

BAB II

TINJAUAN PUSTAKA

2. Penelitian yang Relevan

Penelitian dengan judul ”*Augmented Reality Furniture Menggunakan Metode Luther-Sutopo Berbasis Android*” merujuk pada beberapa penelitian yang relevan yaitu diantaranya sebagai berikut :

1. Penelitian yang dilakukan oleh Yudhi Widya Arthana Rustam (2021) dengan judul “Perancangan Aplikasi *Mobile Katalog Furniture Menggunakan Teknologi Augmented Reality*” tujuan dari penelitian tersebut ialah menampilkan visualisasi tiga dimensi (3D) dari sebuah produk *furniture* yang dipadukan dengan katalog *furniture* yang sudah disediakan untuk dijadikan sebagai media promosi baru agar lebih menarik dimata konsumen, menyediakan aplikasi ini pelaku bisnis dapat menarik minat pembeli dan dapat mengenalkan produknya, membuat perbedaan pemasaran dalam persaingan penjualan *furniture*. Metode yang digunakan Yudhi ialah *Luther-Sutopo* [26]. Relevansi penelitian yang dilakukan Yudhi dengan penelitian penulis ialah perancangan aplikasi *furniture* guna memberikan informasi dan mempermudah pengguna umum dalam memilih *furniture* yang akan dipilih atau dibeli.
2. Penelitian yang dilakukan oleh Eka Purnama Harahap, Untung Rahardja , dan Muhammad Salamuddin (2018) dengan judul “Aplikasi Panduan dan Pembayaran Tiket Masuk Mendaki Gunung Menggunakan Metodologi Sistem Multimedia *Luther-Sutopo Berbasis Android*” tujuan dari penelitian tersebut ialah menerapkan metode *Luther-Sutopo* dalam perncangan aplikasi pantuan dan pembayaran tiket masuk mendaki gunung, yang diharapkan dengan adanya aplikasi tersebut dapat memperlancar kegiatan para pendaki untuk mengakses informasi dan ilmu dasar pendakian [27]. Relevansi dengan penelitian ini ialah pada penggunaan metode *Luther-Sutopo* dalam perancangan aplikasi yang

akan dilakukan oleh penulis, sehingga dengan menerapkan metode *Luther-Sutopo* penulis dapat membangun aplikasi kustomisasi *furniture* sesuai dengan maksud dan tujuan penelitian ini.

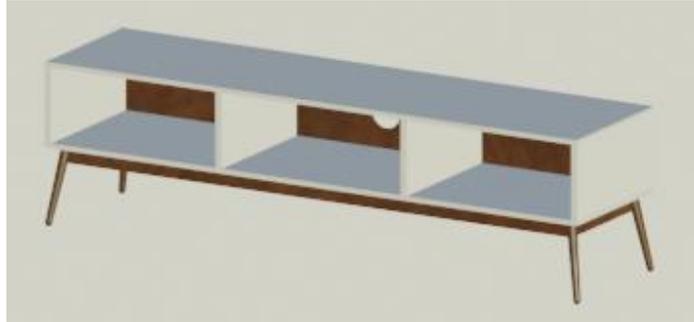
3. Penelitian yang dilakukan oleh Elma Tryana (2021) dengan judul “Aplikasi berbasis *Augmented Reality* Untuk Pengenalan *Furniture* Bahan Rotan” tujuan dari penelitian tersebut ialah memanfaatkan AR sebagai media pengenalan *furniture* berbahan rotan berbasis android [18]. Relevansi dengan penelitian ini ialah pemanfaatan AR sebagai sarana informasi kepada pengguna tentang *furniture*.
4. Penelitian yang dilakukan oleh Wahyu Untoro, I Putu Satwika, Anak Agung Ayu Putri Ardyant, dan Wawan Sujarwo (2019) dengan judul “Perancangan Game Bedugul Forest Dengan Metode Pengembangan Multimedia *Luther-Sutopo*” tujuan dari penelitian tersebut ialah pembuatan game edukasi agar meningkatkan pemahaman siswa sehingga siswa akan tertarik belajar suatu hal baru [33]. Dalam pembuatan game edukasi tersebut metode yang digunakan adalah *Luther-Sutopo*, akan tetapi terdapat penambahan satu tahapan dari tahapan – tahapan yang sudah ada yaitu *maintenance* agar tahapan *publishing game* sukses dan berjalan lancar. Relevansi dengan penelitian ini ialah penggunaan metode *Luther-Sutopo* sebagai tahapan – tahapan pembangunan aplikasi.

3. Pemodelan *Furniture* yang dibuat

Pemodelan *furniture* yang akan dibuat hanya dibagian ruangan kamar tempat tidur saja diantaranya lemari pakaian, kasur, meja belajar, kursi, bingkai photo, tong sampah plastik, dan tempat tidur set. Untuk pilihan warna pemodelan *furniture* yang akan dipilih adalah warna hitam, putih, abu-abu dan warna natural (warna serat kayu).

1. Meja

Dibawah ini merupakan contoh gambar meja.



Gambar 2.1 Meja

2. Wardrobe

Dibawah ini merupakan contoh gambar wardrobe.



Gambar 2.2 wardrobe

3. Sofa

Dibawah ini merupakan contoh gambar sofa.



Gambar 2.3 Sofa

4. Bingkai Foto

Dibawah ini merupakan contoh gambar bingkai foto.



Gambar 2.4 Bingkai Foto

5. Tempat Sampah

Dibawah ini merupakan contoh gambar tempat sampah.



Gambar 2.5 Tempat Sampah

6. Tempat Tidur

Dibawah ini merupakan contoh gambar tempat tidur.



Gambar 2.6 Tempat Tidur

Pada gambar Gambar 2.7 ditampilkan keseluruhan furniture dan diterapkan pada ruang kamar. Adapun furniture yang ditampilkan yaitu meja, wardrobe, sofa, bingkai foto, tempat sampah, tempat tidur.



Gambar 2.7 Furniture Keseluruhan

4. Landasan Teori

Landasan teori yang berkaitan dengan materi atau teori yang digunakan sebagai acuan dalam melakukan penelitian. Landasan teori yang diuraikan merupakan hasil dari literatur dan buku-buku.

I.4.1 Pengertian UML

UML (Unified Modeling Language) adalah suatu metode pemodelan secara visual yang berfungsi sebagai sarana perancangan sistem berorientasi obyek.

UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan system yang besar dan kompleks. UML tidak hanya digunakan dalam proses pemodelan perangkat lunak, namun hampir dalam semua bidang yang membutuhkan pemodelan [28, 30].

I.4.2 Perangkat Pemodelan Berbasis Object

Perangkat Pemodelan Berbasis Object sebagai berikut:

1. *Use Case Diagram*

Use case diagram adalah abstraksi dari interaksi antara sistem dan *actor*. *Use case diagram* bekerja dengan cara mendeskripsikan tipe interaksi antara *user* sebuah sistem dengan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. *Use case* merupakan konstruksi untuk mendeskripsikan bagaimana sistem akan terlihat di mata *user*. Sedangkan *use case diagram* memfasilitasi komunikasi diantara analis dan pengguna serta antara analis dan *client*.

2. *Class Diagram*

Class adalah dekripsi kelompok obyek-obyek dengan *property*, perilaku (operasi) dan relasi yang sama. Sehingga dengan adanya *class diagram* dapat memberikan pandangan *global* atas sebuah *system*. Hal tersebut tercermin dari *class – class* yang ada dan relasinya satu dengan yang lainnya. Sebuah sistem biasanya mempunyai beberapa *class diagram*. *Class diagram* sangat membantu dalam visualisasi struktur kelas dari suatu sistem.

3. *Sequence Diagram*

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah *scenario*. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara obyek juga interaksi antara obyek, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

4. *Activity Diagram*

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.

I.4.3 *Augmented Reality*

Augmented Reality (AR) ialah sebutan yang digunakan pada lingkup dimana dunia nyata dan dunia virtual yang dihasilkan komputer dihubungkan oleh garis yang sangat tipis [8, 9, 15]; Juga merupakan teknologi interaktif yang menggabungkan objek virtual 2D dan atau 3D menjadi nyata, dan memiliki karakteristik realitas tidak terbatas pada penlihatan saja [23, 24]. Pada tahun 1997, Ronald Asma menggambarkan AR sebagai sebuah sistem dengan ciri – ciri seperti berikut:

1. Perpaduan dunia nyata dan dunia virtual.
2. Berjalanlah secara interaktif di dunia nyata.
3. Terintegrasi pada tiga dimensi (3D).

Sederhananya, AR dapat digambarkan sebagai dunia nyata dengan menambahkan objek virtual. Obyek nyata dan virtual dapat digabungkan menggunakan teknologi tampilan sesuai yang dapat berinteraksi melalui perangkat tertentu [8, 15].

AR merupakan variasi dari *Virtual Environments* (VE), atau yang lebih dikenal dengan istilah *Virtual Reality* (VR). Teknologi VE 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 Object virtual yang ditambahkan atau tergabung dengan lingkungan nyata. Tidak seperti VR yang sepenuhnya menggantikan lingkungan nyata, AR sekedar menambahkan atau melengkapi lingkungan nyata.

Tujuan utama dari AR adalah untuk menciptakan lingkungan baru dengan menggabungkan interaktivitas lingkungan nyata dan virtual sehingga pengguna merasa bahwa lingkungan yang diciptakan adalah nyata. Dengan kata lain, pengguna merasa tidak ada perbedaan yang dirasakan antara AR dengan apa yang mereka lihat/rasakan di lingkungan nyata. Dengan bantuan teknologi AR (seperti

visi komputasi dan pengenalan obyek) lingkungan nyata disekitar kita akan dapat berinteraksi dalam bentuk digital (virtual). Informasi tentang obyek dan lingkungan disekitar kita dapat ditambahkan kedalam sistem AR yang kemudian informasi tersebut ditampilkan diatas layer dunia nyata secara *realtime* seolah – olah informasi tersebut adalah nyata. Informasi yang ditampilkan oleh obyek virtual membantu pengguna melaksanakan kegiatan – kegiatan dalam dunia nyata. AR banyak digunakan dalam bidang – bidang seperti kesehatan, militer, industri manufaktur dan juga telah diaplikasikan dalam perangkat – perangkat yang digunakan orang banyak, seperti pada telepon genggam.

I.4.4 Tujuan *Augmented Reality*

Tujuan utama dari sistem AR akan berwujud sebagai sebuah kacamata atau proyektor retina yang akan menyediakan tampilan informasi yang relevan, dipetakan ke lingkungan sekitarnya secara *realtime* [28]. Misalnya, saat melihat sebuah restoran dengan kacamata AR, maka satu panggilan otomatis langsung ke *database review* atau menu dari *website* restoran tersebut. Seorang ilmuwan yang bekerja pada perusahaan farmasi bisa menggunakan kacamata untuk menampilkan model 3D dari berbagai molekul dan menggunakannya untuk memvisualisasikan obat yang lebih baik. Anak – anak mungkin menggunakan jaringan yang terhubung kacamata AR untuk bermain video game kehidupan nyata yang memungkinkan menembakkan “laser” dari tangan mereka, meski kemungkinan agak terbatas.

AR bergantung pada kemajuan teknologi miniaturisasi dan komputasi bergerak (*mobile computing*) [23, 24]. Saat ini, teknologi sistem proyeksi yang efektif dan komputer kecil dan cepat masih kurang matang untuk benar – benar dapat membuat antarmuka AR yang efektif, walaupun kita sudah semakin dekat pada kesempurnaan dengan riset yang dilakukan terus menerus oleh para ahli. Purwarupa telah banyak dibuat selama beberapa dekade, tetapi produk utama yang benar – benar layak belum ada. Salah satu teknologi yang menjanjikan adalah proyeksi retina, sehingga kacamata tidak digunakan lagi. Sistem proyeksi retina

komersial sebenarnya sudah ada, namun resolusi dan palet warnanya masih sangat rendah.

AR saat ini sebenarnya sudah ada dalam bentuk yang belum sempurna. Misalnya, komentator olahraga sering menggunakan pena cahaya untuk “menggambar” di lapangan sepak bola untuk memberikan bantuan visual untuk rekan komentatornya [8, 9, 15]. Contoh lain yaitu gambar iklan yang sering tampil dilapangan bulutangkis ketika ditayangkan di TV, seakan gambar iklan ini melekat pada lapangan. Gambar iklan sponsor ini terus berganti-ganti selama pertandingan. Ini menunjukkan bukti dari konsep AR meski masih terbatas.

I.4.5 Alur Kerja *Augmented Reality*

Alur kerja AR secara umum dimulai dari pengambilan gambar marker dengan kamera atau *webcam*. *Marker* tersebut dikenali berdasarkan *feature* yang dimiliki, kemudian masuk ke dalam *object tracker* (obyek yang dilacak) yang disediakan oleh *Software Development Kit* (SDK) [8, 9, 15]. Di sisi lain, *marker* tersebut telah didaftarkan dan disimpan kedalam *database*. *Object tracker* selanjutnya akan melacak dan mencocokkan marker tersebut agar dapat menampilkan informasi yang sesuai. Hasil keluaran pelacakan marker segera ditampilkan kedalam layar komputer dan layar ponsel cerdas. Informasi yang ditampilkan melekat pada marker yang bersangkutan secara *real time*.

1. Kamera

Digunakan untuk menscan *marker / markerless*.

2. *Frame*

Processing proses perhitungan *pixel* yang digunakan untuk pendeteksian gambar atau obyek.

3. *Database*

Digunakan untuk menyimpan data *marker*.

4. *Object Tracker*

Object yang dilacak yang akan di tampilkan informasi nya jika sesuai dengan *data* yang ada didalam *database*.

5. Informasi

Menampilkan informasi yang sesuai dengan *object tracker* yang sudah diinisialisasi di*database*.

6. *Screen*

Untuk menampilkan keluaran dari aplikasi yang berupa teks, gambar, video, dan object 3D.

I.4.6 Metode *Augmented Reality*

AR memiliki berbagai jenis metode yang semakin berkembang sampai saat ini. Dalam penelitian ini metode yang digunakan untuk marker ialah marker based tracking, berikut adalah penjelasan bagaimana marker based tracking.

Marker adalah suatu pola yang dibuat dalam bentuk gambar yang akan dikenali oleh *webcam*. *Marker* berfungsi sebagai pendefinisi dari AR. Informasi marker akan digunakan untuk mendefinisikan dan menampilkan sebuah obyek AR. *Marker* juga merupakan gambar yang terdiri atas *border outline* dan *pattern image* [8, 9, 15].

Marker biasanya merupakan ilustrasi hitam dan putih persegi dengan batas hitam tebal dan latar belakang putih. Komputer akan mengenali posisi dan orientasi *marker* dan menciptakan dunia virtual 3D yaitu titik (0,0,0) dan 3 sumbu yaitu X,Y,dan Z. Cara pembuatannya pun sederhana, tetapi harus diperhatikan ketebalan *marker* yang akan dibuat, ketebalan *marker* jangan kurang dari 25% dari panjang garis tepi agar pada saat proses deteksi *marker* dapat lebih akurat. Sedangkan obyek warna putih sebagai *background*, yang nantinya akan digunakan sebagai tempat obyek yang akan di-*render*.

Warna putih pada *marker* menunjukkan warna sebuah obyek, sedangkan warna hitam menunjukkan latar belakang. Intensitas warna pada suatu obyek memiliki warna yang lebih rendah (gelap), sedangkan latar belakang mempunyai intensitas yang lebih tinggi (terang). Adapun beberapa aturan umum dalam pembuatan pola *marker*, yaitu seperti harus kotak berbingkai hitam dan ini adalah rahasia dari pelacakan sebuah *marker*, ukuran tidak lebih dari 631x634 *pixel*. Warna selain hitam putih juga bisa dikenali oleh sistem marker membantu komputer dimana letak obyek akan ditampilkan.

Pendeteksian *marker* dikenal dua metode yaitu satu *marker* (*single marker*) dan banyak *marker* (*multi marker*). *Single marker* hanya mendeteksi satu gambar yang dijadikan sebagai media *marker* dan hanya satu obyek saja yang keluar. *Multimarker* yaitu metode yang memungkinkan pendeteksian banyak obyek yang dapat keluar dalam satu waktu pendeteksian *marker*. *Multimarker* merupakan perkembangan dari *single marker*, dimana kamera *tracking marker* lebih dari satu.

Ukuran *marker* yang digunakan dapat mempengaruhi penangkapan pola *marker* oleh kamera. Semakin besar ukuran *marker*, maka semakin jauh jarak yang dapat ditangkap oleh kamera dalam proses pendeteksian *marker*. Namun masalahnya, ketika *marker* bergerak menjauhi kamera, jumlah *pixel* pada layar kamera menjadi lebih sedikit dan ini bisa mengakibatkan pendeteksian tidak akurat.

Pelacakan AR berbasis penanda disajikan yang melacak dan mengidentifikasi penanda 2D yang diketahui secara real-time yang terdiri dari sudut untuk memperkirakan pose kamera yang akurat. Dalam sistem pelacakan marker, digunakan kamera untuk melacak marker yang telah dibuat. Obyek akan muncul ketika marker yang telah dibuat memiliki kualitas baik [8, 9, 15].

1. Face Tracking

Dengan menggunakan algoritma yang sedang dikembangkan, komputer dapat mengenali wajah manusia secara umum dengan cara mengenali posisi mata, hidung, dan mulut manusia, kemudian akan mengabaikan obyek – obyek lain di sekitarnya seperti pohon, rumah, dan benda – benda lainnya.

2. 3D Object Tracking

Berbeda dengan *face tracking* yang hanya mengenali wajah manusia, secara umum teknik 3D *object tracking* dapat mengenali semua bentuk benda yang ada disekitar, seperti mobil, meja, televisi, dan lain – lain.

3. Motion Tracking

Pada teknik ini komputer dapat menangkap gerakan. *Motion tracking* mulai digunakan secara ekstensif untuk memproduksi film – film yang mencoba mensimulasikan gerakan. Contohnya pada film *avatar the last airbender*, di mana *James Cameron* menggunakan teknik ini untuk membuat film tersebut dan menggunakannya secara *realtime*.

I.4.7 Unity 3D

Unity adalah *game developing software*, dengan *built-in IDE* yang dikembangkan oleh *Unity Technologies*. Hal ini digunakan untuk mengembangkan video game untuk *plugin web*, *platform desktop*, *konsol* dan perangkat *mobile*, dan digunakan oleh lebih dari satu juta pengembang [9, 10]. *Unity* tumbuh dari *OS X* didukung permainan alat pengembangan pada tahun 2005 untuk *game developing software game multi-platform*.

Mesin grafis menggunakan *Direct3D* (*Windows*, *Xbox 360*), *OpenGL* (*Mac*, *Windows*, *Linux*, *PS3*), *OpenGL ES* (*Android*, *iOS*), dan kepemilikan *API* (*Wii*). Ada dukungan untuk pemetaan *mesh*, pemetaan refleksi, pemetaan paralaks, bayangan dinamis menggunakan peta bayangan, merender ke tekstur dan efek *post-processing* layar penuh.

Unity mendukung aset seni dan format file dari *3DS Max*, *Maya*, *Softimage*, *Blender*, *modo*, *ZBrush*, *Cinema 4D*, *Cheetah3D*, *Adobe Photoshop*, *Adobe Fireworks* dan *Substansi Allegorithmic*. Aset ini dapat ditambahkan ke proyek *game*, dan dikelola melalui antarmuka pengguna grafis *Unity*.

Bahasa *ShaderLab* digunakan untuk *shader*, mendukung kedua deklaratif " pemrograman" dari program tetap fungsi pipa dan *shader* ditulis dalam GLSL atau Cg. *Shader A* dapat mencakup beberapa varian dan spesifikasi *fallback* deklaratif,

memungkinkan *Unity* untuk mendeteksi varian yang terbaik untuk kartu *video* saat ini, dan jika tidak ada yang kompatibel, jatuh kembali ke *shader* alternatif yang mungkin mengorbankan fitur untuk kinerja.

I.4.8 *Blender 3D*

Blender 3D adalah perangkat lunak visualisasi 3D yang mempunyai fitur cukup lengkap dan populer. *Software* ini bersifat *open source*, kualitas pencitraan digital tidak kalah dengan *software* grafis 3D lainnya seperti *3DS max*. *Blender 3D* dapat digunakan untuk membuat animasi 3D dan ada fitur tambahan yang membuat *software* ini semakin menarik yaitu bisa membuat sebuah *game engine* yang ada pada *software* ini [22, 28].

I.4.9 Bahasa Pemrograman *C Sharp*

C# atau yang dibaca *C sharp* adalah bahasa pemrograman sederhana yang digunakan untuk tujuan umum, dalam artian bahasa pemrograman ini dapat digunakan untuk berbagai fungsi misalnya untuk pemrograman *server-side* pada *website*, membangun aplikasi *desktop* atau *mobile*, pemrograman *game* dan sebagainya. Selain itu *C#* juga merupakan bahasa pemrograman berorientasi obyek, jadi *C#* mengusung konsep obyek seperti *inheritance*, *class*, *polymorphism*, dan *encapsulation* [26, 29].

Dalam prakteknya *C#* sangat bergantung dengan *framework* yang disebut *NET Framework*, *framework* inilah yang akan digunakan untuk meng-*compile* dan menjelaskan kode *C#*. *C#* dikembangkan oleh *Microsoft* dengan merekrut *Anders Helsing*, tujuan dibangunnya *C#* adalah sebagai bahasa pemrograman utama dalam lingkungan *NET Framework*. Banyak pihak yang mengatakan bahwa *Java* dan *C#* saling bersaing, bahkan ada juga yang mengatakan bahwa ketika kita bisa menguasai *Java* maka belajar *C#* akan sangat mudah.

I.4.10 *FASTcorner Detection*

Edward Rosten, Reid Porter, dan Tom Drummond mengembangkan algoritma FAST (Features from Accelerated Segment Test) corner detection, hal ini dimaksudkan untuk mempercepat waktu komputasi secara real-time, sehingga menghasilkan deteksi sudut yang kurang akurat. Vuforia menggunakan algoritma deteksi sudut FAST untuk menentukan bagaimana marker dilacak dan dideteksi pada Vuforia SDK. Alat tersebut menampilkan manajer target yang diunggah melalui Vuforia Web API [1, 16, 22] . Setelah kamera mendeteksi marker, maka dilakukan proses tracking untuk menentukan apakah marker tersebut cocok dengan marker yang telah dikofigurasi [16, 22]. Setiap marker yang diunggah memiliki peringkat mulai dari 0 hingga 5, dengan peringkat yang lebih tinggi menunjukkan kemampuan deteksi dan pelacakan yang lebih baik [4, 13, 17].

I.4.11 *Android SDK (Software Development Kit)*

Android SDK adalah *tools API (Application Programming Interface)* yang diperlukan untuk mulai mengembangkan aplikasi pada platform *Android* menggunakan bahasa pemrograman *Java*. *Android* merupakan *subset* perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di-*release* oleh *Google*. Saat ini disediakan *Android SDK (Software Development Kit)* sebagai alat bantu dan API untuk mengembangkan aplikasi pada platform *Android* menggunakan bahasa pemrograman *Java*. Sebagai platform aplikasi *netral*, *Android* memberi kesempatan untuk membuat aplikasi yang dibutuhkan yang bukan merupakan aplikasi bawaan *handphone / smartphone* [29].

I.4.12 *Vuforia*

Vuforia adalah sebuah software yang digunakan sebagai pengembangan Augmented Reality buatan Qualcomm. Dengan memanfaatkan sumber yang konsisten mengenai computer vision yang berfokus pada image recognition. Untuk membantu developer Vuforia menyediakan banyak kemampuan dan fitur [10, 30].

Dan memberi pengembang kebebasan untuk merancang dan membangun perangkat lunak sesuai dengan fungsionalitas yang memiliki fitur seperti berikut :

1. Teknologi computer vision canggih yang memungkinkan pengembang membuat efek khusus di perangkat.
2. Pengenalan terus menerus dari beberapa gambar.
3. Pelacakan dan deteksi lanjutan.
4. Solusi fleksibel untuk mengelola database gambar.

Vuforia memiliki dua alur kerja berbasis database yang dapat dipilih oleh pengembang , yaitu database cloud dan database local [2, 3]. Target pada *vuforia* merupakan obyek pada dunia nyata yang dapat dideteksi oleh kamera, untuk menampilkan obyek virtual. Beberapa jenis target pada *vuforia* adalah :

1. *Image targets*, contoh : foto, papan permainan, halaman majalah ,sampul buku, kemasan produk, poster, kartu ucapan. Jenis target ini menampilkan gambar sederhana dari *augmented*.
2. *Frame markers*, tipe *frame* gambar 2D dengan pattern khusus yang dapat digunakan sebagai potongan permainan di permainan pada papan.
3. *Multi-target*, contohnya kemasan produk atau produk yang berbentuk kotak ataupun persegi. Jenis ini dapat menampilkan gambar sederhana *Augmented 3D*.
4. *Virtual buttons*, yang dapat membuat tombol sebagai daerah kotak sebagai sasaran gambar.

Pada *Vuforia*, ada 2 jenis *workflow* dengan dasar database yang dapat dipilih oleh *developer*, yaitu *Cloud Database* dan *Device Database*.

I.4.13 Black-Box Testing

Pengujian *black box* fokus pada keperluan penelusuran kesalahan fungsional dari *software*. Ujicoba *black box* berusaha untuk menentukan kesalahan dalam beberapa kategori, diantaranya :

1. Fungsi-fungsi yang salah atau hilang
2. Kesalahan *interface*
3. Kesalahan dalam struktur *data* atau akses *database eksternal*.
4. Kesalahan *performa*.
5. Kesalahan inisialisasi dan terminasi.

Teknik pengujian *black box* terdiri dari 10 jenis diantaranya *Equivalence Partitioning, Boundary Value Analysis/Limit Testing, Comparison Testing, Sample Testing, Robustness Testing, Behavior Testing, Requirement Testing, Performance Testing, Endurance Testing, Cause-Effect Relationship Testing*. Salah satunya yang akan dibahas adalah *Equivalence partitioning* [30,31].

Equivalence partitioning merupakan metode ujicoba *black box* yang membagi domain input dari program menjadi beberapa kelas data dari kasus uji coba yang dihasilkan. Kasus uji penanganan *single* yang ideal menemukan sejumlah kesalahan (misalnya: kesalahan pemrosesan dari seluruh data karakter) yang merupakan syarat lain dari suatu kasus yang dieksekusi sebelum kesalahan umum diamati [30, 32].

