

**IMPLEMENTASI TEKNOLOGI MARKERLESS AUGMENTED REALITY
MENGUNAKAN METODE ALGORITMA FAST
CORNER DETECTION BERBASIS ANDROID
(STUDI KASUS MULTIMEDIA BUKU INTERAKTIF
KEBUDAYAAN LOKAL KALIMANTAN BARAT)**

^[1]**Satria Islami Wahana Putra**

^[1]Jurusan Rekayasa Sistem Komputer, Fakultas MIPA Universitas Tanjungpura
Jalan Prof. Dr. H. Hadari Nawawi, Pontianak
Telp./Fax.: (0561) 577963
e-mail: ^[1]nama.wahana@studen.untan.ac.id

ABSTRAK

Buku merupakan salah satu media cetak yang sangat penting untuk dibaca, tidak terkecuali dengan buku kebudayaan lokal daerah. Dalam perkembangan media cetak, umumnya terkesan dan terlihat satu arah sebab hanya bisa dibaca dan terkadang pembaca tidak dapat menerima informasi lebih terutama informasi mengenai buku kebudayaan daerah karena terdapat hanya beberapa gambar di buku tersebut. Pada penelitian ini, Augmented Reality (AR) dapat diimplementasikan kedalam buku cetak yang dibuat dalam bentuk pengenalan budaya lokal Kalimantan Barat berbasis ponsel Android, sehingga terlihat lebih menarik dan atraktif untuk dibaca sebab pembaca juga dapat berperan dalam hal ini. AR dalam konsep markerless menawarkan keunggulan dalam penggabungan objek 3D dengan lingkungan realitas, namun hal tersebut harus diimbangi dengan proses pendeteksian marker sebagai objek yang ditampilkan. Pada metode FAST Corner detection yang digunakan, dibahas mengenai apa saja yang dapat mempengaruhi baik atau buruk dari implementasi AR. Sebab FAST merupakan algoritma dalam pengenalan objek 2D yang memanfaatkan tingkat kecerahan suatu objek gambar 2D yang akan dijadikan marker, sehingga setiap nilai dari titik sudut piksel gambar tersebut dapat dikenal oleh sistem. Untuk perolehan hasil dari aplikasi, kamera ponsel mendeteksi marker dengan sudut pandang terdeteksi diantara 0° - 75° dan kadar cahaya sedang pada jarak 0 - 30 cm.

Kata Kunci : *Kebudayaan, AR, metode FAST, marker 2D, Android.*

1. PENDAHULUAN

Sebagai negara yang terdiri dari puluhan bahkan ratusan suku bangsa dan terbagi dalam wilayah yang cukup banyak, tidak heran jika Indonesia memiliki kebudayaan yang begitu beragam. Seperti di daerah Kalbar, provinsi yang memiliki wilayah pemerintah berbatasan langsung dengan Sarawak, Malaysia ini merupakan salah satu wilayah dengan kekayaan budaya terbesar di Indonesia. Mulai dari rumah, pakaian, tarian, hingga berbagai hal yang menggambarkan Kalbar bisa dijumpai dengan mudah saat berkunjung ke tempat ini. Dengan perbagai kekayaan yang

diwariskan, banyak tidak diketahui sebagian masyarakat luas oleh sebab itu diperlukan upaya pelestarian budaya lokal dengan pengenalan informasi khalayak ramai secara luas.

Di era saat ini faktor dominan pada kehidupan masyarakat sangat tergantung erat pada perangkat teknologi, salah satunya perangkat ponsel pintar. Inovasi digital didalam fitur - fitur yang disajikan, memiliki sebuah tampilan menarik apabila dapat disandingkan dengan implementasi sistem perangkat lunak *Augmented Reality*. Sehingga dapat dijadikan sebuah salah satu keputusan

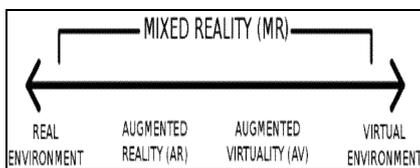
yang digunakan. Kemudahan akses dengan sebuah hasil informasi tak terbatas dan tampilan yang interaktif, menjadi solusi yang dimungkinkan.

Sebuah konsep ide dengan literatur studi *Implementation of AR system for Smartphone Advertisement* yang dilakukan seorang peneliti Y.Kim, dipaparkan proses AR sangat efektif bila representasinya pada sebuah media digital dalam sebuah produk yang dikenalkan. Proses yang cepat dalam tampilan objek animasi 3D dengan olah deteksi algoritma *FAST*, pada hasil studi yang diteliti dari *Machine learning for High-Speed Corner Detection* oleh E. Rosten, dihasilkan sebuah nilai ekstraksi dalam piksel gambar untuk sebuah kelayakan *marker* atau tanda dengan rancang gambar yang unik sebagai media cetak *binary*. Sehingga selain tampil kedalam perangkat lunak ponsel, juga dapat ditampilkan kedalam informasi baca yang dibukukan dengan menarik daya minat anak hingga dewasa dalam ilmu pengetahuan budaya lokal.

2. LANDASAN TEORI

2.1 Augmented Reality (AR)

Augmented Reality (AR) merupakan realitas tertambah dalam sebuah teknologi grafis dalam dunia nyata, dengan definisi gabungan sebuah objek dunia nyata dan maya yang interaktif pada olahan data waktu nyata [1]. Sedangkan definisi lain AR adalah sebuah rangkaian proses *reality-virtuality continuum*, dimana proses dari lingkaran nyata pada lingkungan virtual[2].



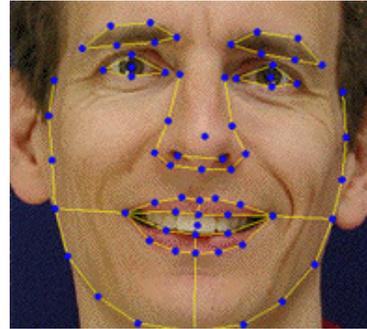
Gambar 1. Sistem virtual

2.1.1 Sistem Kerja AR

AR bekerja berdasarkan manipulasi citra pada perangkat optik elektronik dan komponen mekanik untuk membentuk citra dalam jalur mata pengguna pada objek fisik yang digabungkan.

2.1.2 Markerless AR

Konsep *markerless* AR adalah tanpa penggunaan frame sebagai tampilan objek 3D yang dikenal pada *marker image*. Sehingga konsep ini bebas dengan hanya melakukan pengulangan teknik seperti: *tracking face, object, motion*, dan *GPS*.

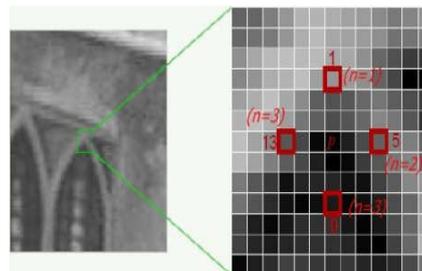


Gambar 2. Sistem virtual

2.2 Algoritma FAST

Deteksi sudut atau *corner detection* adalah suatu proses sistem visi komputer dalam mendeteksi sudut-sudut objek. Deteksi dari pendekatan ini digunakan untuk nilai khusus dari suatu objek seperti segi bentuk dan deteksi, sehingga definisi dari suatu sudut sendiri adalah haluan antara 2 sisi (*edge*) dengan nilai sudut sebagai titik dominan dan kebalikan arah dari titik tersebut. Pada skema algoritma kerjanya ditentukan sebuah titik sudut dengan asumsi calom titik variabel p dengan masukan data 16 piksel disekitar p . Terdapat kasus berbeda dalam hal ini penetapan untuk masing-masing perbedaan, yaitu:

$$C = \begin{cases} I_n < I_p - t & \text{gelap} \\ I_p - t < I_n < I_p + t & \text{normal} \\ I_p + t < I_n & \text{terang} \end{cases} \quad (1)$$



Gambar 3. FAST deteksi sudut per- titik

2.2.1 Vuforia SDK

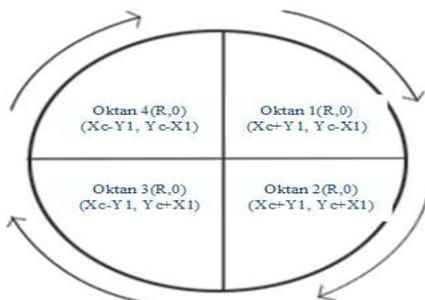
Pendukung dari sebuah aplikasi AR adalah *Vuforia* dimana sebagai wadah simpan atau *library* data citra gambar. Proses *Vuforia* dalam menyimpan data citra gambar atau *marker image* dengan menganalisa *image* menggunakan deteksi *marker* dengan hasil objek 3D yang dikenal oleh API (*application program interface*). Pada *image* tersebut diproses untuk menentukan titik sudut sesuai kriteria yang ditentukan.

Gambar	Sudut Terdeteksi	Keterangan
	4 sudut	sebuah persegi akan berisi fitur yang terdiri dari empat sudut
	0 sudut	sebuah lingkaran tidak berisi fitur karena tidak mengandung sudut
	2 sudut	objek ini hanya berisi fitur dengan dua sudut

Gambar 4. Kriteria titik sudut image

2.2.2 Algoritma Lingkaran Bressenhem

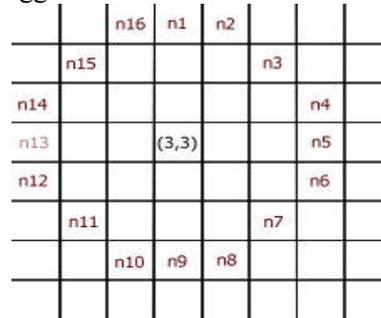
Algoritma lingkaran *midpoint* atau lingkaran *bressenhem* adalah generator lingkaran keliling yang dibuat efisien. Algoritma ini digunakan untuk membentuk semua titik berdasarkan nilai titik pusat dengan tambahan semua jalur dikisaran lingkaran. Algoritma ini diturunkan dalam bentuk garis, dimana proses garis-garis lingkaran 45° disetiap garis yang dihubungkan sehingga terbentuk lingkaran. Pada penentuan radius R dan titik pusat lingkaran dari titik koordinat (X_c, Y_c) ditentukan dengan rumus nilai oktan agar diperoleh sebuah titik koordinat $(X_1, Y_1) = (0, R)$.



Gambar 5. Rumus oktan lingkaran bressenhem

Proses nilai parameter dihitung $D_0=1-R$, dengan nilai tambah 1 di X_1 setiap proses dan

nilai Y_1 dikurang 1 setiap proses. Untuk keterangan gambar 5 setiap titik koordinat pada piksel n_1 hingga n_4 menggunakan rumus oktan 1, piksel n_5 hingga n_8 rumus oktan 2, piksel n_9 hingga n_{12} rumus oktan 3 dan piksel n_{13} hingga n_{16} rumus oktan 4.



Gambar 6. Nilai piksel n lingkaran bressenhem

Keterangan:

R = Jari-jari

X_c = Koordinat titik pusat X

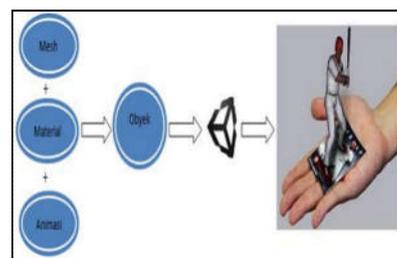
Y_c = Koordinat titik pusat Y

X_1 = Koordinat titik X piksel ke- n

D_x = Data keputusan

2.3 Unity 3D

Unity 3D adalah perangkat program pengolah sistem atau *engine system* yang akan dibangun dan terintegrasi oleh fitur pembuat konten 3D dengan kandungan olah visual interaktif sistem. Program pengolah sistem dengan gabungan adegan rancang dan lingkungan untuk membuat karakter sistem seperti: cahaya, audio, efek dari *image* virtual dan mendukung program tambahan diantaranya: ios, Android, desktop. Pada proses dasar, objek 3D (*mesh*) dan sistem angkut dari semua komponen lain dihubungkan kedalam material data sisipan dengan diolah (*render*) pada partikel data dengan penyatuan sistem.



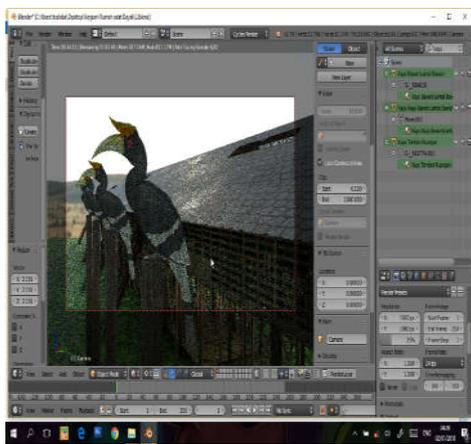
Gambar 7. Proses olah unity 3D

2.3.1 Bahasa Program C# (Sharp)

Bahasa program C# atau *python* adalah bahasa program yang berorientasi pada objek dari pengembang Microsoft sebagai rangka bagian NET *framework*. Dalam Unity, terintegrasi dengan teks editor yang tersedia berupa “MonoDevelop” dimana dapat menulis *script* di perangkat pengolah digital tersebut.

2.3.2 Blender 3D Studio

Perangkat pembuat gambar digital Blender 3D adalah mekanisme pembuatan objek berkaitan pada visualisasi objek 3D. Fitur – fitur yang ada didalam untuk kualitas grafis pada objek 3D agar terlihat hidup.



Gambar 8. Layar kerja blender 3D

2.4 Citra Digital

Citra (*image*) adalah gambar pada bidang dwimatra (dua dimensi). Ditinjau dari sudut pandang matematis, citra merupakan fungsi menerus (*continue*) dari intensitas cahaya pada bidang dwimatra. Sumber cahaya menerangi objek, objek memantulkan kembali sebagian dari berkas cahaya tersebut. Pantulan cahaya ini ditangkap oleh alat-alat optik, misalnya mata pada manusia, kamera, pemindai (*scanner*), dan sebagainya, sehingga bayangan objek citra tersebut terekam.

Agar dapat diolah dengan komputer digital, maka suatu citra harus direpresentasikan secara numerik dengan nilai-nilai diskrit. Representasi citra dari fungsi kontinu menjadi nilai-nilai diskrit disebut digitalisasi. Citra yang dihasilkan inilah yang disebut citra digital (*digital image*). Pada umumnya citra

digital berbentuk empat persegi panjang, dan dimensi ukurannya dinyatakan sebagai tinggi x lebar atau lebar x panjang.

2.5 Android

Perangkat *Android* merupakan sebuah perangkat sistem operasi untuk perangkat lunak ponsel, dan disertakan mesin virtual atau *middleware* dari proses sejumlah data aplikasi utama. Perangkat *Android* sendiri adalah modifikasi sistem dari kernel Linux yang ditujukan untuk pembuatan platform terbuka, dengan kemudahan pengembang untuk membuat aplikasi yang terkait dari batas minim dalam membangun.

Pada *Android* terdapat beberapa penunjang dari komponen-komponen terkait didalamnya, sebagai berikut:

1. Platform yang lengkap
2. Platform dengan operasi terbuka
3. Platform tidak berbayar

3. METODE PENELITIAN

Penelitian ini dimulai dengan studi pustaka yang dilakukan untuk memperoleh teori-teori pendukung yang berhubungan dengan pembuatan aplikasi AR. Teori-teori pendukung tersebut diperoleh dari beberapa sumber, yaitu buku, jurnal ilmiah, laporan penelitian, dan sumber-sumber tertulis lainnya yang berkaitan dengan sistem pada penelitian ini, seperti teori tentang AR jenis - jenis *markerless*, proses pola dari algoritma sistem, pengolahan data, dan bahasa perogram sistem.

Tahap selanjutnya adalah pengumpulan data yang dilakukan dengan mengumpulkan gambar sampel budaya dari aplikasi yang akan dijadikan objek *marker* 2D dan objek animasi 3D. Pada gambar - gambar tersebut meliputi gambar rumah dan pakaian adat, kerajinan dan senjata, tarian khas, makanan dan alat musik, dan destinasi wisata. Selanjutnya tahap analisis kebutuhan, meliputi kebutuhan mengenai objek yang akan diteliti, serta kebutuhan perangkat keras dan perangkat lunak untuk menunjang pembuatan aplikasi.

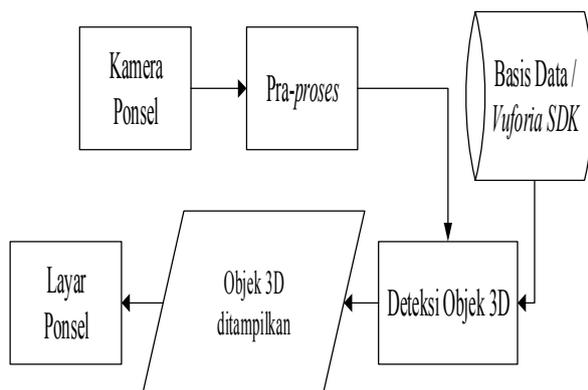
Tahap selanjutnya adalah perancangan sistem yang meliputi rancang data, rancang algoritma program yang digunakan pada kerangka dan antarmuka aplikasi. Aplikasi

dibuat menggunakan perangkat *Unity 3D* sebagai pengolah sistem digital, data *library Vuforia* sebagai basis data, dan metode FAST sebagai algoritma didalam pendeteksian.

Selanjutnya tahap implementasi atau tahap terapan sistem yang terkait, dilakukan dari dasar rancang program aplikasi dan rancang tampilan antarmuka aplikasi.

Selanjutnya tahap uji yang dilakukan adalah pengujian apakah sistem sudah bekerja berdasarkan langkah - langkah yang telah ditentukan. Tahap terakhir integrasi aplikasi yang akan dibangun dengan merealisasikan rencana penelitian ke dalam bentuk fisik yang nyata dengan melakukan integrasi kedalam perancangan perangkat lunak dan perangkat keras. selanjutnya sistem akan dikembangkan hingga berfungsi sesuai dengan tujuan pada penelitian.

3.1 Diagram Blok Sistem

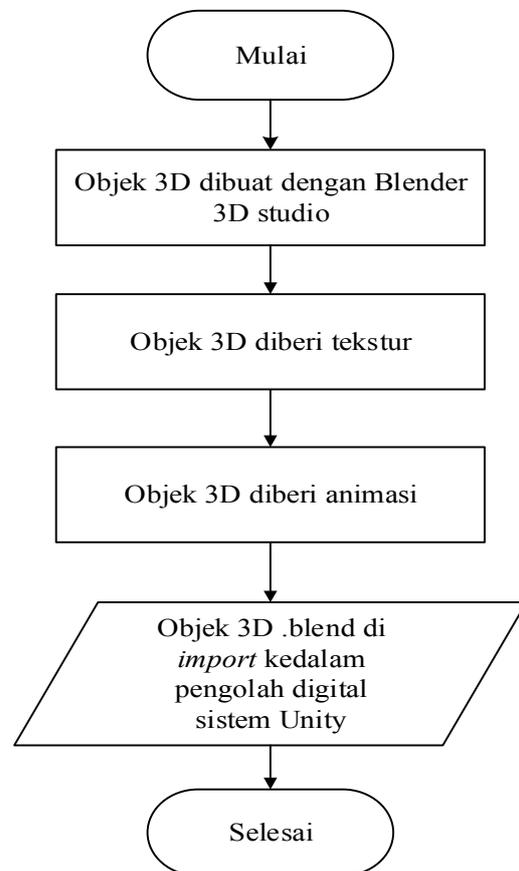


Gambar 9. Diagram blok sistem

Gambar 9 menunjukkan alur sistem yang bekerja pada aplikasi. Tahap pertama adalah pengambilan citra gambar atau *marker image* melalui kamera ponsel. Selanjutnya citra akan mengalami praproses meliputi tahap *grayscale* dan binerisasi. Tahap dua dalam sistem adalah ekstraksi fitur. Nilai hasil ekstraksi akan disimpan didalam data *library* untuk selanjutnya dilakukan proses deteksi. Tahap empat, nilai hasil dari proses deteksi dihasilkan sebuah informasi dari proses deteksi *marker image* dengan data hasil teks dan gambar pada layar ponsel *Android*.

3.2 Proses Pembuatan Objek 3D

Proses dari data terkait dapat dilihat pada *flowchart* pembuatan data objek 3D.

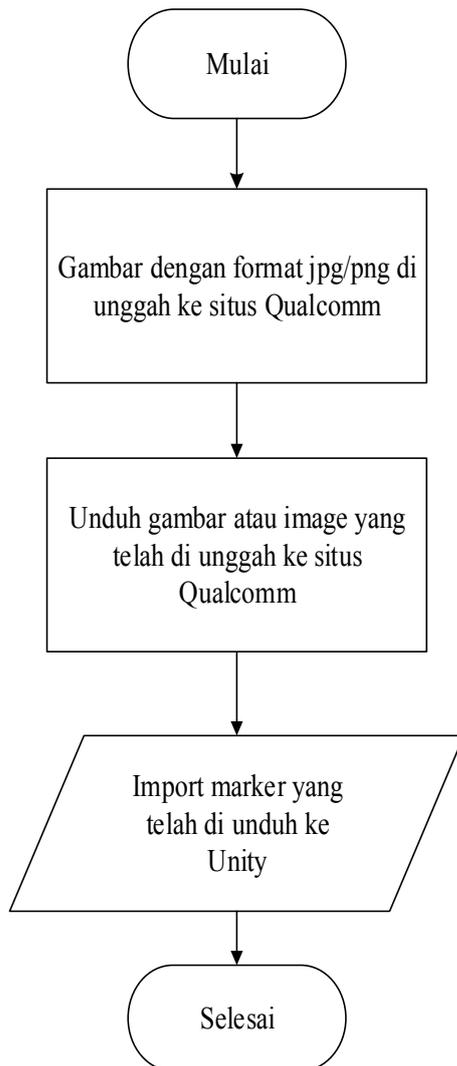


Gambar10. Flowchart pembuatan Objek 3D

Tahap proses awal adalah pembuatan data objek 3D dengan pembuatan model budaya yang sesuai dengan *marker image*. Selanjutnya pewarnaan pada tekstur model sehingga tampilan menarik dan diproses kedalam rendering model sehingga dapat sesuai dengan target objek 3D yang dihasilkan. Pada model data objek 3D yang diproses kedalam olahan sistem digital *Unity* adalah proses hasil format *.OBJ agar dapat diolah kedalam rangkaian sistem.

3.3 Flowchart Pembuatan Objek Marker

Proses *flowchart* pembuatan data objek *marker* dapat dilihat pada gambar 11.

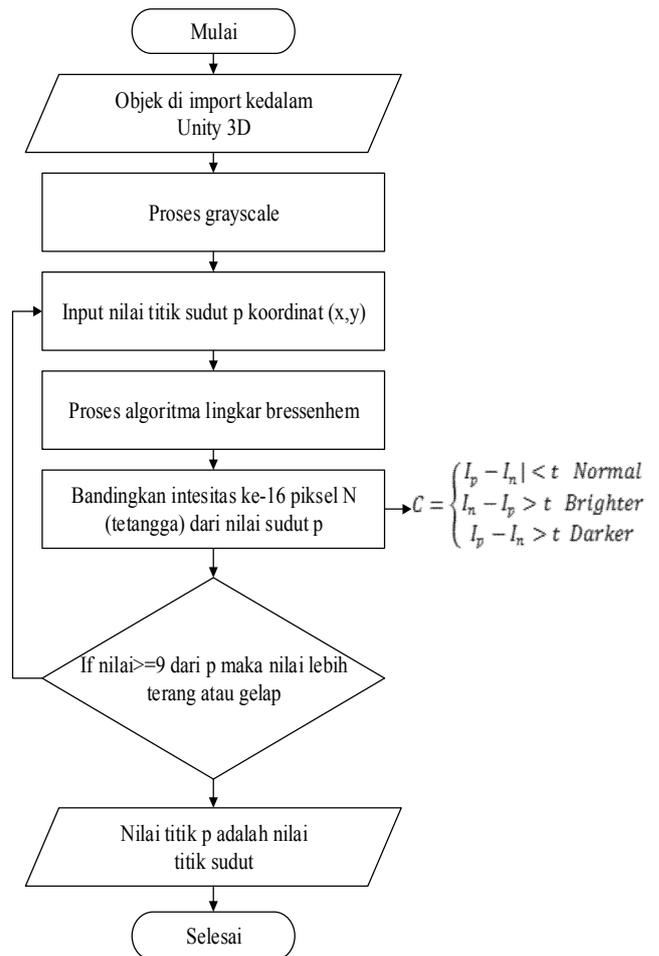


Gambar 11. Flowchart pembuatan marker 3D

Pada rancangan *flowchart* pembuatan marker, pada proses pembuatan digunakan sebuah aplikasi “image editing” photoshop. Data terkait selanjutnya disimpan dengan format PNG/JPG untuk diunggah ke dalam situs Qualcomm. Adapun penggunaan situs terkait untuk penyimpanan basis data yang dibuat.

3.4 Flowchart Algoritma FAST

Algoritma *FAST Corner Detection* yang diterapkan pada sistem aplikasi terdapat pada *flowchart* pendeteksian data pada objek yang ditargetkan sebagai acuan target manajer, atau dapat dilihat pada gambar 12.



Gambar 12. Flowchart algoritma FAST

Pada *flowchart* algoritma *FAST* data citra gambar di input sebagai langkah dalam pemrosesan *grayscale*. Proses *grayscale* ini dilakukan pada tahap pra-proses citra dimana citra asli dijadikan skala abu-abu dari gambar tersebut dengan nilai intensitas 0-255. Selanjutnya proses algoritma bresenham untuk dilakukan hasil perolehan nilai piksel dimana proses mengelilingi calon titik sudut p pada putaran 16 piksel di langkah 16 titik terdekat. Pada tahap ini pula dihasilkan data perbandingan nilai 16 piksel terdekat dengan nilai piksel sebagai parameter data tersebut. Dalam alur terkait, nilai yang dapat dijadikan parameter memiliki minimal 9 nilai terdekat yang bernilai gelap atau terang. Adapun

proses koordinat yang dilakukan memiliki radius jari-jari 3 dalam proses uji tersebut.

4. HASIL DAN PEMBAHASAN

4.1 Implementasi Antarmuka Aplikasi



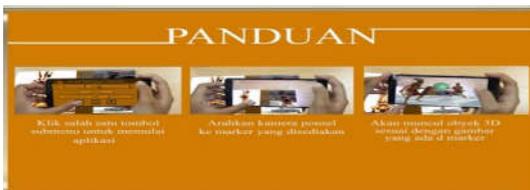
Gambar 13. Antarmuka halaman utama

Halaman utama adalah halaman yang digunakan sebagai halaman proses ketika aplikasi akan digunakan untuk deteksi objek. Untuk memilih citra uji pada target objek *marker* yang akan dideteksi, dapat dilakukan dengan dua cara, yaitu melalui menu pilih pada tombol menu atau *choose picture* dengan mengklik logo gambar yang tertera di layar. Tombol “Info” pada bagian bawah kiri layar aplikasi berfungsi sebagai keterangan dari judul aplikasi yang terkait. Tombol “Panduan” pada bagian tengah bawah layar aplikasi digunakan untuk panduan penggunaan aplikasi. Tombol “keluar” pada bagian bawah kanan layar aplikasi digunakan untuk keluar dari halaman aplikasi jika sudah selesai digunakan.



Gambar 14. Antarmuka halaman info

Halaman “Info” data merupakan halaman isi berkaitan dengan pembuat aplikasi dan tentang judul aplikasi yang dibuat.



Gambar 15. Antarmuka halaman panduan

Halaman “Panduan” data merupakan halaman panduan penggunaan pada aplikasi yang diiringi media buku interaktif *binary* kebudayaan Kalbar. Tampilan gambar yang dibuat agar *marker image* sesuai dengan objek 3D yang ditampilkan dalam data yang sudah disesuaikan. Untuk contoh sampel keterangan pada target deteksi *marker image* dalam menampilkan objek 3D dapat dilihat pada gambar 16.



Gambar 16. Antarmuka Halaman Data

4.2 Pengujian dan Analisa

Pengujian sistem bertujuan untuk data hasil sebuah kesalahan atau kekurangan pada perangkat lunak yang diuji. Pengujian yang dimaksud adalah perangkat lunak yang dibuat sudah memenuhi kriteria yang sesuai atau tidak. Data uji yang dilakukan pada fungsi sistem dari aplikasi dengan penggunaan tehnik *blackbox*, sehingga dapat diketahui hasil dari ada atau tidaknya kesalahan sistem.

4.2.1 Pengujian Fungsional Aplikasi

Tabel 1. Uji menu aplikasi

Kasus dan Hasil Uji			
Masukan	Keluaran yang diharapkan	Hasil Pengamatan	Keterangan
Memilih Menu Pindai Deteksi	Sistem dapat menampilkan hal. pindai deteksi	Sistem menampilkan hal pindai deteksi	[√] Berhasil [] Gagal
Memilih Menu Panduan	Sistem dapat menampilkan hal panduan pengguna	Sistem menampilkan hal. panduan pengguna	[√] Berhasil [] Gagal
Memilih Menu Info	Sistem dapat menampilkan hal. info	Sistem menampilkan hal info	[√] Berhasil [] Gagal
Memilih Menu Keluar	Sistem Dapat menampilkan hal keluar	Sistem menampilkan hal keluar	[√] Berhasil [] Gagal

Setelah uji menu aplikasi tidak ada kesalahan selanjutnya proses uji kembali pada halaman utama, dimana proses ini dapat kembali ke menu awal aplikasi. dapat dilihat pada tabel 2.

Tabel 2. Uji proses menu kembali

Kasus dan Hasil Uji			
Masukan	Keluaran yang Diharapkan	Hasil Pengamatan	Keterangan
Menekan Tombol Kembali di Hal. Pindai Deteksi	Sistem dapat keluar dan kembali ke menu utama	Sistem keluar dari hal. proses deteksi	[√] Berhasil [] Gagal

4.2.2 Pengujian Sudut dan Jarak

Uji sudut dan jarak untuk diketahui seberapa jauh rentang dan sudut pada pendeteksian objek, dapat dilihat pada tabel 3.

Tabel 3. Uji sudut & jarak

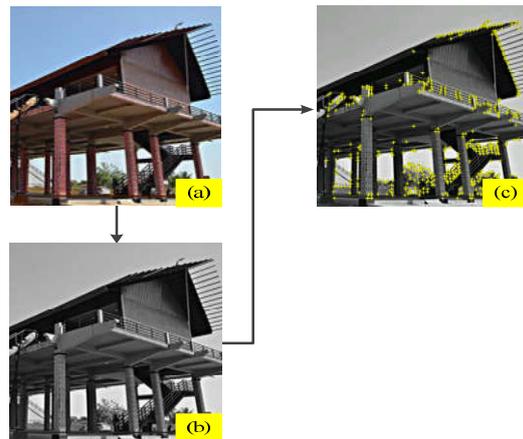
Kasus dan Hasil Uji				
Masukan		Keluaran	Keterangan	Hasil Uji
Jarak	Sudut			
5-20cm	75°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	60°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	45°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	30°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	15°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
20-30cm	75°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	60°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	45°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	30°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	15°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
30-40cm	75°	Deteksi objek 3D	Objek 3D Muncul	[] Terdeteksi [√] Gagal
	60°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	45°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	30°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	15°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
40-50cm	75°	Deteksi objek 3D	Objek 3D Muncul	[] Terdeteksi [√] Gagal
	60°	Deteksi objek 3D	Objek 3D Muncul	[] Terdeteksi [√] Gagal
	45°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal

Lanjutan Tabel 3. Uji sudut & jarak

50-65cm	30°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	15°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	75°	Deteksi objek 3D	Objek 3D Muncul	[] Terdeteksi [√] Gagal
	60°	Deteksi objek 3D	Objek 3D Muncul	[] Terdeteksi [√] Gagal
	45°	Deteksi objek 3D	Objek 3D Muncul	[] Terdeteksi [√] Gagal
	30°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
>70cm	15°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	75°	Deteksi objek 3D	Objek 3D Muncul	[] Terdeteksi [√] Gagal
	60°	Deteksi objek 3D	Objek 3D Muncul	[] Terdeteksi [√] Gagal
	45°	Deteksi objek 3D	Objek 3D Muncul	[] Terdeteksi [√] Gagal
	30°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal
	15°	Deteksi objek 3D	Objek 3D Muncul	[√] Terdeteksi [] Gagal

4.3 Tahap proses Citra Marker

4.3.1 Proses Citra Secara Visual



Gambar 17. Tahap proses citra

Pada gambar 17 tahap proses yang dilakukan terhadap sebuah data *marker image* dimana proses tersebut untuk diperoleh gambar yang layak dijadikan sebuah *marker*. Gambar (a) citra data masuk dengan format ekstensi .JPG dengan latar belakang berwarna. Gambar (b) Proses *grayscale* pada gambar untuk dijadikan sebuah *marker*. Gambar (c) hasil dari proses yang terkait dengan taburan nilai ekstraksi pada deteksi objek yang sudah layak dijadikan *marker*.

4.3.2 Proses Citra Secara Manual



Tanda panah yang di hubungkan dari koordinat yang di ambil untuk citra asli yang di wakili oleh 3 byte, yaitu Red, Green, dan Blue.

Np R=129 G=148 B=152	N1 R=207 G=228 B=229	N2 R=214 G=238 B=238	N3 R=210 G=231 B=239	N4 R=183 G=199 B=199
N5 R=193 G=209 B=209	N6 R=216 G=232 B=232	N7 R=226 G=244 B=248	N8 R=232 G=250 B=252	N9 R=205 G=221 B=221
N10 R=162 G=176 B=176	N11 R=90 G=102 B=102	N12 R=171 G=183 B=183	N13 R=193 G=204 B=206	N14 R=211 G=225 B=228
N15 R=188 G=203 B=206	N16 R=187 G=207 B=208			

Gambar 18. Potongan nilai piksel citra RGB

Pada gambar 18 sampel piksel nilai RGB dengan koordinat titik yang ditujukan pada tanda anak panah. Pada proses RGB diubah ke proses grayscale dengan cara hitung rata-rata elemen warna RGB dengan cara berikut:

$$f_o(x, y) = \frac{129 + 148 + 152}{3}$$

$$f_o(x, y) = 143$$

Penggunaan cara yang sama dari nilai warna RGB lain dengan hasil sebagai berikut:

143	222	231	225	194
204	227	245	216	216
172	98	179	201	221
199	201			

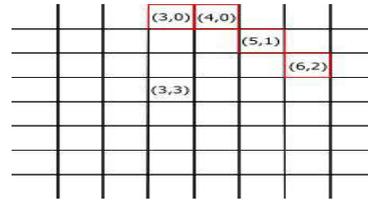
Gambar 19. Nilai piksel grayscale

Selanjutnya proses algoritma bresenhem digunakan untuk proses pelingkaran koordinat sebesar 16 piksel dari jarak nilai terdekat, hal ini sebagai tahap mencari calon titik sudut piksel dari gambar.

Tabel 4. Algoritma bresenhem oktan 1

Keputusan	X1	Y1	Koordinat Titik Dekat (Xc+X1, Yc-Y1)
D0=1-R	0	R	(3, 0)
D0= 1-3 = -2	0	3	(3, 0)
D1=D0 + 2*(X1) + 3	1	3	(4,0)
D1= -2 + 2*(1) + 3 = 3	1	3	(4,0)
D2=D1 + 2*(X1-Y1) + 5	2	2	(5,1)
D2= 3 + 2*(2 - 2) + 5 = 8	2	2	(5,1)
STOP	3	1	(6,2)

Tabel 4 adalah proses awal pembentukan titik koordinat lingkaran dari 1 hingga titik 4 sebesar rentang 45°. Dapat dilihat hasil pada gambar 20.



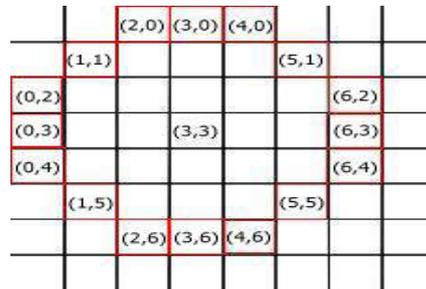
Gambar 20. Proses awal bentuk titik lingkaran

Proses diatas terus - menerus dilakukan hingga membentuk sebuah poros lingkaran putar sebesar 16 piksel. Adapun proses yang dilakukan hingga titik terakhir dilihat pada tabel 5 dan gambar 21 berikut.

Tabel 5. Algoritma bresenhem oktan 4

Keputusan	Y1	X1	Koordinat Titik Dekat (Xc-Y1, Yc-X1)
D0=1-R	3	0	(0, 3)
D0= 1-3 = -2	3	0	(0, 3)
D1=D0 + 2*(X1) + 3	3	1	(0,2)
D1= -2 + 2*(1) + 3 = 3	3	1	(0,2)
D2=D1 + 2*(X1-Y1) + 5	2	2	(1,1)
D2= 3 + 2*(2 - 2) + 5 = 8	2	2	(1,1)
STOP	1	3	(2,0)

Tabel 5 adalah proses akhir dari pembentukan titik koordinat 13 hingga titik 16, sehingga terbentuk lingkaran 16 piksel. Pada penjelasan terkait, dapat dilihat pada gambar 19.



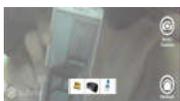
Gambar 21. Nilai titik lingkaran terbentuk

Pada nilai yang dicari dari proses *grayscale* dititik koordinat terdekat atau *neighbor* 1 hingga 16 menggunakan rumus persamaan 2.2 dimana untuk diketahui syarat minimal nilai dari intensitas gelap ataupun terang. Kriteria yang dimungkinkan untuk syarat minimal 9 nilai untuk terbentuknya nilai sudut deteksi.

4.3.3 Proses Citra Secara Intesitas

Proses citra secara intensitas cahaya adalah proses yang dilakukan terhadap intensitas cahaya pada ruang uji. Parameter yang digunakan untuk proses tersebut adalah ruang terang dan redup, sehingga hasil yang diperoleh pendeteksian *marker* dengan hasil target objek yang dicapai.

Tabel 6. Proses citra secara intensitas

Kondisi	Hasil Pengujian	
	Keluaran hasil	Keterangan
Ruang Terang		Deteksi sangat cepat. Salah satu objek 3D budaya dapat ditampilkan dengan baik.
Ruang Redup		Tidak terdeteksi. Sebab target <i>marker image</i> sulit dikenali dalam keadaan redup cahaya

5. KESIMPULAN

Kesimpulan yang dapat diambil dari hasil uji dan analisis yang telah dilakukan pada aplikasi AR kebudayaan lokal Kalbar sebagai berikut:

1. Implementasi penelitian ini berhasil diuji aplikasi visual 3D budaya lokal menggunakan teknik *markerless* pada metode FAST, sehingga *marker* yang digunakan dapat bebas tanpa diharuskan membuat pola atau barcode.
2. Berdasarkan uji deteksi *marker* diantara jarak terdekat adalah 20cm dan jarak terjauh kurang lebih 70cm. pada sudut pandang yang dimiliki, rentang sudut diantara 30° hingga 75°, namun masih dapat dipengaruhi oleh jarak.
3. Persentasi keberhasilan dari uji data 30 deteksi objek adalah sebesar 80%.
4. Berdasarkan data uji, cahaya ruang dipengaruhi deteksi *marker*. Ruang gelap

tidak dapat digunakan untuk melakukan uji coba deteksi *marker*.

5. Berdasarkan penelitian penentuan titik sudut pada *marker* dilakukan dengan perbandingan nilai intensitas 16 neighbor pada nilai intensitas titik pusat (sudut), sehingga titik sudut tercipta sekurangnya memiliki 9 data nilai intensitas *neighbor* yang gelap atau terang dari intensitas tersebut.

6. SARAN

Adapun saran untuk pengembangan penelitian selanjutnya adalah sebagai berikut:

1. Pada penelitian selanjutnya dapat dilakukan dengan mengubah warna objek 3D budaya sehingga dapat lebih menarik lagi.
2. Pada penelitian selanjutnya, dapat dikembangkan lagi masalah yang berkaitan. Sehingga dapat menjadi ajang media promo pariwisata daerah.
3. Aplikasi AR dapat dikembangkan pada bidang lain, seperti kesehatan, militer, pembelajaran dan lainnya. Juga dapat dikembangkan kedalam platform lain seperti Windows phone, iOS, dekstop dan perangkat lainnya.

DAFTAR PUSTAKA

- [1] Azuma, R.T (1997). Presence: Teleoperators and Virtual. *A Survey of Augmented Reality*, 355-385.
- [2] Kim, Y.G, & Kim, W.J. (2014). Implementation of Augmented Reality System for Smartphone Advertisement. *Journal of Multimedia and Ubiquitous Engineering*, 385-392.
- [3] Putra, D. (2010). Binerisasi Citra Tangan dengan Metode Otsu. *Jurusan Teknik Elektro*, 11-13.
- [4] Rosten, E. (2006). FAST Corner Detection. *Machine Learning for High-Speed Corner Detection*, 1-14.
- [5] Afissunani, A., A., Saleh,A., & Assidiqi, M.H. (2012). MULTI TARGET MARKER AUGMENTED REALITY UNTUK APLIKASI MAGIC BOOK *Engineering*, 1-4.