* adalah sudut. Setelah itu titik *p* akan digeser ke posisi (x_{p+1}, y_p) dan melakukan intensitas keempat titik disekitarnya lagi. Iterasi ini terus dilakukan sampai semua titik pada citra sudah dibandingkan.

Psudocode fungsi dari metode Fast Corner Detection yang digunakan untuk menentukan point-point marker pada gambar ditunjukkan pada Gambar 3.15. Contoh hasil penentuan titik point ditunjukkan pada Gambar 3.16.

```
clear all;
close all;
clc;
/* membaca gambar */
filename='h.jpg';
I=imread(filename);
[x,y,z]=size(I);

if(z>1)
    Ig=rgb2gray(I);
else
    Ig=I;
endif

/* Proses Corner Detection*/
Ig=double(Ig);
sigma=2; rad=5;

/* Compute x and y derivatives of image */
dx=[-1 0 1; -1 0 1; -1 0 1];
dy=dx';
Ix=conv2(Ig,dx,'same');
Iy=conv2(Ig,dy,'same');

/* Compute products of derivatives at every pixel*/
Ix2=Ix.*Ix;
Iy2=Iy.*Iy;
Ixy=Ix.*Iy;

/* Compute the sums of the product of derivatives at each pixel*/
h=fspecial('gaussian',max(1,fix(6*sigma)),sigma);
Sx2=conv2(Ix2,h,'same');
Sy2=conv2(Iy2,h,'same');
Sxy=conv2(Ixy,h,'same');


/* Compute the response of the detector at each pixel*/
k=0.04; % k=0.04 - 0.06
R=zeros(size(Ig));

/* Compute the response of detector at each pixel*/
R=(Sx2.*Sy2 - Sxy.^2)./(Sx2 + Sy2 + eps);
ambang=12;

/* Threshold on value of R. Compute nonmax suppresion Size of mask*/
sze = 2*rad+1;

/* Grey-scale dilate*/
mx = ordfilt2(R,sze^2,ones(sze));

/* Find max*/
R = (R==mx)&(R>ambang);
[row,col] = find(R);
imshow(I);
hold on;
points=[col row];
plot(points(:,1),points(:,2),'rp');
title('Detected Corner');
```

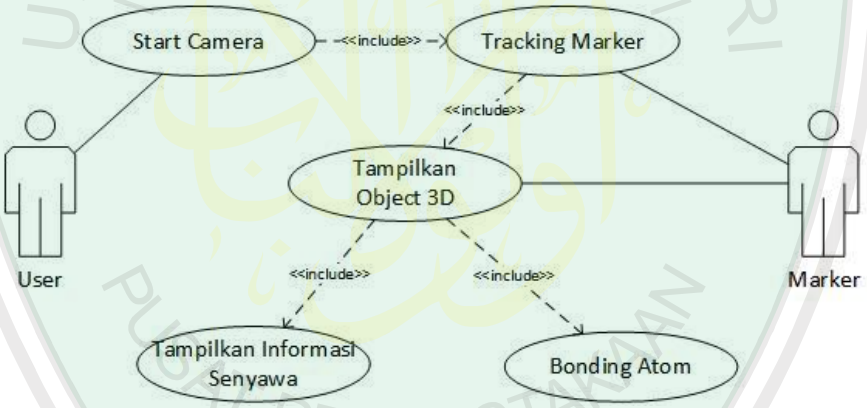
Gambar 3.16 Psudocode Fast Corner Detection



Gambar 3.17 Contoh hasil penentuan titik point

3.5 Usecase Diagram

Usecase diagram merupakan model untuk mendeskripsikan hubungan-hubungan yang terjadi antar aktor dengan aktivitas yang terdapat pada sistem. Pada sistem ini terdapat aktor dan pengguna sistem, yaitu user dan marker. Usecase ditunjukkan pada Gambar 3.17.



Gambar 3.18 Usecase diagaram Apliaksi

3.5.1 Definisi Usecase

Usecase diagram merupakan konstruksi untuk mendeskripsikan hubungan-hubungan yang terjadi antara aktor dengan aktivitas yang terdapat pada aplikasi. Tujuan pemodelan Usecase diantaranya adalah mendefinisikan kebutuhan fungsional dan operasional sistem. Definisi dari setiap use case yang ada pada aplikasi dapat dilihat pada tabel 3.1.

Tabel 3.1 Deskripsi Usecase

Nama Usecase	Deskripsi
Start	Fungsionalitas untuk memulai mengambil gambar
Tracking Marker	Fungsional untuk membandingkan gambar yang berasal dari kamera dengan gambar yang ada pada file tracking.
Tampilkan Objek 3D	Fungsionalitas untuk menampilkan objek 3D
Bonding Atom	Fungsionalitas untuk melakukan ikatan antar atom
Tampilkan Informasi Senyawa	Fungsionalitas untuk menampilkan penjelasan dari senyawa yang terbentuk.

3.5.2 Skenario Usecase

Skenario Usecase menjelaskan urutan-urutan proses yang dilakukan oleh aktor terhadap sistem maupun yang dilakukan oleh sistem terhadap aktor.

1. Skenario Start

Tabel 3.2 Skenario Start

Nama Usecase	Start	
Tujuan dalam konteks	Sistem dapat mendeteksi marker	
Kondisi sebelum	Sistem menampilkan Main menu	
Keberhasilan kondisi akhir	Sistem menjalankan kamera AR	
Kegagalan kondisi akhir	Sistem tidak dapat menjalankan kamera AR	
Aktor utama	User	
Include	Tracking Marker	
Pemicu	User memilih menu start	
Aliran Utama	**Langkah**	**Aksi**
	1	Memilih menu start
	2	User menunjukkan marker

	3 (Include: Tracking Marker)	Sistem mendeteksi marker
	4	Sistem me-*render* objek 3D
	5	Sistem menampilkan objek 3D sesuai marker yang terdeteksi
Perluasan	**Langkah**	**Aksi Percabangan**
	4.1	Sistem tidak dapat mengenali marker
	5.1	Sistem tidak dapat me-*render* objek 3D sesuai dengan marker

2. Skenario Tracking Marker

Tabel 3.3 Skenario tracking marker

Nama Usecase	**Tracking Marker**	
Tujuan dalam konteks	Sistem dapat mengenali marker yang terdeteksi	
Kondisi sebelum	Marker belum terdeteksi	
Keberhasilan kondisi akhir	Sistem dapat mengenali marker	
Kegagalan kondisi akhir	Sistem tidak dapat mengenali marker	
Aktor utama	User	
Aktor kedua	Marker	
Include	Menampilkan objek 3D	
Pemicu	Kamera dapat mendeteksi keberadaan marker	
Aliran Utama	**Langkah**	**Aksi**
	1	User mengarahkan marker
	2	Sistem menginisialisasi marker
	3	Sistem mengidentifikasi marker
	4	Sistem dapat melakukan proses grayscale
	5	Sistem dapat melakukan proses pendeteksian fitur

	6	Sistem mampu mendeteksi marker sesuai dengan library
Perluasan	**Langkah**	**Aksi Percabangan**
	3.1	Sistem tidak mengidentifikasi marker
	6.1	Sistem tidak dapat mendeteksi marker

3. Skenario Tampilkan Object 3D

Tabel 3.4 Skenario tampilkan object 3d

Nama Usecase	**Tampilkan Objek 3D**	
Tujuan dalam konteks	Sistem menampilkan objek 3D	
Kondisi sebelum	Marker yang terdeteksi dapat dikenali	
Keberhasilan kondisi akhir	Sistem menampilkan objek 3D sesuai marker yang dideteksi	
Kegagalan kondisi akhir	Sistem tidak dapat menampilkan objek 3D	
Aktor utama	User	
Aktor kedua	Marker	
Include	Bonding Atom	
Extend	Tampilkan Informasi Senyawa	
Pemicu	Marker dapat dikenali dengan baik dan sesuai dengan pola marker dataset	
Aliran Utama	**Langkah**	**Aksi**
	1	User mengarahkan marker
	2	Kamera membaca marker
	3	Sistem mengenali marker
	4	Sistem render Objek 3D
	5	Objek 3D ditampilkan tetap di atas marker
Perluasan	**Langkah**	**Aksi Percabangan**

	3.1	Sistem tidak dapat mengenali pola marker
	3.3	Sistem tidak menampilkan objek 3D

4. Skenario Bonding Atom

Tabel 3.5 Skenario bonding atom

Nama Use Case	**Bonding Atom**	
Tujuan dalam konteks	Sistem melakukan ikatan antar atom	
Kondisi sebelum	Objek 3D sudah ditampilkan	
Keberhasilan kondisi akhir	Sistem dapat melakukan ikatan pada object 3D atom yang ditampilkan	
Kegagalan kondisi akhir	Sistem tidak melakukan ikatan pada object 3D atom yang ditampilkan	
Aktor utama	User	
Aktor kedua	Marker	
Pemicu	Objek 3D sudah dapat ditampilkan dan sistem melakukan ikatan antar atom	
Aliran Utama	**Langkah**	**Aksi**
	1	User melakukan trigger pada marker bond
	2	Sistem melakukan ikatan pada object 3D atom
Perluasan	**Langkah**	**Aksi Percabangan**
	2.1	Sistem tidak dapat melakukan ikatan pada object 3D atom

5. Skenario Tampilkan Informasi Senyawa

Tabel 3.6 Tampilkan informasi senyawa

Nama Use Case	**Tampilkan Informasi Senyawa**
Tujuan dalam konteks	Sistem menampikan informasi senyawa

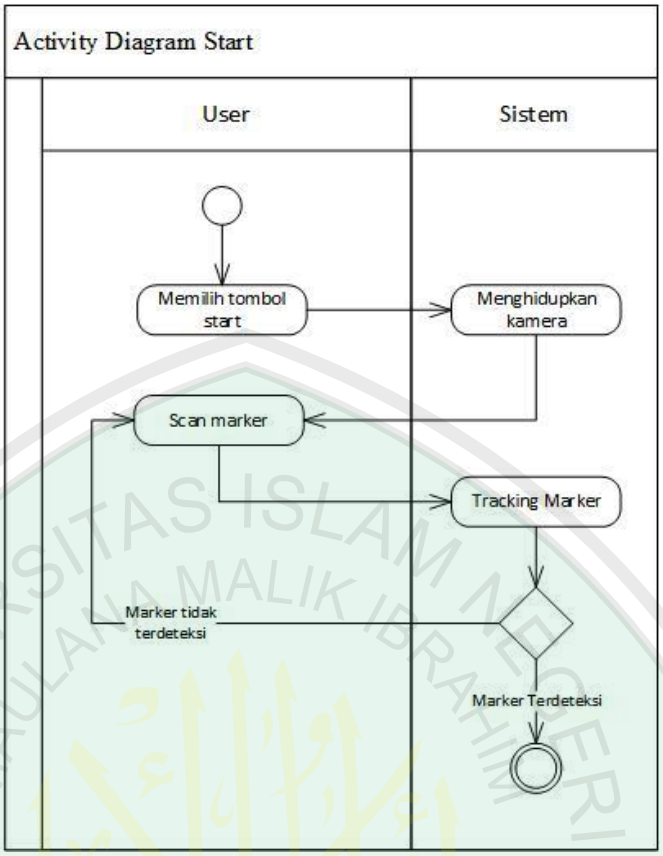
Kondisi sebelum	Objek 3D sudah ditampilkan	
Keberhasilan kondisi akhir	Sistem dapat menampilkan informasi senyawa yang sudah terbentuk	
Kegagalan kondisi akhir	Sistem tidak menampilkan informasi senyawa yang sudah terbentuk	
Aktor utama	User	
Aktor kedua	Marker info	
Pemicu	Objek 3D sudah dapat ditampilkan dan user melakukan trigger pada marker bond	
Aliran Utama	**Langkah**	**Aksi**
	1	User meakukan trigger pada marker bond
	2	Sistem menampilkan informasi senyawa yang sudah terbentuk
Perluasan	**Langkah**	**Aksi Percabangan**
	2.1	Sistem tidak dapat menampilkan informasi senyawa yang sudah terbentuk

3.5.3 Activity Diagram

Activity diagram dari aplikasi yang akan dibangun adalah sebagai berikut:

1. Activity Diagram Start

Activity diagram Start merupakan aktifitas dari menu pilihan start untuk menjalankan kamera AR. Activity diagram Start dapat dilihat pada Gambar 3.19.

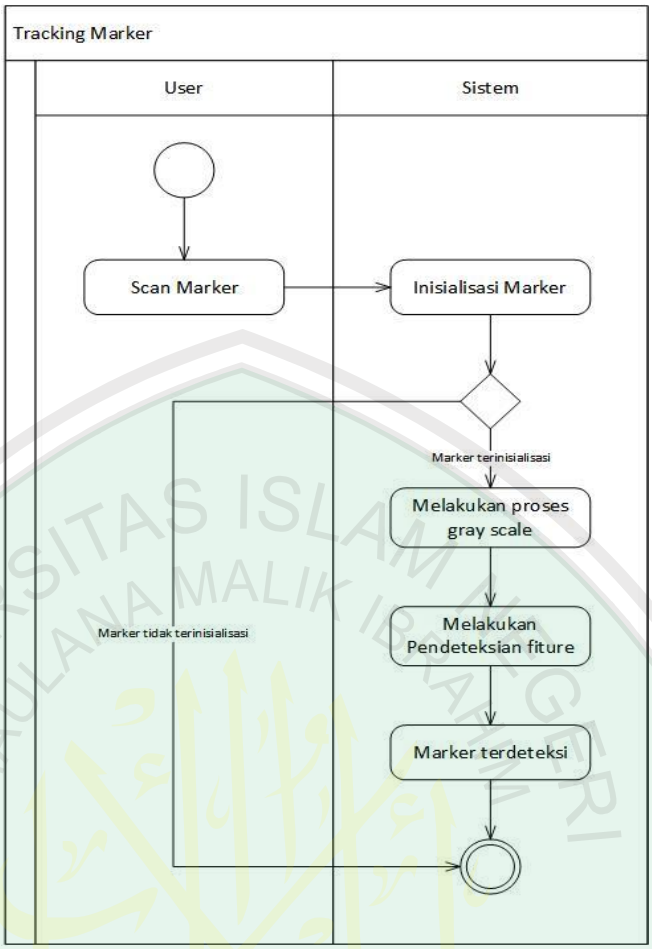


Gambar 3.19 Activity diagram start

Pada Gambar 3.19 ketika user memilih tombol start, sistem akan menghidupkan kamera, kemudian user menunjukkan marker pada kamera untuk dideteksi/scan, kemudian sistem melakukan tracking marker. Jika marker tidak ada maka akan kembali ke proses menujukkan marker oleh user, jika terdeteksi maka akan diteruskan ke proses selanjutnya.

2. Activity Diagram Tracking Marker

Activity diagram tracking marker merupakan alur aktifitas pendeteksian marker. Activity diagram tracking marker dapat dilihat pada Gambar 3.20.

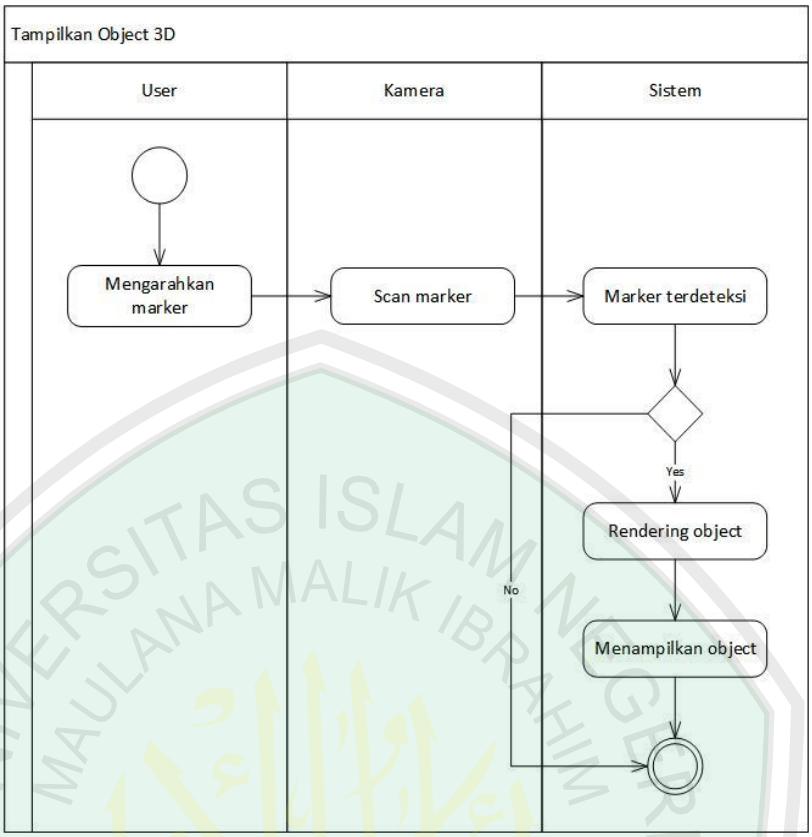


Gambar 3.20 Activity diagram tracking marker

Pada Gambar 3.20 ketika user menunjukan marker untuk dideteksi/scan, sistem melakukan inisialisasi pada marker. Jika marker terinisialisasi maka sistem melakukan pendetaksian fiture. Jika marker tidak terinisialisasi maka akan diteruskan ke proses berikutnya.

3. Activity Diagram Tampilkan Object

Activity diagram tampilkan object merupakan alur aktifitas untuk menampilkan objek 3D. Activity diagram tampilkan object dapat dilihat pada Gambar 3.20.

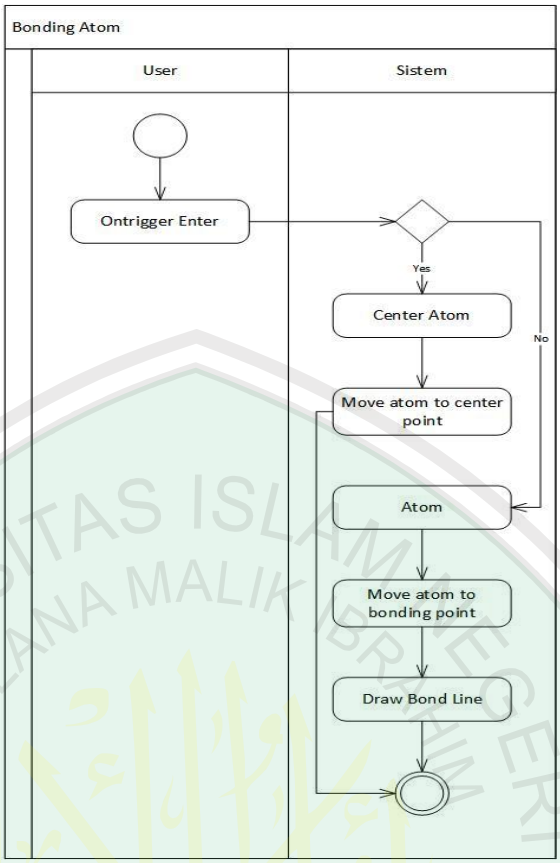


Gambar 3.21 Activity diagram tampilkan objek

Pada Gambar 3.20 user mengarahkan marker pada kemera, kemudian sistem melakukan deteksi marker. Jika marker terdeteksi sistem melakukan rendering, kemudian sistem akan menampilkan objek 3d pada marker.

4. Activity Diagram Bonding Atom

Activity diagram bonding atom merupakan alur aktifitas untuk melakukan ikatan pada object 3D atom.

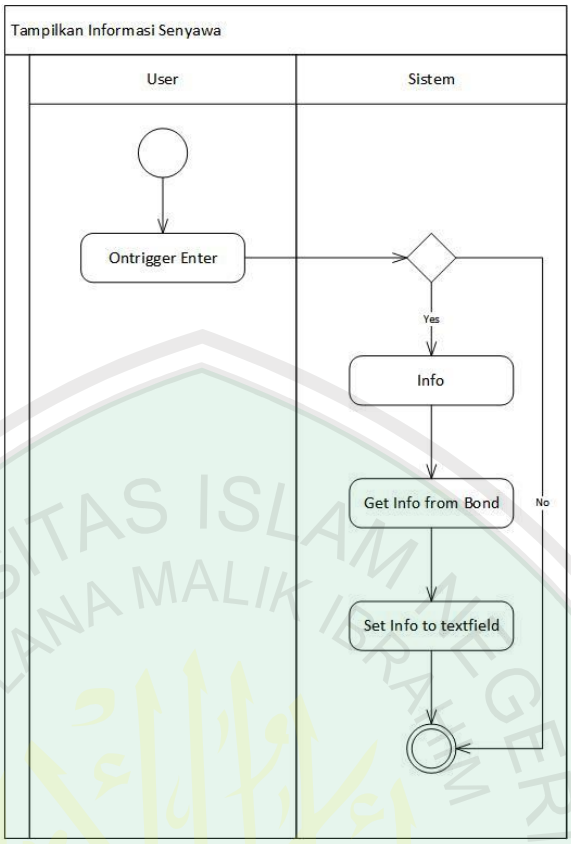


Gambar 3.22 Activity diagram bonding atom

Pada Gambar 3.22 user mendekatkan marker atom pada marker bond. OnTriggerEnter adalah pendeteksian sentuhan dari collider objek 3d, ketika marker dari atom ter-trigger maka akan ditentukan apakah merupakan certer atom atau atom yang diikat. Jika merupakan center atom maka sistem akan memindahkan objek atom menuju center point. Jika atom maka sistem akan memindahkannya menuju bonding point, kemudian sistem menggambar garis ikatan dari center atom menuju atom yang diikat.

5. Activity Diagram Tampilkan Informasi Senyawa

Activity diagram tampilkan informasi senyawa merupakan alur aktifitas untuk menampilkan informasi senyawa yang sudah terbentuk dari proses ikatan atom.



Gambar 3.23 Activity diagram dari tampilkan informasi

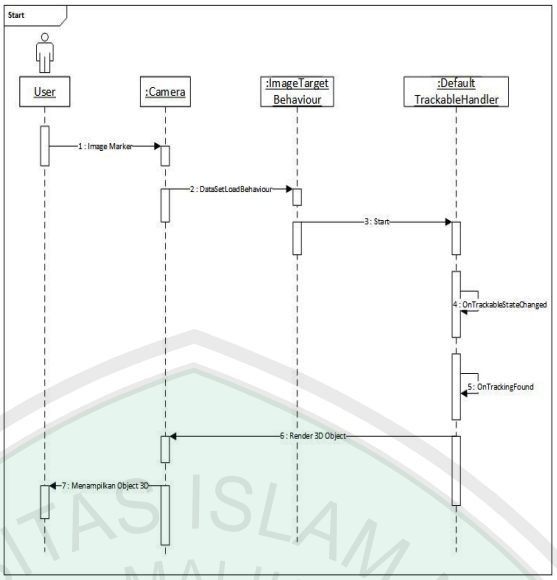
Pada Gambar 3.23 user mendekatkan marker hint pada marker bond. Ketika marker hint ter-trigger maka sistem akan megambil data info dari objek bond, kemudian memasukkannya pada textfield yang ada pada objek marker hint.

3.5.4 Sequence Diagram

Sequence diagram menggambarkan interaksi antara masing-masing objek pada setiap usecase dalam urutan waktu. Interaksi berupa pengiriman serangkaian data antar objek-objek yang saling berinteraksi. Sequence diagram dari aplikasi yang akan dibangun adalah sebagai berikut:

1. Sequence Diagram Start

   Sequence diagram start dapat dilihat pada Gambar 3.24.

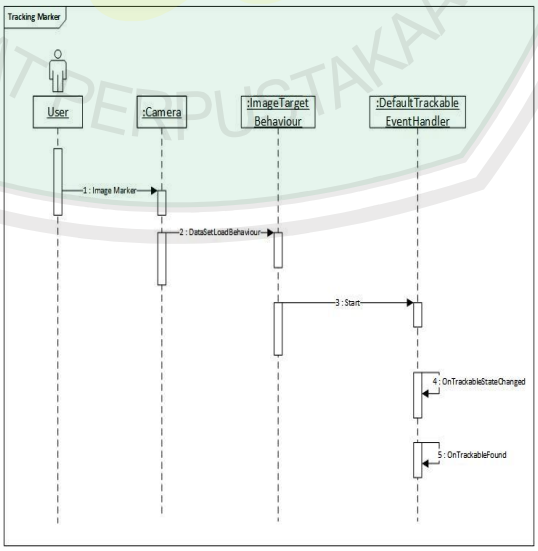


Gambar 3.24 Sequence diagram start

Pada Gambar 3.24 dijelaskan bahwa user menscan marker dengan kamera kemudian dideteksi oleh sistem. Class ImageTargetBehaviour mendeteksi marker yang ada didepan kamera kemudian memanggil class DefaultTrackableBehaviour, setelah itu akan dimunculkan objek 3D.

2. Sequence Diagram Tracking Marker

Sequence diagram tracking marker dapat dilihat pada Gambar 3.25.

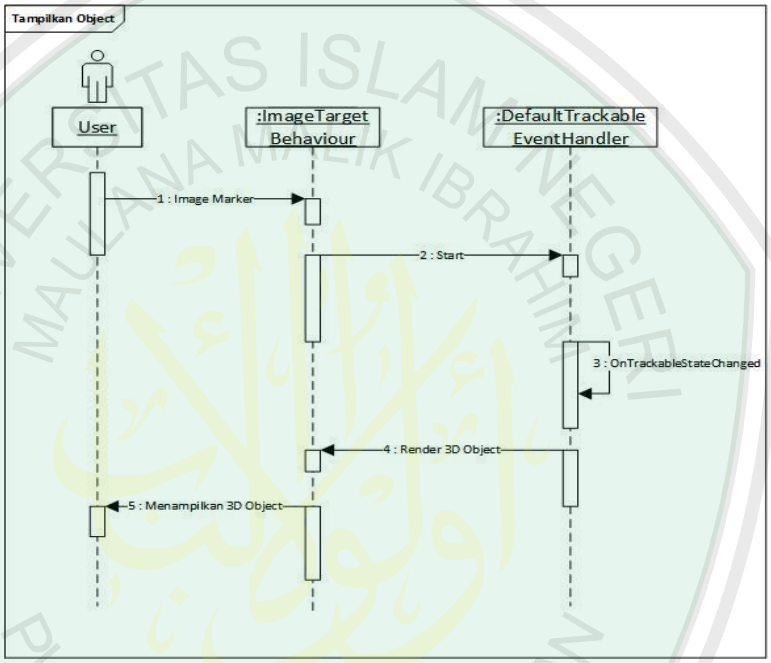


Gambar 3.25 Sequence diagram tracking marker

Pada Gambar 3.25 dijelaskan bahwa user menscan marker dengan kamera kemudian dideteksi oleh sistem. Class ImageTargetBehaviour mendeteksi marker yang ada didepan kamera kemudian memanggil class DefaultTrackable EventHandler untuk men-*tracking* marker.

3. Sequence Diagram Tampilkan Object

Sequence diagram tampilkan object dapat dilihat pada Gambar 3.26.

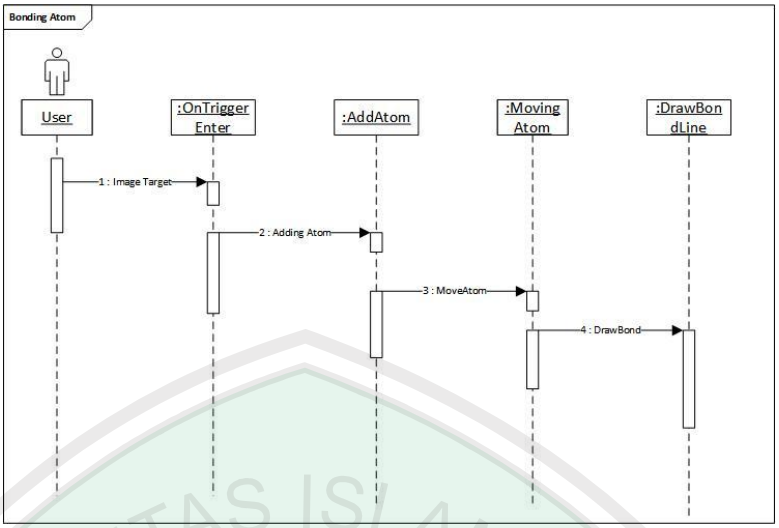


Gambar 3.26 Sequence diagram tampilkan object

Pada Gambar 3.26 dijelaskan bahwa user menginputkan marker kemudian dideteksi oleh sistem. Class ImageTargetBehaviour mendeteksi marker yang ada didepan kamera kemudian memanggil class DefaultTrackableEventHandler, setelah itu akan dimunculkan objek 3D.

4. Sequence Diagram Bonding Atom

Sequence diagram bonding atom dapat dilihat pada Gambar 3.27.

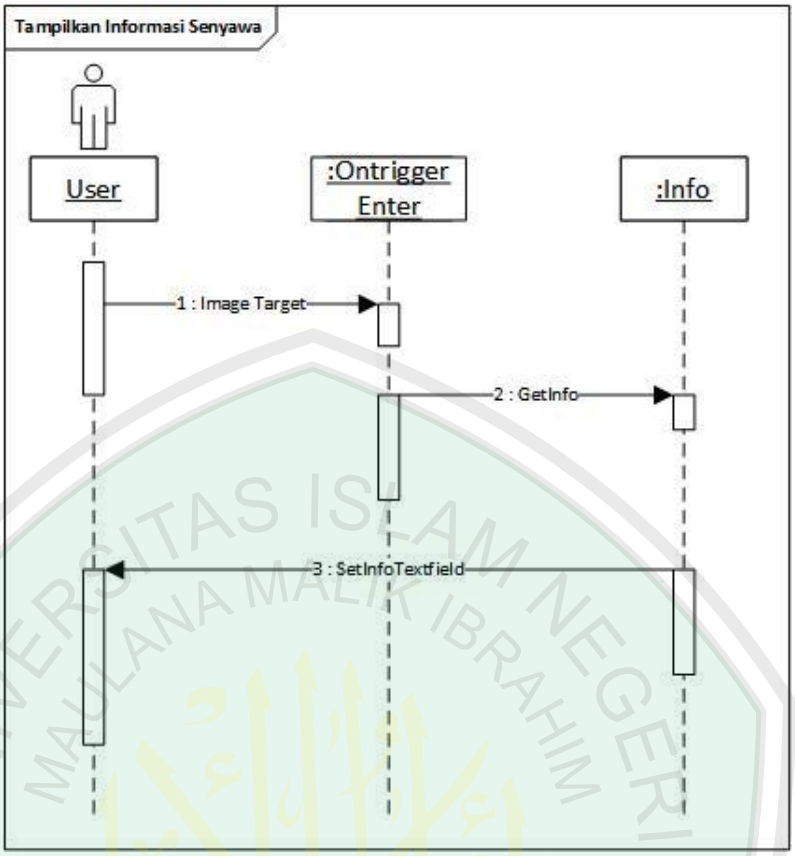


Gambar 3.27 Sequence diagram bonding

Pada Gambar 3.27 dijelaskan bahwa user mendekatkan marker. Class OntriggerEnter mendeteksi *collide* (sentuhan) antar marker kemudian memanggil class AddAtom untuk menambahkan objek atom, setelah itu akan memanggil class MovingAtom untuk memindahkan objek atom pada posisi yang ditentukan kemudian memanggil class DrawBondLine untuk menggabar garis ikatan antar objek atom.

5. Sequence Diagram Tampilkan Informasi Senyawa

   Sequence diagram tampilkan informasi senyawa dapat dilihat pada Gambar 3.28.

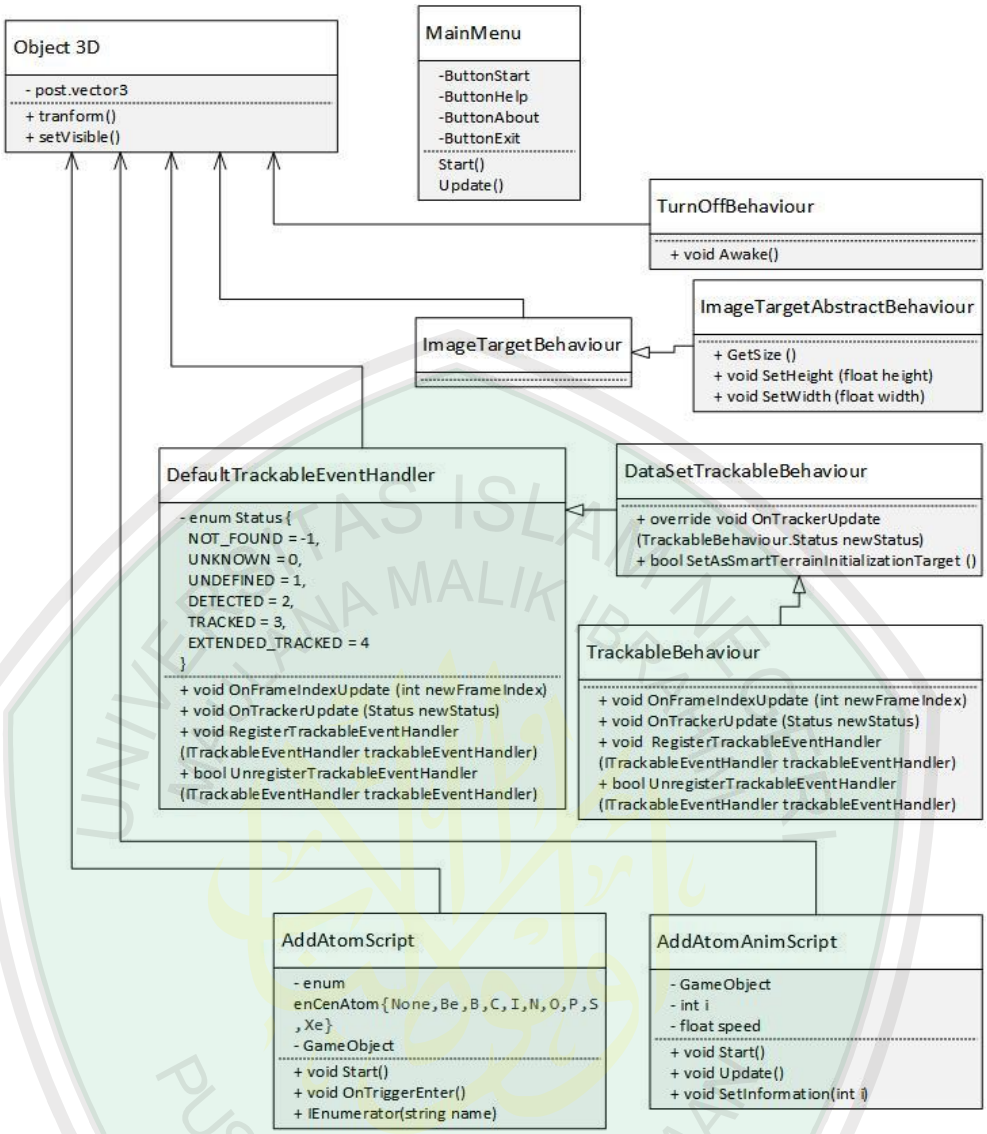


Gambar 3.28 Sequence diagram tampilkan informasi senyawa

Pada Gambar 3.28 dijelaskan bahwa user mendekatkan marker. Class OntriggerEnter mendeteksi *collide* (sentuhan) antar marker kemudian memanggil class Info untuk mengambil attribute info objek ikatan atom untuk ditampilkan dalam textfield.

3.5.5 Class Diagram

Class diagram menggambarkan struktur dan hubungan antar objek-objek yang ada pada sistem. Struktur itu meliputi atribut-atribut dan metode-metode yang ada pada masing-masing kelas. Gambar 3.29 merupakan class diagram dari aplikasi AR ini.



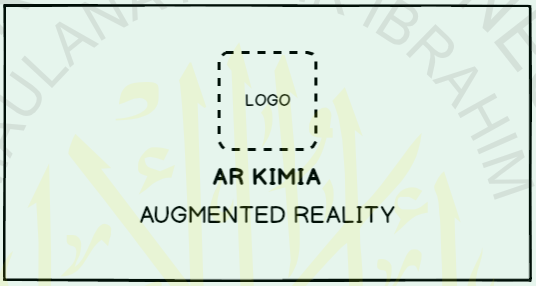
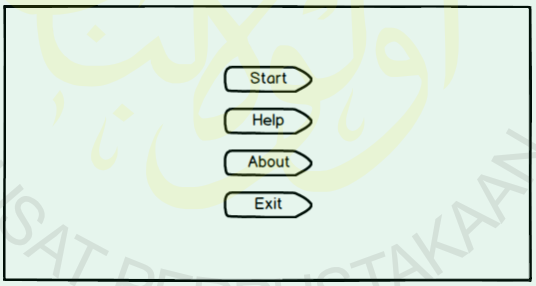
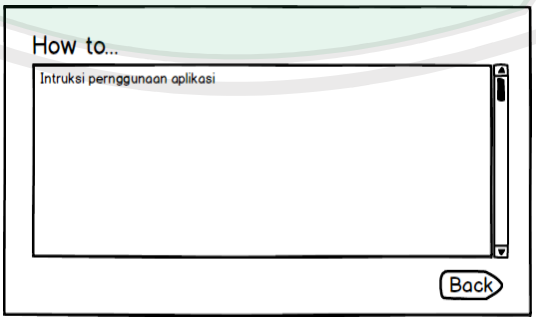
Gambar 3.29 Class diagram aplikasi

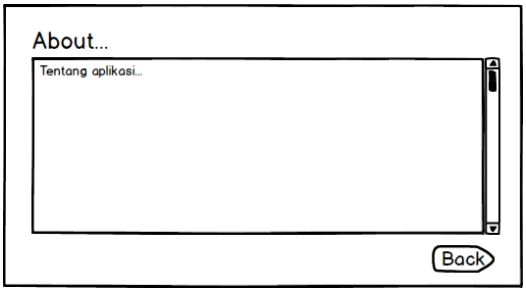
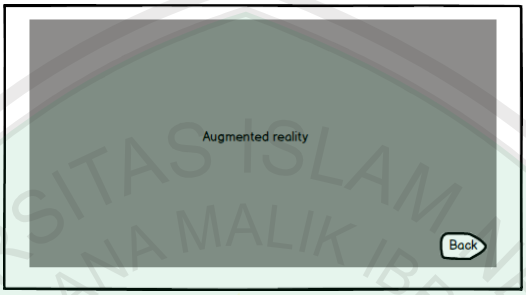
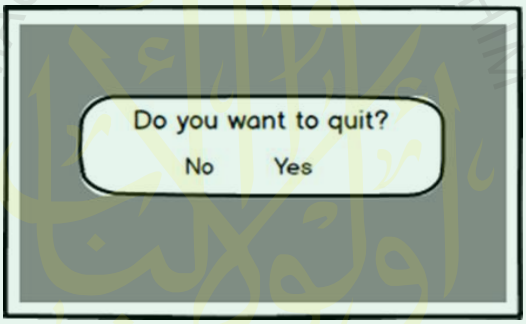
MainMenu adalah class awal ketika user membuka aplikasi. Class Object 3D digunakan untuk mendeteksi marker dan menampilan objek 3D dari marker yang telah terdeteksi. TurnOffBehaviour berfungsi sebagai switch pelacakan marker. ImageTargetBehaviour berfungsi untuk menangani rendering object 3D yang akan ditampilkan. DefaultTrackableHandler berfungsi menangani pelacakan marker. AddAtomScript berfungsi untuk menambahkan object 3D pada image target. AddAtomAnimScript berfungsi untuk menggerakkan (animasi) object 3D yang ada pada image target.

3.5.6 User Interface

Antar muka merupakan visualisasi ide dari aplikasi yang akan dibangun, sehingga dapat memberikan gambaran dari aplikasi yang akan dibuat. Perancangan antarmuka mendeskripsikan rencana tampilan dari setiap form yang akan digunakan pada aplikasi. Rancangan user interface dari aplikasi AR ini dapat dilihat pada Tabel 3.7.

Tabel 3.7 User Interface

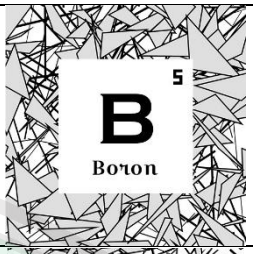
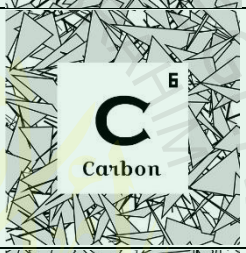
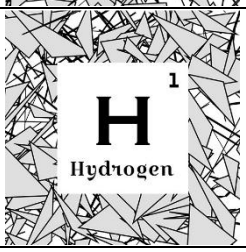
Scene	Nama	Gambar	Keterangan
1	Splash screen		Splash screen berisi logo aplikasi yang muncul beberapa detik kemudian lanjut ke scene menu utama.
2	Menu utama		Tombol start menuju ke scene 5, Tombol Help menuju ke scene 3, Tombol About menuju ke scene 4
3.	Help		Tombol back untuk kembali ke scene 2

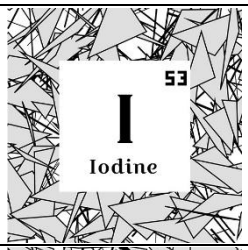
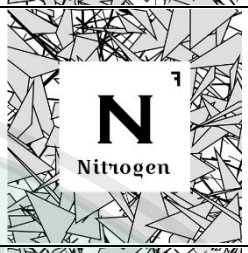
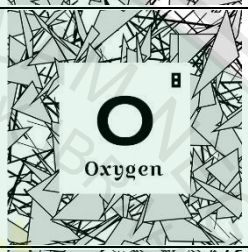
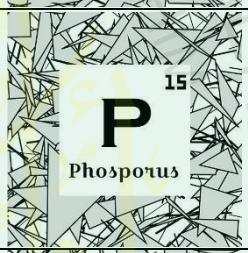
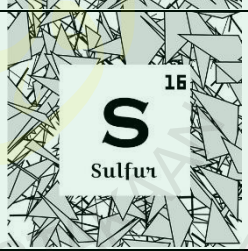
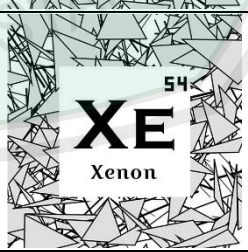
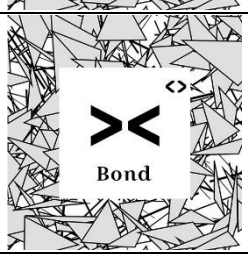
4.	About		Tombol menu untuk kembali ke scene 0.
5	Deteksi marker		Tombol Back untuk kembali ke scene 2.
6	Exit		Tombol Yes untuk keluar aplikasi. Tombol No untuk batal keluar aplikasi.

3.5.7 Desain Marker

Disain marker merupakan rancangan marker yang digunakan. Adapun marker yang digunakan dalam aplikasi ini adalah menggunakan image target marker yang dibuat dan didaftarkan pada website developer vuforia. Ukuran marker yang digunkan dalam aplikasi ini adalah 5 cm x 5 cm dengan file format *.jpg yang nantinya dicetak pada kertas. Perancangan marker dari aplikasi AR ini dapat dilihat pada Tabel 3.8.

Tabel 3.8 Disain marker

No	Nama Marker	Gambar Marker
1	Atom B	
2	Atom Br	
3	Atom C	
4	Atom Cl	
5	Atom F	
6	Atom H	

7	Atom I	
8	Atom N	
9	Atom O	
10	Atom P	
11	Atom S	
12	Atom Xe	
13	Bond	

14	Info		

UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM
PUSAT PERPUSTAKAAN

BAB IV

IMPLEMENTASI DAN PENGUJIAN SISTEM

4.1 Implementasi Sistem

Implementasi sistem merupakan tahap penerjemahan perancangan aplikasi sesuai dengan hasil analisis yang telah dilakukan. Setelah implementasi maka dilakukan pengujian sistem. Pengujian sistem dilakukan untuk mengetahui kekurangan-kekurangan pada aplikasi untuk selanjutnya diadakan perbaikan sistem.

4.2 Implementasi Interface

Implementasi interface merupakan pengaplikasian tampilan dari perancangan aplikasi yang telah dibuat. Dalam implementasi interface pada aplikasi ini, dibutuhkan scene-scene untuk menangani tiap-tiap proses. Setiap scene memiliki fungsi sendiri dan saling terkait satu sama lain.

4.2.1 Tampilan Pembuka

Tampilan pembuka ini berupa splash screen aplikasi. Splash screen ini akan tampil beberapa detik sebelum menu utama. Pada splash screen ini terdapat logo dan nama aplikasi yaitu AR Kimia. Gambar 4.1 merupakan tampilan dari splash screen aplikasi AR Kimia.

Gambar 4.1 Tampilan pembuka splash screen

4.2.2 Tampilan Menu Utama

Tampilan Menu Utama ialah tampilan yang muncul setelah splash screen. Pada menu utama terdapat 4 tombol yaitu: tombol Start, tombol Help, tombol About, dan tombol Exit. Tampilan menu utama seperti pada Gambar 4.2.

Gambar 4.2 Tampilan menu utama

4.2.3 Tampilan Help

Tampilan Help merupakan tampilan panduan dan cara penggunaan aplikasi AR Kimia. Tampilan Help seperti pada Gambar 4.3.

Gambar 4.3 Tampilan help

4.2.4 Tampilan About

Tampilan About merupakan informasi aplikasi berupa tentang aplikasi, versi aplikasi, fitur aplikasi, dan info pengembang. Tampilan About seperti pada Gambar 4.4.

Gambar 4.4 Tampilan about

4.2.5 Tampilan Exit

Tampilan Exit merupakan notifikasi saat user akan keluar dari aplikasi. Ketika user memilih tombol exit pada menu utama, user akan diarahkan menuju tampilan notifikasi untuk keluar dari aplikasi. Tampilan Exit dapat dilihat pada Gambar 4.5.

Gambar 4.5 Tampilan exit

4.2.6 Tampilan Deteksi Marker

Tampilan Deteksi marker akan terbuka setelah user menekan tombol start pada menu utama. Scene deteksi marker ini mengaktifkan kamera perangkat secara otomatis dan melakukan deteksi marker yang ditunjukkan oleh user ke kamera perangkat. Tampilan deteksi seperti Gambar 4.6.

Gambar 4.6 Tampilan Deteksi Marker

4.3 Pengujian Sistem

Pengujian sistem yang dilakukan meliputi pengujian proses setiap scene, pengujian pada beberapa perangkat android dan pengujian pendeteksian marker untuk mengetahui hal-hal yang mempengaruhi keberhasilan pendeteksian marker.

4.3.1 Pengujian Proses

Pengujian pertama yang dilakukan pada aplikasi adalah pengujian proses. Pengujian proses menguji jalannya setiap scene pada aplikasi. Adapun pengujian yang dilakukan pada scene menu utama, yaitu pada tombol Start, tombol Help, tombol About dan tombol Exit. Pengujian proses juga dilakukan pada pendeteksian marker, Ikatan atom, dan Informasi senyawa. Gambar 4.7 merupakan kode program pada menu utama.

```
14    void Update () {
15        transform.position = Vector3.Lerp (transform.position, currentMount.position, speedFactor);
16        transform.rotation = Quaternion.Slerp (transform.rotation, currentMount.rotation, speedFactor);
17        transform.position.Normalize ();
18    }
19
20    public void SetMount(Transform newMount){
21        currentMount = newMount;
22    }
23
24    public void ToLevel2(){
25        Application.LoadLevel (2);
26    }
27
28    public void QuitAplication(){
29        Application.Quit ();
30    }
```

Gambar 4.7 Kode program menu utama

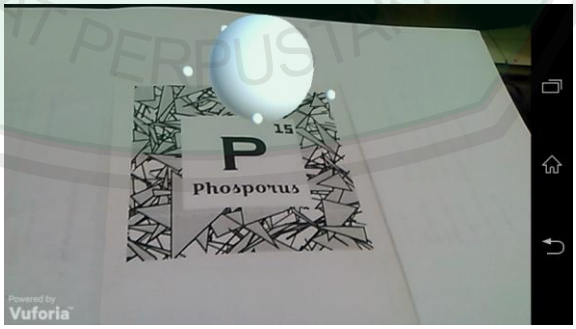
1. Pengujian Start

Pada pengujian tombol start ini, ketika tombol ditekan maka scene menu utama akan berpindah ke scene deteksi marker. Hasil pengujian menunjukkan bahwa scene menu utama berhasil berpindah ke scene deteksi marker. Gambar 4.8 adalah scene deteksi marker saat kamera siap untuk mendeteksi marker.

Gambar 4.8 Kamera siap untuk mendeteksi marker

2. Pengujian Marker dan Deteksi Marker

Pengujian marker ini dilakukan dengan mengarahkan marker pada kamera, kemudian sistem akan mendeteksi marker tersebut. Jika marker berhasil terdeteksi, maka sistem akan menampilkan objek 3D dari marker tersebut. Gambar 4.9 adalah scane saat marker terdeteksi oleh kamera.



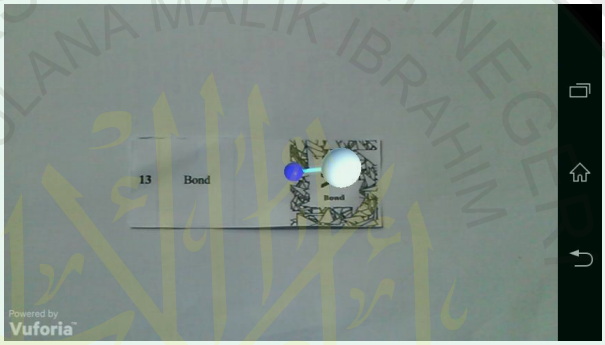
Gambar 4.9 Pendeteksian marker

3. Pengujian Ikatan Atom (Bonding)

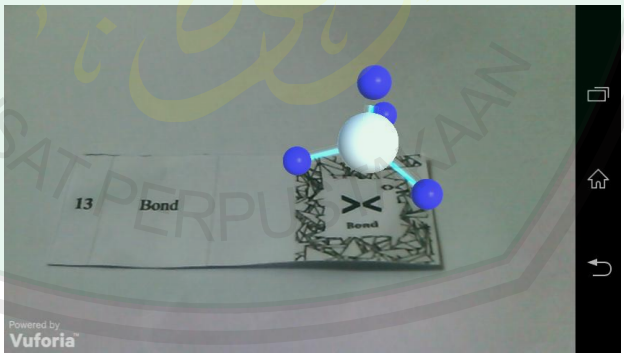
Pengujian ikatan atom (bonding) ini dilakukan dengan medekatkan marker atom pada marker bond. Marker bond akan menampilkan atom

sebagai center ikatan atom saat belum ada atom pada marker bond. Atom selanjutnya akan ditampilkan sebagai atom yang terikat.

Gambar 4.10 Center atom ditambahkan pada marker bond



Gambar 4.11 Atom ikatan ditambahkan pada marker bond



Gambar 4.12 Ikatan atom terbentuk

Gambar 4.10 merupakan tampilan saat center atom ditambahkan pada marker bond, kemudia Gambar 4.11 merupkan tampilan saat atom yang diikat ditambahkan pada marker bond kemudian digambar garis ikatan antara center atom dan atom yang diikat. Gambar 4.12 merupakan tampilan

senyawa hasil ikatan atom. Kode program dari menambah ikatan atom dapat dilihat pada Gambar 4.13. Kode program dari meggambar garis ikatan dapat dilihat pada Gambar 4.14.

```
53    IEnumerator AddAtom(string name){
54
55        // CH4, CH3
56        if (name == "ColliderC" && center == enCenAtom.None) {
57            cb = Instantiate (centerAtomC, transform.position, transform.rotation) as GameObject;
58            cb.transform.SetParent (transform.parent);
59            AddAtoAnimScript anim = GetComponent<AddAtoAnimScript> ();
60            anim.centerAtom = cb;
61            anim.movingCenterAtom = true;
62            anim.Pos = Pos;
63            center = enCenAtom.C;
64        }
65        if (name == "ColliderH" && center == enCenAtom.C) {
66            if (cb != null && i <= 4) {
67                go = Instantiate (H, transform.position, transform.rotation) as GameObject;
68                go.transform.SetParent (transform.parent);
69                AddAtoAnimScript anim = GetComponent<AddAtoAnimScript> ();
70                anim.atom = go;
71                anim.movingAtom = true;
72                anim.i = i;
73
74                yield return new WaitForSeconds (1f);
75
76                drawBond = Instantiate (bond, Pos.transform.position, Pos.transform.rotation) as GameObject;
77                drawBond.transform.SetParent (transform.parent);
78                DrawBond draw = drawBond.GetComponent<DrawBond> ();
79                draw.origin = cb.transform;
80                draw.destination = go.transform;
81
82                i += 1;
83            }
84        }


218    void OnTriggerEnter(Collider col){
219        if (col.transform.tag == "Collider") {
220            string name = col.transform.name;
221            //AddAtom (name);
222            StopCoroutine("AddAtom");
223            StartCoroutine ("AddAtom",name);
224            Debug.Log (name);
225        }
226    }
```

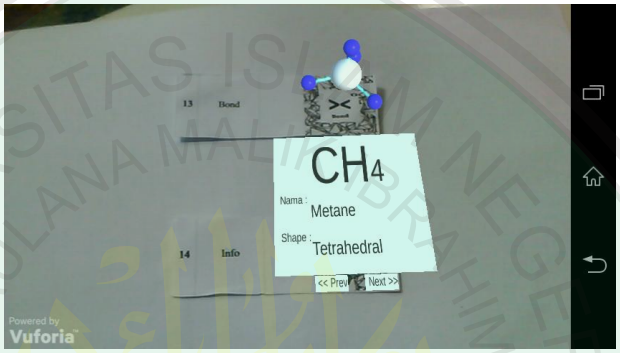
Gambar 4.13 Kode program ikatan (bonding) atom menambah atom

```
15    void Start () {
16        lineRenderer = GetComponent<LineRenderer> ();
17        lineRenderer.SetPosition (0, origin.position);
18        lineRenderer.SetWidth (.45f, .45f);
19
20        dist = Vector3.Distance (origin.position, destination.position);
21    }
22
23    // Update is called once per frame
24    void Update () {
25        if (lineRenderer != null) {
26            if (counter < dist) {
27                counter += .1f / lineDrawSpeed;
28                float x = Mathf.Lerp (0, dist, counter);
29
30                Vector3 pointA = origin.position;
31                Vector3 PointB = destination.position;
32
33                Vector3 pointALongLine = x * Vector3.Normalize (PointB - pointA) + pointA;
34
35                lineRenderer.SetPosition (1, pointALongLine);
36            }
37        }
38    }
```

Gambar 4.14 Kode program ikatan (bonding) atom menggambar garis ikatan

4. Pengujian Informasi Senyawa

Pengujian Informasi senyawa dilakukan dengan medekatkan marker hint pada marker bond yang sudah melakukan ikatan atom. Kemudian marker hint akan menampilkan simbol senyawa, nama senyawa, dan bentuk geometry senyawa berdasarkan teori VSEPR. Pengujian Informasi senyawa dapat dilihat pada Gambar 4.15.



Gambar 4.15 Informasi senyawa pada marker hint

Gambar 4.15 merupkan tampilan dari informasi senyawa saat objek marker hint disentuhkan/didekatkan pada marker bond, kemudian ditampilkan informasi dari senyawa yang terbentuk pada marker bond. Kode program informasi senyawa dapat dilihat pada Gambar 4.16.

```
12    public void SetInfoAtom(string kodeMolecule,string namaMolecule1,
13        string namaMolecule2, string shapeMolecule1, string shapeMolecule2){
14        this.kodeMolecule = kodeMolecule;
15        this.namaMolecule1 = namaMolecule1;
16        this.namaMolecule2 = namaMolecule2;
17        this.shapeMolecule1 = shapeMolecule1;
18        this.shapeMolecule2 = shapeMolecule2;
19    }
```

Gambar 4.16 Kode program Informasi senyawa pada marker hint

5. Pengujian Help

Pengujian selanjutnya adalah pengujian Help. Ketika user menekan tombol help pada menu utama, maka sistem akan menampilkan informasi dari mengenai cara penggunaan aplikasi AR Kimia. Hasil pengujian

menunjukkan bahwa sistem berhasil menampilkan Help aplikasi AR. Gambar 4.17 merupakan tampilan Help aplikasi AR yang telah dibuat.

Gambar 4.17 Help pada aplikasi AR

6. Pengujian About

Pengujian selanjutnya adalah pengujian About. Ketika user menekan tombol About pada menu utama, maka sistem akan menampilkan informasi dari mengenai tentang aplikasi, versi aplikasi, fitur aplikasi, change log aplikasi, dan info pengembang. Hasil pengujian menunjukkan bahwa sistem berhasil menampilkan About aplikasi AR. Gambar 4.18 merupakan tampilan About aplikasi AR yang telah dibuat.

Gambar 4.18 About pada aplikasi

7. Pengujian Exit

Pengujian selanjutnya adalah pengujian pada Exit. Pengujian ini dilakukan untuk mengetahui apakah aplikasi akan tertutup atau tidak ketika

user memilih keluar dari aplikasi. Ketika user menekan tombol Exit, maka akan muncul tampilan notifikasi seperti pada Gambar 4.19. Hasil pengujian menunjukkan bahwa ketika user menekan tombol exit dan memilih yes, maka aplikasi tertutup.

Gambar 4.19 Notifikasi keluar aplikasi

4.3.2 Pengujian Perangkat Android

Pengujian perangkat dilakukan untuk mengetahui kekurangan aplikasi saat diterapkan pada smartphone. Pengujian ini dilakukan pada beberapa smartphone dengan spesifikasi berbeda. Daftar perangkat android yang digunakan untuk uji coba aplikasi AR beserta spesifikasinya terdapat pada Tabel 4.1.

Tabel 4.1 Spesifikasi perangkat android

Perangkat	OS	CPU	GPU	Resolusi	Kamera
Sony Xperia M	Android 4.3 Jelly Bean	Dual-core 1.0 GHz Krait	Adreno 305	4.0 Inch, 480 x 854 pixels	Belakang: 5 MP
Samsung Tab 3 7.0	Android 4.1.2 Jelly Bean	Dual Core 1.2 GHz Cortex-A9	PowerVR SGX540	7.0 Inch, 600x1024 pixels	Belakang: 3.15 MP

Evercoss A7T	Android 4.2.2 Jelly Bean	Dual-core 1.3 GHz Mediatek MT6572	-	4.0 Inch, 480x800 pixels	Belakang: 5 MP
Asus Zenfone 2	Android 6.0 Marshmallow	Quad-core 1.8 GHz	PowerVR G6430	5.5 inches, 1080x1920 pixels	Belakang: 13 MP
Lenovo A369i	Android 4.2.2 Jelly Bean	Dual-core 1.3 GHz Cortex-A7	Mali 400	4.0 inches, 480x800 pixels	Belakang: 2 MP

Hasil pengujian aplikasi AR pada beberapa perangkat android dapat dilihat pada Tabel 4.2.

Tabel 4.2 Hasil pengujian pada perangkat android

Nama Perangkat	**Hasil Uji Coba**	
	Status	**Keterangan**
Sony Xperia M	Berhasil	No delay
Samsung Tab 3 7.0	Berhasil	No delay
Evercoss A7T	Berhasil	Delay rata-rata 4 detik
Asus Zenfone 2	Berhasil	No delay
Lenovo A369i	Berhasil	No delay

Saat pengujian dilakukan terhadap Evercross A7T aplikasi AR dapat berjalan lancar tetapi terdapat delay rata-rata 4 detik pada aplikasi, sedangkan pada Samsung Tab 3 7.0 dan Asus Zenfone 2 aplikasi AR tetap berjalan lancar dan tidak ada delay meski terdapat perbedaan resolusi layar perangkat. Hal ini dikarenakan

ketika proses rendering, resolusi untuk compiler .apk disetting pada resolusi maksimal 720x1280 pixel.

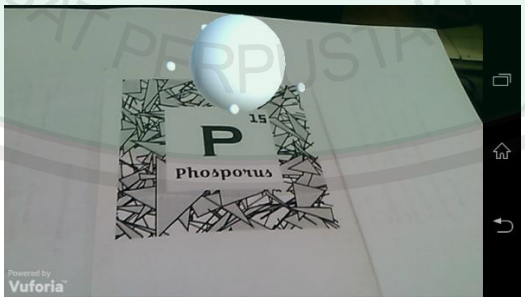
4.4 Pengujian Tracking Marker

Pengujian tracking marker ini dilakukan untuk mengetahui hal-hal yang mempengaruhi pada proses tracking marker. Pengujian ini meliputi pengujian intensitas cahaya, pengujian oklusi, dan pengujian akurasi.

4.4.1 Pengujian Intensitas Cahaya

Pengujian intensitas cahaya dilakukan dalam dua tempat, yaitu di luar ruangan dan di dalam ruangan. Untuk pengujian dalam ruangan dilakukan dengan dua keadaan yaitu diluar ruangan, dalam ruangan. Pengujian ini mengunakan perangkat android sony experia m. Hasil pengujian intensitas cahaya dapat dilihat pada Tabel 4.3.

Tabel 4.3 Hasil pengujian intensitas cahaya menggunakan perangkat android sony experia m

Kondisi	**Hasil Pengujian**	
	Gambar	**Keterangan**
Luar ruagan siang hari		Deteksi lebih cepat. Objek 3D dapat ditampilkan dengan baik.

Dalam ruangan dengan lampu		Deteksi lebih lambat. Objek 3D dapat ditampilkan dengan baik.

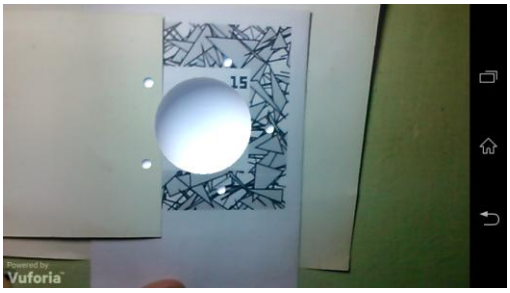
Cahaya memegang peran penting dalam pendeteksian marker. Seperti yang telah ditunjukkan pada Tabel 4.3, pendeteksian lebih lambat saat intensitas cahaya berkurang.

4.4.2 Pengujian Oklusi

Pengujian oklusi yaitu pengujian marker yang terhalang sesuatu. Pengujian ini dilakukan dengan tujuan mengetahui apakah marker tetap dapat terdeteksi dengan kondisi tidak normal. Pengujian oklusi ini dilakukan dengan cara menutup marker 20% bagian, 35% bagian, 50% bagian dan 75% bagian. Pengujian ini mengunakan perangkat android sony experia m.

Tabel 4.4 Pengujian oklusi mengunakan perangkat android sony experia m

Marker	**Hasil Pengujian**	
	Gambar	**Keterangan**
Tertutup 20%		Marker terdeteksi dan objek 3D dapat ditampilkan

Tertutup 35%		Marker masih terdeteksi dan objek 3D dapat ditampilkan.
Tertutup 50%		Marker tetap terdeteksi namun pendeteksian lebih lambat, dan objek 3D dapat ditampilkan.
Tertutup 75%		Marker tidak dapat terdeteksi, dan objek 3D tidak dapat ditampilkan.

Hasil pengujian yang ditunjukkan Tabel 4.4, pada saat marker tertutup 50%, pendeteksian marker mulai lambat, saat marker tertutup 75%, marker tidak dapat terdeteksi dan objek 3d tidak dapat ditampilkan. Kesimpulan dari hasil pengujian okulusi ini bahwa marker tidak boleh tertutup lebih dari 75%.

4.4.3 Pengujian Akurasi

Pengujian akurasi yaitu sebuah pengujian pemindaian objek marker pada sudut tertentu yaitu pada sudut: 15^o, 30^o, 45^o, 60^o, 75^o, 90^o dan pada jarak tertentu yaitu pada jarak: 10cm, 20cm, 40cm, 60cm, 80cm, 100cm dari kamera.

Tabel 4.5 Pengujian Akurasi

Jarak	Sudut	Cahaya luar ruagan siang hari	Cahaya dalam ruangan dengan lampu
10	15^o	Tidak berhasil	Tidak berhasil
	30^o	Tidak berhasil	Tidak berhasil
	45^o	Tidak berhasil	Tidak berhasil
	60^o	Tidak berhasil	Tidak berhasil
	75^o	Tidak berhasil	Tidak berhasil
	90^o	Tidak berhasil	Tidak berhasil
20	15^o	Tidak berhasil	Tidak berhasil
	30^o	Berhasil	Berhasil
	45^o	Berhasil	Berhasil
	60^o	Berhasil	Berhasil
	75^o	Berhasil	Berhasil
	90^o	Berhasil	Berhasil
40	15^o	Tidak berhasil	Tidak berhasil
	30^o	Berhasil	Berhasil
	45^o	Berhasil	Berhasil
	60^o	Berhasil	Berhasil
	75^o	Berhasil	Berhasil
	90^o	Berhasil	Berhasil
60	15^o	Tidak berhasil	Tidak berhasil
	30^o	Berhasil	Berhasil
	45^o	Berhasil	Berhasil
	60^o	Berhasil	Berhasil
	75^o	Berhasil	Berhasil
	90^o	Berhasil	Berhasil
80	15^o	Tidak berhasil	Tidak berhasil
	30^o	Tidak berhasil	Tidak berhasil
	45^o	Berhasil	Tidak berhasil
	60^o	Berhasil	Berhasil
	75^o	Berhasil	Berhasil
	90^o	Berhasil	Berhasil
100	15^o	Tidak berhasil	Tidak berhasil
	30^o	Tidak berhasil	Tidak berhasil
	45^o	Tidak berhasil	Tidak berhasil
	60^o	Tidak berhasil	Tidak berhasil
	75^o	Tidak berhasil	Tidak berhasil
	90^o	Tidak berhasil	Tidak berhasil

Hasil pengujian yang ditunjukkan pada Tabel 4.5, pada jarak 10cm marker tidak dapat terdeteksi. Hal ini dikerenakan marker melebihi frame kamera dan kamera tidak dapat fokus pada jarak tersebut. Pada sudut 15^o marker tidak dapat

terdeteksi pada setiap jarak dan intesitas cahaya yang diuji. Keberhasilan pendeteksian terjadi pada sudut 30^o dan jarak kamera pada 20cm – 60cm. Pada jarak 80cm marker terdeteksi pada sudut 45^o – 90^o pada cahaya luar ruangan siang hari dan pada sudut 60^o – 90^o pada cahaya dalam ruangan dengan lampu. Pendeteksian tidak berhasil pada jarak 100cm di setiap sudut yang diuji. Hal ini dikarenakan marker tampak sangat kecil sehingga gambar sudut yang ada pada gambar marker tidak jelas. Kesimpulan dari pengujian ini adalah pedeteksian marker sangat baik ketika jarak marker ke kamera yaitu 20cm – 60cm dan pada sudut 30^o, 45^o, 60^o, 75^o, 90^o.

4.5 Integrasi Nilai Islam

Manusia dan seluruh lingkungan hidupnya merupakan mahluk yang terbentuk dari elemen-elemen dan subtansi-subtansi yang tergabung menjadi sebuah ikatan kimia menurut hukum Allah. Allah berfirman dalam surat Al-Hijr ayat 26:

وَلَقَدْ خَلَقْنَا الْإِنْسَانَ مِنْ صَلْصَالٍ مِنْ حَمَإٍ مَسْنُونٍ

"Dan sesungguhnya Kami telah menciptakan manusia (Adam) dari tanah liat kering (yang berasal) dari lumpur hitam yang diberi bentuk". (Q.S Al-Hijr (15): 26)

Ayat di atas menjelaskan bagaimana manusia diciptakan. Manusia tercipta melalui sebuah proses kimia interaktif antara berbagai unsur dalam tanah yang bekerja menurut hukum-hukum Allah melalui proses perubahan dan kombinsi tertentu.

Begitu juga dengan proses penciptaan alam semesta dan semua benda yang ada di dalamnya. Allah berfirman dalam surat Al-Fussilat ayat 11-12:

ثُمَّ اسْتَوَىٰ إِلَى السَّمَاءِ وَهِيَ دُخَانٌ فَقَالَ لَهَا وَلِلْأَرْضِ ائْتِيَا طَوْعًا أَوْ كَرْهًا قَالَتَا أَتَيْنَا طَائِعِينَ

فَقَضَاهُنَّ سَبْعَ سَمَاوَاتٍ فِي يَوْمَيْنِ وَأَوْحَىٰ فِي كُلِّ سَمَاءٍ أَمْرَهَا وَزَيَّنَّا السَّمَاءَ الدُّنْيَا بِمَصَابِيحَ وَحِفْظًا ذَٰلِكَ تَقْدِيرُ الْعَزِيزِ الْعَلِيمِ

"Kemudian Dia menuju langit dan langit itu masih merupakan asap, lalu Dia berkata kepadanya dan kepada bumi:"Datanglah kamu keduanya menurut perintah-Ku dengan suka hati atau terpaksa. "Keduanya menjawab:"Kami datang dengan suka hati. "Maka Dia menjadikannya tujuh langit dalam dua masa dan Dia mewahyukan pada tiap-tiap langit urusan-Nya. Dan kami hiasi langit yang dekat dengan bintang-bintang yang cemerlang dan Kami menjadikannya dengan sebaik-baiknya. Demikianlah ketentuan Yang Maha Perkasa lagi Maha Penyayang". (Q.S Fussilat (41): 11-12)

Dua ayat diatas memberikan petunjuk bagi manusia tentang pembuatan subtansi baru dengan menggabungkan berbagai unsur dan tentang mempelajari rekasi kimia dari penggabungan unsur-unsur itu dengan berbagai proporsinya.

Ikatan kimia adalah ikatan yang terjadi antar atom atau antar molekul. Bila suatu atom kehilangan elektron, atom tersebut akan menjadi kation yang memiliki jumlah elektron yang sama dengan gas mulia terdekat, sementara bila atom mendapatkan elektron, atom tersebut akan menjadi anion yang memiliki jumlah elektron yang sama dengan atom gas mulia terdekatnya. Hal ini menyebabkan atom bergabung untuk mencapai konfigurasi elektron yang lebih stabil. Allah berfirman dalam surat Al-Dzariyat ayat 49 dan dalam surat Yaasiin ayat 36:

وَمِنْ كُلِّ شَيْءٍ خَلَقْنَا زَوْجَيْنِ لَعَلَّكُمْ تَذَكَّرُونَ

"Dan segala sesuatu Kami ciptakan berpasang-pasangan supaya kamu mengingat akan kebesaran Allah". (Q.S Al-Dzariyat (51): 49)

سُبْحَانَ الَّذِي خَلَقَ الْأَزْوَاجَ كُلَّهَا مِمَّا تُنْبِتُ الْأَرْضُ وَمِنْ أَنْفُسِهِمْ وَمِمَّا لَا يَعْلَمُونَ

"Maha suci Tuhan yang telah menciptakan pasangan-pasangan semuanya, baik dari apa yang ditumbuhkan oleh bumi dan dari diri mereka maupun dari apa yang tidak mereka ketahui". (Q.S Yaasiin (36): 36)

Dua ayat diatas menjelaskan bahwa Allah telah menciptakan semuaya dalam pasangan-pasangan hal ini sesuai dengan konsep ikatan kimia bahwa atom akan berpasangan untuk mencapai suatu kestabilan senyawa.

BAB V

PENUTUP

5.1 Kesimpulan

Kesimpulan yang dapat diambil berdasarkan penelitian yang telah dilakukan adalah teknologi augmented reality dengan metode FAST corner detections dapat digunakan sebagai media simulasi dan diterapkan pada perangkat android dengan baik. Aplikasi AR Kimia ini telah melalui beberapa pengujian, meliputi pengujian proses, pengujian sistem pada perangkat android berbeda, dan pengujian pendeteksian marker.

Hasil pengujian proses menunjukkan bahwa setiap proses pada aplikasi AR Kimia berjalan sesuai dengan rancangan. Hasil pengujian sistem menunjukkan bahwa aplikasi AR Kimia berhasil dipasang pada beberapa perangkat android yang memiliki perbedaan spesifikasi. Perbedaan spesifikasi setiap perangkat android mempengaruhi cepat lambatnya proses loading aplikasi. Hasil pengujian intensitas cahaya, oklusi (tertutup), jarak dan sudut menunjukkan bahwa pendeteksian mempengaruhi proses pendeteksian marker. Ketika intensitas cahaya rendah, proses pendeteksian marker semakin melambat. Kemudian ketika gambar marker tertutup, sistem tetap dapat mendeteksi marker dengan baik hingga gambar marker tertutup 35%. Ketika gambar marker tertutup 50%, sistem tetap dapat mendeteksi marker dan menampilkan objek 3D namun pendeteksian mulai melambat, dan ketika gambar marker tertutup 75% sistem tidak dapat mendeteksi marker dan menampilkan objek 3D. Hasil dari pengujian akurasi menunjukkan bahwa pendeteksian marker sangat baik ketika jarak marker ke kamera yaitu 20cm – 60cm dan pada sudut 30^o, 45^o, 60^o, 75^o, 90^o.

5.2 Saran

Berdasarkan penelitian ini, saran untuk penelitian yang selanjutnya adalah pengembangan aplikasi AR Kimia yang lebih sempurna. Pengembangan dapat dilakukan pada beberapa bagian sebagai berikut:

1. Penambahan jumlah senyawa ikatan atom.
2. Pengembangan pada bagian desain aplikasi.
3. Penambahan animasi pada aplikasi.

UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM PUSAT PERPUSTAKAAN

DAFTAR PUSTAKA

Andriyadi, Anggi. 2011. *Augmented Reality with ARToolKit*. Nulis Buku. Bandar Lampung.

BSNP. 2006. *Permendiknas RI No. 22 Tahun 2006 tentang Standar Isi untuk Satuan Pendidikan Dasar dan Menengah*. Jakarta.

Cawood, Stephen dan Mark Fiala. 2008. *Augmented Reality: A Practical Guide*. The Pragmatic Programmers, LLC.

Chang, Raymond. 2010. *Chemistry, Tenth Edition*. McGraw-Hill. New York.

Harris, Chris and Mike Stephens, 1988. *A Combined Corner and Edge Detector*. Plessey Research Roke Manor. United Kingdom.

Hernowo, T.F et al, 2014. *Rancang Bangun Edugame Pembelajaran Kesehatan Gigi Untuk Anak-anak Berbasis Unity 3D*.

Martono, Kurniawan Teguh. *Augmented Reality sebagai Metafora Baru dalam Teknologi Interaksi Manusia dan Komputer*. Jurnal sistem komputer, Vol. I, 2011.

Roedavan, Rickman. 2014. *Unity Tutorial Game Engine*. Informatika. Bandung.

Ronald, T. Azuma. 1997. *A Survey of Augmented Reality*. Jurnal Teleoperators and Virtual Environments.

Sudareswaran, V dan Wang, Kenneth; et al. 2003. *3D Audio Augmented Reality: Implementation and Experiments.* Proceedings of the 2nd IEEE and ACM International Symposium on Mixed and Augmented Reality.

Sirhan, Ghassan. 2007. *Learning Difficulties in Chemistry : an Overview*. Journal of Turkish Science Education Volume 4, Issue 2.

Takeuchi, Yashito. 2006. *Buku Teks Pengantar Kimia*. Tokyo: Iwanami Shoten, Publishers. [terjemahan bahasa indonesia]

Unity 3D documentation 5.3.4f.

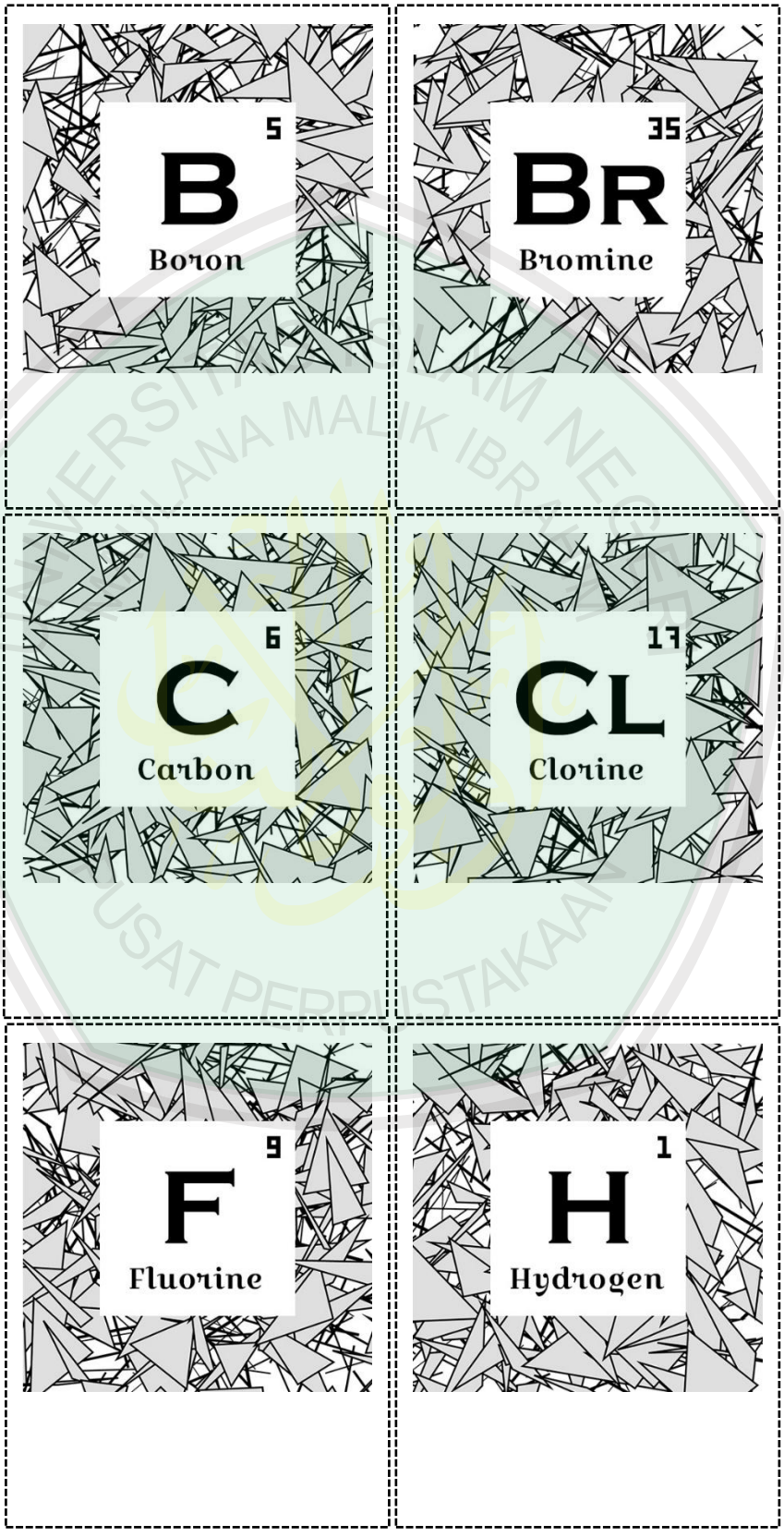
Zumdahl, Steven S. and Susan A. Zumdahl. 2014. *Chemistry, Ninth Edition*. Brooks Cole. USA.

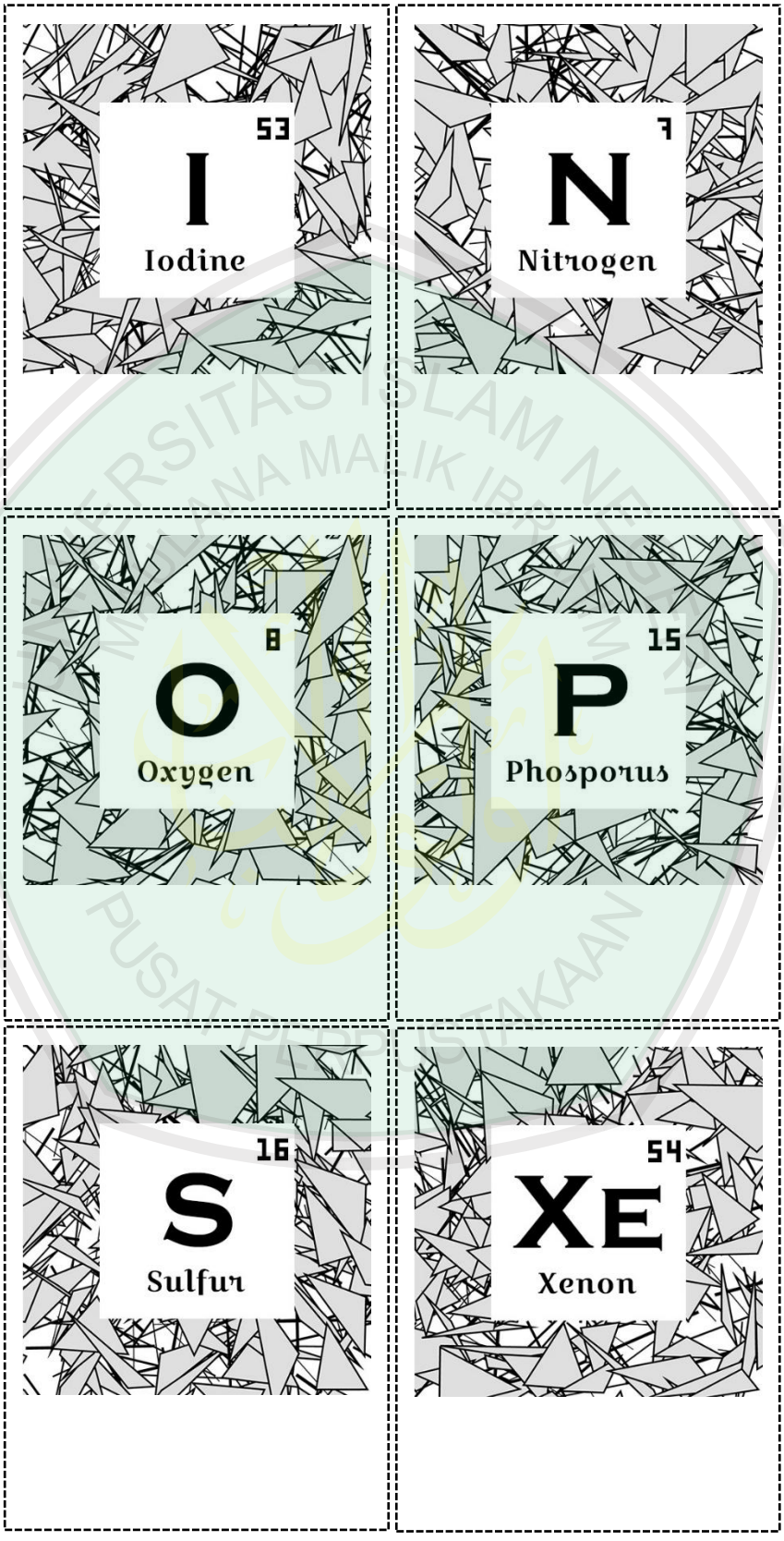
http://developer.vuforia.com/library/articles/training [diakses pada 19-04-2016].

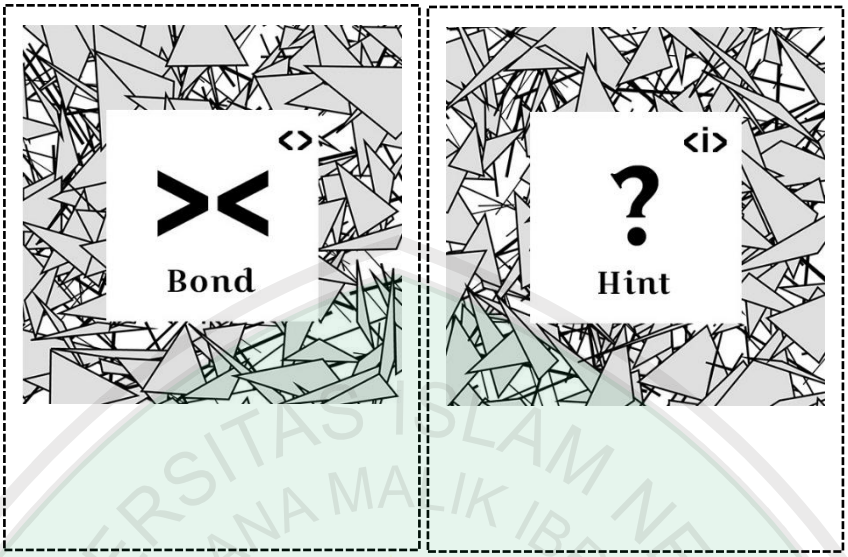
http://www.unity3d.com. [diakses pada 19-04-2016].

LAMPIRAN

PRINTABLE MARKER




53
I
Iodine
7
N
Nitrogen
8
O
Oxygen
15
P
Phosporus
16
S
Sulfur
54
Xe
Xenon



UNIVERSITAS ISLAM NEGERI
MAULANA MALIK IBRAHIM
PUSAT PERPUSTAKAAN