

BAB III

DASAR TEORI

Bab ini akan membahas uraian dasar teori yang akan digunakan penulis dalam melakukan perancangan dan pembuatan program yang dapat dipergunakan sebagai pembandingan atau acuan di dalam pembahasan masalah.

3.1. Pakaian Adat Indonesia

Menurut Cerdas Interaktif (2015), pakaian adat mencirikan setiap daerah dan memiliki model, warna, hiasan, dan motif yang berbeda-beda pula. Biasanya perbedaan itu muncul karena adanya percampuran budaya asli dengan budaya pendatang misalnya budaya India, Arab dan Cina.

Pada mulanya, pakaian berguna untuk menutupi atau melindungi bagian tubuh yang rentan terhadap lingkungan sekitar. Namun seiring berjalannya waktu, pakaian pun dijadikan sebagai kelengkapan untuk mengetahui status sosial, budaya, dan kondisi wilayah tertentu sehingga muncullah pakaian adat Indonesia yang beragam untuk setiap daerah.

Pakaian adat Indonesia memiliki fungsi yang beragam. Tidak hanya berfungsi sebagai pelindung atau sekedar identitas dari sebuah daerah di Indonesia saja. Pakaian adat Inonesia memiliki fungsi yang beragam, misalnya sebagai perlengkapan dalam pernikahan, upacara adat, dan sebagainya tetapi lebih menunjukkan unsur kekentalan budaya masing-masing.

Indonesia memiliki 34 provinsi, akan tetapi bukan berarti pakaian adat Indonesia hanya terdiri dari 34

jenis saja. Pakaian adat Indonesia di setiap provinsinya bisa lebih dari satu jenis dikarenakan kegunaan dalam proses adat di suatu daerah di provinsi tersebut. Pada gambar 3.1 terlihat contoh pakaian adat.



Gambar 3.1 Contoh pakaian adat Indonesia (As-salim, 2014)

3.2. *Augmented Reality*

Augmented Reality (AR) adalah suatu teknologi yang menggabungkan benda maya dua dimensi dan ataupun tiga dimensi ke dalam sebuah lingkungan nyata tiga dimensi lalu memproyeksikan benda-benda maya tersebut dalam waktu nyata (Mariyantoni, et al., 2014). Sedangkan menurut Sari, et al. (2014) *Augmented Reality* (AR) adalah sebuah teknologi yang pada awal dikembangkannya memiliki lingkup utama di "*Visual Augmentation*",

penambahan objek digital dalam. Secara sederhana *Augmented Reality* bisa didefinisikan sebagai lingkungan nyata yang ditambahkan objek *virtual*. Penggabungan objek nyata dan *virtual* ini dimungkinkan dengan teknologi *display* yang sesuai serta interaktivitas dimungkinkan melalui perangkat-perangkat *input* tertentu.

Augmented Reality (AR) merupakan kebalikan dari *Virtual Reality* (VR), tidak seperti VR yang sepenuhnya menggantikan lingkungan nyata, AR sekedar menambahkan atau melengkapi lingkungan nyata (Indrawaty, et al., 2014). Sistem AR memiliki tiga komponen utama, yaitu:

- a. *Tracking system* menentukan posisi dan orientasi objek-objek dalam dunia nyata.
- b. *Graphic system* menggunakan informasi yang disediakan *tracking system* untuk menggambarkan gambar-gambar *virtual* pada tempat yang sesuai, sebagai contoh melalui objek-objek nyata.
- c. Tampilan sistem menggabungkan dunia nyata dengan gambar *virtual* dan mengirimkan hasilnya ke pengguna (Kusnadi & Oktavianuddin, 2014).

Ada beberapa generasi *image* dari *Augmented Reality display* yang berhubungan dengan pengamatan pengguna, terhadap perangkat *Augmented Reality* yang digunakan.

- a. Pertama, *Retinal Display* generasi ini memungkinkan retina dari pengguna berhubungan langsung dengan perangkat *Augmented Reality*. Yang artinya perangkat tersebut terpasang di depan mata pengguna atau pengamat.
- b. Kedua, *Head Mounted Display* generasi *optic* ini hampir sama dengan generasi sebelumnya *retinal*

display. Pengguna atau pengamat memasang perangkat *Augmented Reality* dengan kepala mereka, namun tidak bersentuhan langsung dengan mata pengguna.

- c. Ketiga, *Hand-held Display*, generasi ini perangkat *Augmented Reality* benar-benar terlepas dari kepala pengguna atau pengamat. Perangkat tersebut berada di genggaman tangan pengguna. Dengan pergerakan menggunakan tangan, penglihatan pengguna tetap akan melakukan pengamatan. Generasi ini banyak digunakan dalam perangkat ponsel atau perangkat bergerak, perangkat ini menggunakan lensa dan layar sebagai penghubung pengguna dan objek yang diamati.
- d. Terakhir yang keempat, generasi ini perangkat *Augmented Reality* benar-benar telah terlepas dari tubuh pengguna dan mengintegrasikannya kedalam lingkungan nyata (Kurniawan, 2012).

Ada 2 buah metode *Augmented Reality* untuk menggabungkan objek nyata dan objek virtual yaitu:

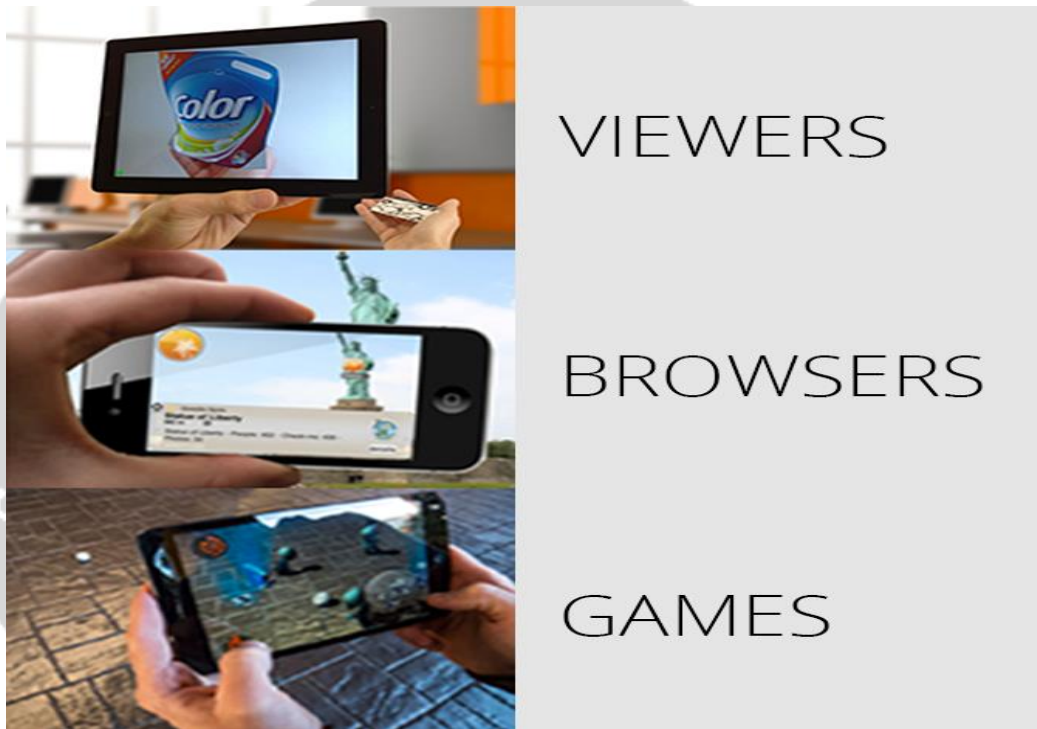
- a. *Augmented Reality* dengan *marker*

Merupakan metode *Augmented Reality* yang membutuhkan sebuah *marker* (gambar khusus) yang akan diproses dan dikenali oleh sistem sebagai alas atau permukaan yang akan memproyeksikan objek *virtual* hasil *Augmented Reality* di atas permukaan *marker* tersebut.

- b. *Markerless Augmented Reality*

Dengan metode ini pengguna tidak perlu lagi menggunakan sebuah *marker* untuk menampilkan elemen-elemen digital. Seperti yang saat ini dikembangkan oleh perusahaan Niantic yang bekerja sama dengan Nintendo, membuat sebuah aplikasi permainan yang

menggunakan teknologi *Augmented Reality* tanpa menggunakan *marker*. (Dinata, 2015). Pada gambar 3.2 terdapat berbagai macam jenis penggunaan teknologi *Augmented Reality*.



VIEWERS

BROWSERS

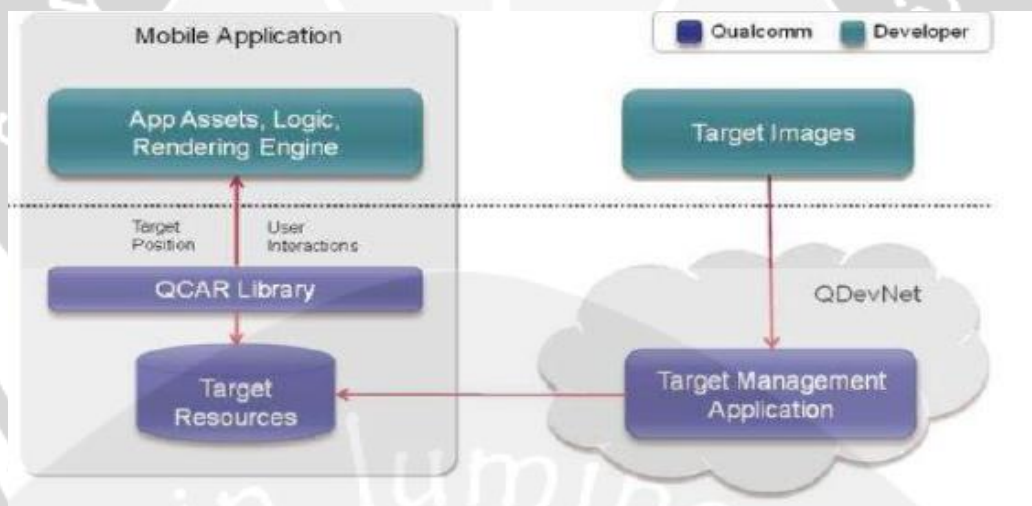
GAMES

Gambar 3.2 Jenis utama *Augmented Reality* (augment.com, 2016)

3.3. Vuforia

Vuforia adalah *Augmented Reality Software Development Kit (SDK)* untuk perangkat telepon genggam yang memungkinkan pembuatan aplikasi *Augmented Reality*. Dulunya lebih dikenal dengan *QCAR (Qualcomm Company Augmentend Reality)*. Ini menggunakan teknologi *Computer Vision* untuk mengenali dan melacak gambar planar (*Target Image*) 2D dan objek 3D sederhana seperti kotak, secara *real-time*. *SDK* Vuforia mendukung berbagai jenis target 2D dan 3D termasuk tanpa target gambar '*markerless*', 3D *Multi target* konfigurasi, dan bentuk

Marker Frame. Vuforia menyediakan *Application Programming Interfaces (API)* di *C++*, *Java*, *Objective-C*. Vuforia SDK mendukung pembangunan untuk *IOS* dan *Android* menggunakan Vuforia karena itu kompatibel dengan berbagai perangkat telepon genggam termasuk *iPhone*, *iPad*, dan telepon genggam *Android* serta *tablet* yang menjalankan *Android* sistem operasi versi 2.2 atau yang lebih besar dan prosesor *ARMv6* atau *7* dengan *FPU (Floating Point Unit)* kemampuan pengolahan. (qualcom.eu, 2015). Arsitektur *library QCAR SDK* dapat dilihat pada gambar 3.3.



Gambar 3.3 Arsitektur Library QCAR SDK (Dinata, 2015)

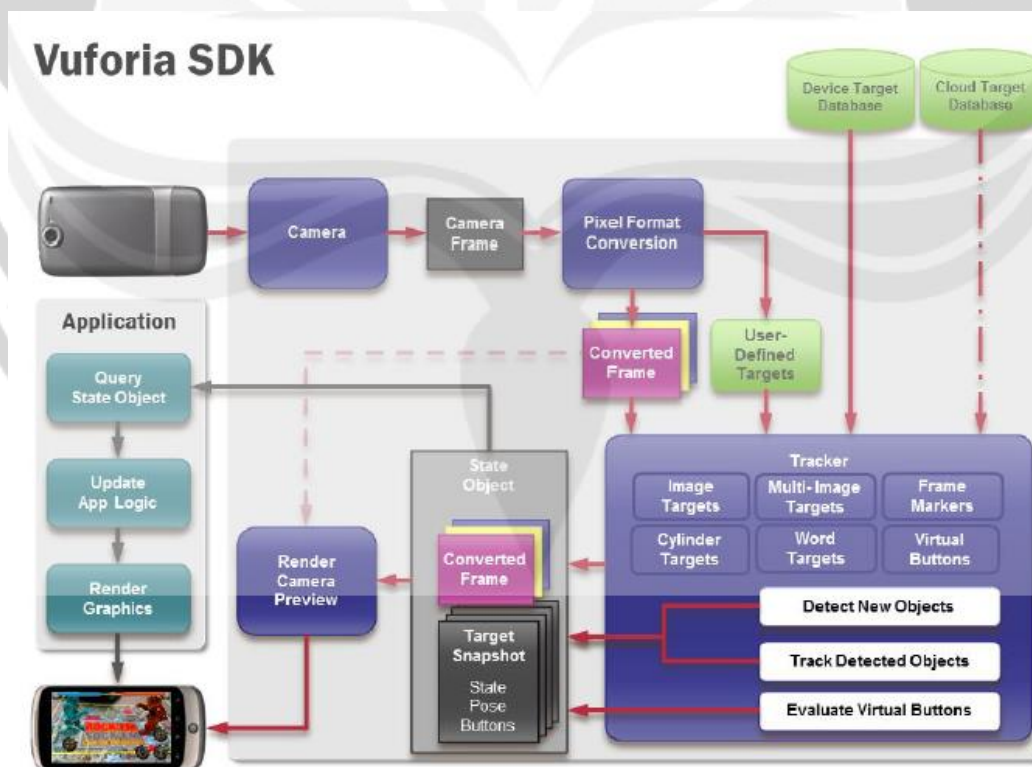
Dalam pengembangan aplikasi menggunakan Vuforia Qualcomm ini terdiri dari 2 komponen diantaranya adalah:

a. *Target Manager System*

Mengizinkan pengembang melakukan unggah gambar yang sudah diregistrasi oleh *marker* dan kemudian melakukan pengunduhan target gambar yang akan dimunculkan.

b. QCAR SDK Vuforia

Mengijinkan pengembang untuk melakukan koneksi antara aplikasi yang sudah dibuat dengan *library static i.e libQCAR.a* pada *IOS* atau *libQCAR.so* pada *Android*. Pembangunan aplikasi dengan *qualcomm Augmented Reality platform* yang terdiri dari *SDK QCAR* dan *Target System Management* yang dikembangkan pada *portal QdevNet*. *User* mengunggah gambar masukan untuk target yang ingin dilacak dan kemudian mengunduh sumber daya target, yang dibundel dengan *app*. *SDK QCAR* menyediakan sebuah objek yang terbagi *libQCAR.so* yang harus dikaitkan dengan aplikasi. (Dinata, 2015). Berikut diagram aliran data Vuforia dapat dilihat pada gambar 3.4.



Gambar 3.4 Diagram Aliran Data Vuforia (Dinata, 2015)

3.4. Marker

Proses pembuatan aplikasi *Augmented Reality* diperlukan sebuah *marker* sebagai penanda untuk menampilkan suatu objek. Dalam pembuatan *marker* diperlukan sebuah *file* gambar dengan ekstensi *JPG* yang nantinya akan diunggah ke situs resmi *Vuforia*. *Marker* yang telah diunggah akan dinilai kualitasnya oleh sistem, semakin banyak rating dengan tanda bintang maka kualitas *marker* akan semakin baik. *Marker* yang digunakan harus cenderung memiliki warna kontras untuk mendapatkan rating terbaik, *marker* yang buruk akan sulit di deteksi *device* atau bahkan tidak bekerja. Setelah semua *marker* berhasil diunggah, maka seluruh *marker* tersebut akan diubah menjadi sebuah *library marker* yang nantinya kembali harus diunduh agar dapat digunakan untuk proses pengkodean pada aplikasi *Unity* (Sari, et al., 2014).

3.5. Unity

Unity merupakan sebuah aplikasi yang terintegrasi untuk membuat bentuk objek dua dimensi (2D) maupun objek tiga dimensi (3D) pada *video games* atau untuk konteks interaktif lain seperti Visualisasi Arsitektur atau animasi 2D dan 3D *real-time*. Lingkungan dari pengembangan *Unity* berjalan pada *Microsoft Windows* dan *Mac Os X*, serta aplikasi yang dibuat oleh *Unity* dapat berjalan pada *Windows*, *Mac*, *Xbox 360*, *Playstation 3*, *Wii*, *iPad*, *iPhone* dan tidak ketinggalan pada *platform Android*. *Unity* juga dapat membuat game berbasis *browser* yang menggunakan *Unity web player plugin*, yang dapat bekerja pada *Mac* dan *Windows*, tapi tidak pada *Linux*.

Web player yang dihasilkan juga digunakan untuk pengembangan pada *widgets Mac*. Adapun fitur-fitur yang dimiliki oleh *Unity* antara lain sebagai berikut. a. *Integrated development environment (IDE)* atau lingkungan pengembangan terpadu. b. Penyebaran hasil aplikasi pada banyak *platform*. c. *Engine grafis* menggunakan *Direct3D (Windows)*, *OpenGL (Mac, Windows)*, *OpenGL ES (iOS)*, dan *proprietary API (Wii)* (Sari, et al., 2014).

3.6. Android

Android adalah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet yang bersifat *open source*. (Wahyutama, et al., 2013). Awalnya, *Google Inc.* membeli *Android Inc.*, pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan *Android*, dibentuklah *Open Handset Alliance*, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk *Google*, *HTC*, *Intel*, *Motorola*, *Qualcomm*, *TMobile*, dan *Nvidia* (Kusnadi & Oktavianuddin, 2014). Saat ini disediakan *Android SDK (Software Development Kit)* sebagai alat bantu dan *API (Application Programming Interface)* diperlukan untuk mulai mengembangkan aplikasi pada platform *Android* menggunakan bahasa pemrograman *Java* (Saputra, 2014). Sistem operasi yang mendasari *Android* dilisensikan dibawah *GNU, General Public Licence* versi 2 (GPLv2), yang sering dikenal dengan istilah "copyleft" lisensi dimana setiap perbaikan pihak ketiga harus terus jatuh dibawah *terms* (Perdana, et al., 2012). Basuki, et al.,

(2014) mengungkapkan bahwa perangkat berbasis *Android* OS dapat menjalankan *software* untuk menerapkan *Augmented Reality* system yang mampu mengidentifikasi pola citra satwa dan melakukan *tracking* terhadap citra secara *real time*. Terdapat beberapa versi pada *Android*, mulai dari versi 1.5 (*CupCake*), versi 1.6 (*Donut*), versi 2.1 (*Eclair*), versi 2.2 (*Froyo*), versi 2.3 (*GingerBread*), versi 3.0 (*HoneyComb*), versi 4,0 (*Ice Cream Sandwich*), versi 4.3 (*JellyBean*), hingga versi yang terbaru yaitu versi 4.4 (*KitKat*). (Dinata, 2015). *Android* memiliki arsitektur seperti pada gambar 3.5.



Gambar 3.5 Arsitektur Android (Dinata, 2015)

Telah dijelaskan berbagai teori yang melandasi penulis dalam membangun aplikasi yang nantinya akan dibuat. Bab selanjutnya akan menganalisis perancangan sistem yang dibangun.