

Pemanfaatan Augmented Reality Pada Permainan Othello

Rendy Layman Aguston
Program Studi Teknik Informatika
Institut Informatika Indonesia
rendy_lay@hotmail.com

Hermawan Andika
Program Studi Teknik Informatika
Institut Informatika Indonesia
andi@ikado.ac.id

Edwin Meinardi Trianto
Program Studi Teknik Informatika
Institut Informatika Indonesia
edwin@ikado.ac.id

Abstrak - Perkembangan teknologi telah mengubah cara pengerjaan suatu pekerjaan dari cara konvensional menjadi cara yang lebih praktis. Dengan hadirnya teknologi *Augmented Reality*, cara bermain yang menggunakan pion dan membalikkan pion musuh secara manual menjadi lebih mudah dalam memainkan permainan selain juga dapat berinteraksi langsung. Pembuatan permainan Othello menggunakan program Unity dengan *framework* Vuforia untuk mewujudkan *Augmented Reality* pada permainan Othello. Untuk menerapkan *Augmented Reality* dengan baik, dibutuhkan papan permainan sebagai *image target* yang sesuai dengan kriteria, jenis kamera yang digunakan, jarak kamera terhadap papan permainan, intensitas cahaya yang ditangkap kamera, serta tingkat sensitivitas tombol virtual. Pada permainan Othello ini tersedia fitur komputer yang menggunakan algoritma Alpha Beta Pruning dengan 3 level kedalaman yang menggunakan perhitungan fungsi evaluasi berupa *mobility*, *potential mobility* dan penguasaan *corner* yang menghasilkan kemenangan mencapai 73,33% dari 15 kali uji coba terhadap aplikasi Othello serupa dan 78,34% dari 37 kali uji coba terhadap *user*.

Kata Kunci : *Game, Othello, Augmented Reality, Image Target, Virtual Button, Alpha Beta Pruning, Unity, Vuforia.*

I. PENDAHULUAN

Augmented Reality merupakan teknologi yang sedang populer di dunia yang memproyeksikan benda dua dimensi ataupun tiga dimensi ke dalam sebuah lingkungan nyata. Perkembangan *Augmented Reality* sangatlah pesat karena teknologi yang dihadirkan dapat memvisualisasikan obyek maya yang telah dirancang untuk ditampilkan sehingga terlihat tampak nyata. Selain menghadirkan inovasi baru dalam pengaplikasiannya, *Augmented Reality* juga menghadirkan dimensi baru yaitu melakukan deteksi obyek yang sudah ditentukan terlebih dahulu serta mengizinkan pengguna melakukan interaksi terhadap aplikasi tersebut.

Pada saat ini, perkembangan *Augmented Reality* telah dimanfaatkan untuk pembuatan permainan *mobile*. Banyak aplikasi permainan yang telah dibuat dengan menggunakan konsep *Augmented Reality*. Salah satu permainan yang cukup

populer adalah permainan Othello. Permainan Othello cukup mudah dimainkan dengan membutuhkan pion putih dan pion hitam yang dimainkan pada suatu papan permainan kotak-kotak berukuran 8x8. Memainkan permainan othello sangatlah mudah, dengan empat pion (2 pion putih dan 2 pion hitam) ditengah pada awal permainan, pemain akan berusaha menguasai pion lebih banyak sampai seluruh isian papan telah terisi pion atau musuh tidak memiliki pion yang terdapat pada papan permainan.

Dengan latar belakang tersebut, penulis tertarik untuk mendalami pembuatan permainan Othello dengan menggunakan teknologi *Augmented Reality* yang dipresentasikan melalui proyektor.

II. METODOLOGI PENELITIAN

A. *Augmented Reality*

Augmented Reality merupakan teknologi yang menggabungkan benda maya dua dimensi dan ataupun tiga dimensi ke dalam sebuah lingkungan nyata tiga dimensi lalu memproyeksikan benda-benda maya tersebut secara *real-time*. Tiga dimensi biasa disingkat 3D atau disebut ruang dari benda yang memiliki panjang, lebar, dan tinggi. Istilah ini biasanya digunakan dalam bidang seni, animasi, komputer dan matematika. Benda-benda maya menampilkan informasi berupa label maupun obyek *virtual* yang hanya dapat dilihat dengan kamera *handphone* maupun dengan komputer. Sistem dalam *Augmented Reality* bekerja dengan menganalisa secara *real-time* obyek yang ditangkap dalam kamera [1].

B. Permainan Othello

Permainan Othello adalah permainan yang dimainkan oleh dua orang pemain pada sebuah papan permainan. Papan permainan yang digunakan berupa petak-petak kecil (kotak persegi) berukuran 8 baris dan 8 kolom. Setiap pemain bergerak berdasarkan giliran. Pada awal permainan akan diletakkan 2 pion hitam dan 2 pion putih ditengah papan. Pemain akan saling berganti giliran untuk meletakkan pion hingga tidak ada langkah yang dapat ditempati oleh kedua pemain.

C. Alpha Beta Pruning

Alpha Beta Pruning adalah algoritma pencarian pada sebuah pohon yang menelusuri pohon permainan dengan

meletakkan 2 nilai pada setiap *node*, yaitu alpha (α) dan beta (β). Nilai α ditetapkan sama dengan $-\infty$ sedangkan nilai β sama dengan $+\infty$. Jika $\alpha < \beta$, maka kesempatan untuk mencari langkah terbaik masih ada dan pencarian akan tetap dilanjutkan. *Node* akan melakukan maksimalisasi dan memperbaiki nilai α dari nilai anak-anaknya, kemudian nilai yang telah memperbaiki nilai α tersebut akan dibandingkan dengan nilai β sementara. Jika $\alpha > \beta$, maka evaluasi dihentikan [2].

D. Unity 3D

Unity merupakan salah satu dari *game engine* untuk membuat bentuk obyek 3D pada *video game*. Dengan menggunakan *software* ini, *developer* dapat membuat *game* dengan lebih mudah dan cepat. Pengembangan Unity 3D ini dapat berjalan di Windows dan Mac OS, sedangkan permainan yang dibuat dapat digunakan di berbagai *platform* seperti Windows, Mac, Xbox 360, Playstation 3, Nintendo Wii, iPad, iPhone, dan Android.

E. Qualcomm Vuforia

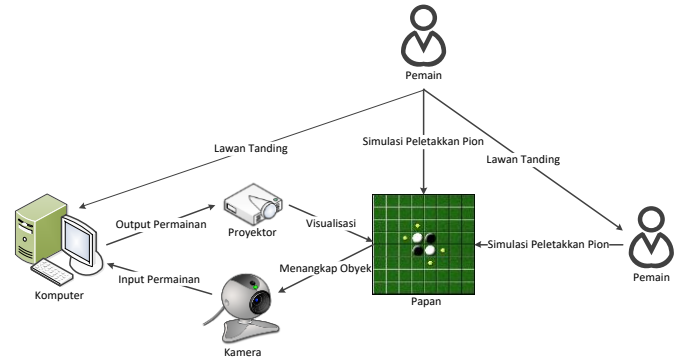
Qualcomm Vuforia adalah *Augmented Reality Software Development Kit* (SDK) yang di gunakan pada perangkat *mobile* untuk pembuatan aplikasi *Augmented Reality*. SDK ini menggunakan teknologi *computer vision* untuk mengenali dan melacak gambar target dan obyek 3D yang sederhana secara *real time*. Dengan kemampuan ini, memungkinkan para *developer* untuk membuat posisi dan orientasi obyek *virtual* seperti model 3D dan media-media lainnya yang berkaitan dengan dunia nyata dapat dilihat melalui kamera dari perangkat *mobile*. Obyek *virtual* tersebut selanjutnya melacak posisi orientasi dari gambar secara *real-time* sehingga perspektif dari pengguna pada obyek tersebut sesuai dengan perspektif mereka pada target gambar, sehingga terlihat bahwa obyek *virtual* tersebut adalah bagian dari dunia nyata [3].

III. HASIL DAN PEMBAHASAN

Sebelum dilakukan implementasi program, perlu dilakukan analisa dan desain sistem untuk mempermudah implementasi program karena sebagai acuan untuk menghasilkan program yang baik.

A. Desain Arsitektural Permainan

Pada desain arsitektural ini akan dijelaskan mengenai alur kerja dari permainan yang dibuat. Alur kerja ini berisi semua interaksi yang ada pada permainan antara pemain, komputer dengan papan permainan dimana membutuhkan bantuan kamera dan proyektor. Secara garis besar, alur sistem permainan dapat dilihat pada gambar berikut.

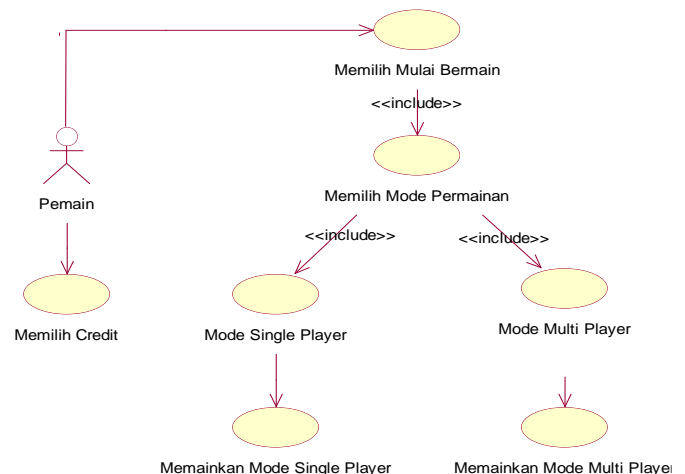


Gambar 1. Desain Arsitektur Permainan Othello

Pemain merupakan pihak utama dalam memainkan permainan Othello ini dimana perangkat penjalan utama menggunakan komputer. Papan yang digunakan merupakan papan fisik yang dapat dibuat dari selembar kertas yang nantinya akan rekam oleh sebuah kamera untuk menangkap interaksi yang terjadi pada papan permainan. Hasil penangkapan obyek tersebut akan diproses oleh komputer yang selanjutnya hasil proses tersebut akan ditampilkan menggunakan sebuah proyektor. Pemain akan berinteraksi pada papan permainan itu.

B. Use Case Diagram

Use Case Diagram menjelaskan urutan kegiatan yang dilakukan aktor dan sistem untuk mencapai tujuan tertentu, menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. *Use Case Diagram* juga serta mempresentasikan interaksi antar aktor/pelaku dengan sistem [4]. Berikut *Use Case Diagram* yang digunakan pada permainan Othello.



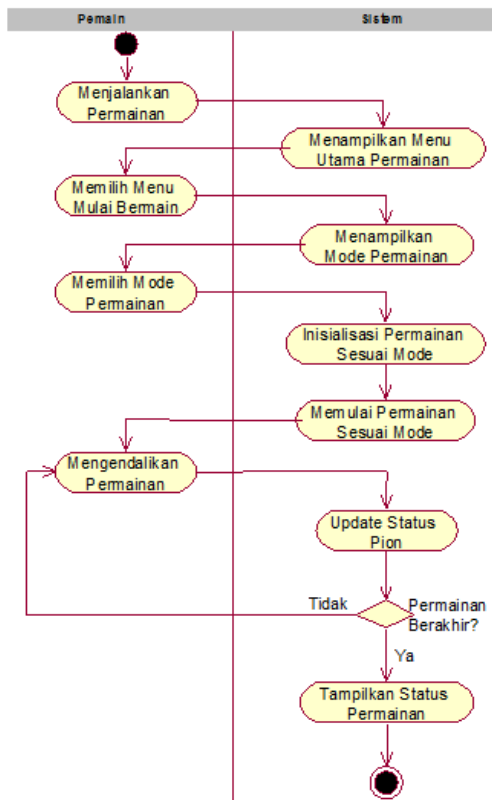
Gambar 2. Use Case Diagram Permainan Othello

C. Activity Diagram

Activity diagram adalah representasi grafis dari seluruh tahapan alur kerja. Diagram ini mengandung aktivitas, pilihan tindakan, perulangan dan hasil dari aktivitas. *Activity diagram* tidak menggambarkan *behavior internal* sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi

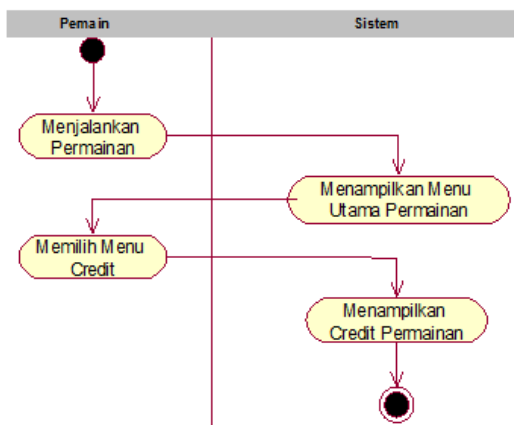
lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. *Activity diagram* juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. *Activity diagram* dibuat berdasarkan sebuah atau beberapa *use case* pada *use case diagram* [4] Berikut *activity diagram* yang terdapat pada permainan Othello.

a) *Activity Diagram* Permainan Othello



Gambar 3. *Activity Diagram* Permainan Othello

b) *Activity Diagram* Menu Credit



Gambar 4. *Activity Diagram* Menu Credit

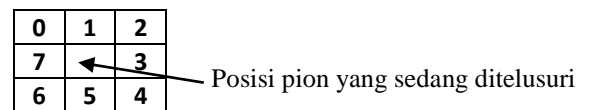
D. *Game Play* Permainan

- **Inisialisasi Permainan**
 Untuk dapat mengetahui data papan sehingga dapat digunakan untuk pengendalian permainan, diperlukan inisialisasi papan permainan. Inisialisasi papan permainan digunakan untuk mengetahui semua data pion di papan sehingga dapat diproses untuk pencarian langkah yang tersedia dan dapat digunakan untuk penelusuran algoritma *alpha beta pruning*. Berikut inisialisasi nilai awal permainan dimana nilai 0 sebagai kotak kosong, nilai 1 sebagai pion hitam dan nilai 2 sebagai pion putih.

I\J	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	2	1	0	0	0
4	0	0	0	1	2	0	0	0
5	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0

Gambar 5. Inisialisasi Nilai Awal Permainan

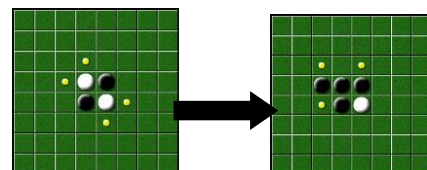
- **Pencarian Langkah Tersedia**
 Pengecekan data papan untuk memperoleh langkah yang tersedia akan membantu pemain dalam menentukan pilihan peletakkan pion. Selain itu juga membantu pemain untuk menghindari kesalahan peletakkan pion pada papan permainan. Penentuan arah tetangga diperoleh dari analisa pengecekan tetangga seperti pada gambar berikut.



	0	1	2	3	4	5	6	7
Horizontal	-1	0	1	1	1	0	-1	-1
Vertikal	-1	-1	-1	0	1	1	1	0

Gambar 6. Analisa Cek Tetangga

- **Penempatan Pion dan Reversi**
 Ketika pemain meletakkan pion sebagai langkah yang dipilih, sistem akan mengubah kotak tersebut menjadi pion pemain. Sistem juga mengecek semua tetangga untuk menelusuri ada pion musuh yang dapat diubah atau tidak.



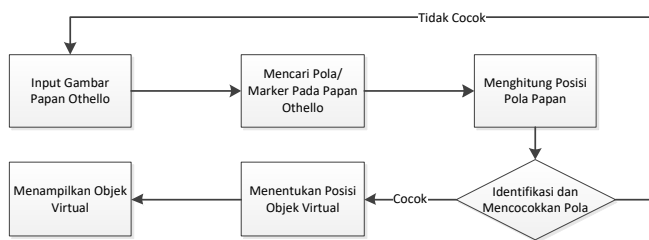
Gambar 7. Contoh Penempatan Pion

E. Pemanfaatan *Augmented Reality*

Augmented Reality merupakan bagian utama dari pembuatan permainan Othello ini. Pemanfaatan *Augmented Reality* pada permainan Othello terletak pada bagaimana sistem mengenali obyek sebagai target gambar dan interaksi pemain dengan sistem yaitu dengan penerapan fitur *Virtual Button* yang memungkinkan sistem dapat menangkap input yang dipilih oleh pemain.

• Deteksi *Image Target*

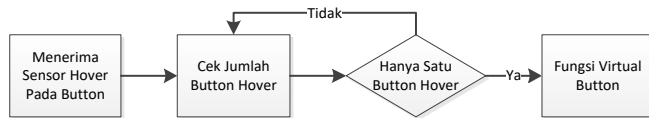
Untuk dapat mendeteksi apakah obyek tersebut merupakan target gambar yang ditentukan atau tidak, diperlukan media perekam yaitu kamera. Kamera akan menangkap obyek yang terpantau oleh lensa. Hasilnya akan dikirim ke komputer untuk diproses apakah obyek yang terpantau merupakan target gambar atau bukan.



Gambar 8. Deteksi *Image Target*

• Cara Kerja *Virtual Button*

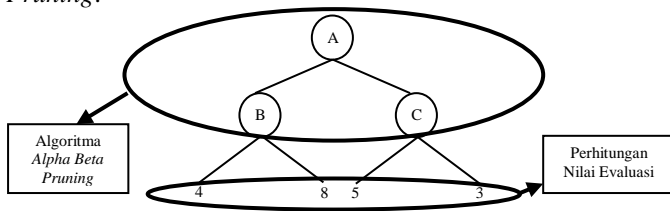
Virtual button merupakan teknik dari *Augmented Reality* dimana pengguna dapat berinteraksi dengan sistem dengan menekan tombol yang telah ditentukan secara *virtual* bukan tombol yang diklik dengan menggunakan perangkat keras.



Gambar 9. Cara Kerja *Virtual Button*

F. Penerapan Algoritma Alpha Beta Pruning

Pada pembuatan permainan Othello digunakan algoritma *Alpha Beta Pruning* sebagai metode pengembangan sistem cerdas untuk sisi komputer. Penunjang informasi dalam pemilihan jalur terbaik digunakan perhitungan nilai evaluasi. Perhitungan nilai evaluasi terletak dibagian paling bawah dari percabangan di area lingkup kerja algoritma *Alpha Beta Pruning*.

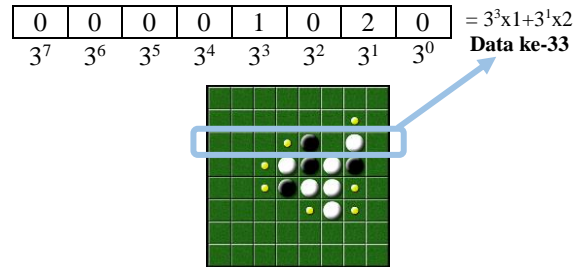


Gambar 10. *Alpha Beta Pruning* dan Perhitungan Nilai Evaluasi

Perhitungan nilai evaluasi pada aplikasi ini menggunakan metode *edge table*, *mobility*, *liberties*, *potential mobility* dan penguasaan *corner*. Berikut penjelasan metode-metode perhitungan nilai evaluasi dijelaskan sebagai berikut.

• *Edge Table*

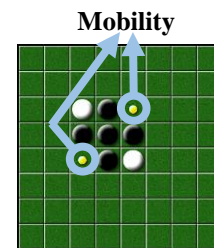
Edge table digunakan untuk membantu nilai evaluasi dengan mengevaluasi sisi papan, dimana hanya perlu mengevaluasi pada satu sisi saja dan kemudian mencerminkan ke semua sisi papan [5]. Berikut contoh penggunaan *edge table* pada suatu kondisi papan.



Gambar 11. Pengecekan Nilai *Mobility*

• *Mobility*

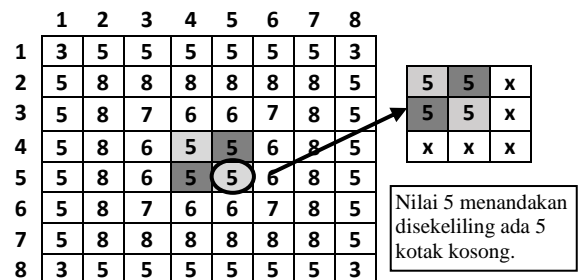
Konsep *mobility* merupakan konsep menjumlahkan langkah yang dapat dimainkan oleh pemain pada satu kali kesempatan. Semakin banyak *mobility* yang dimiliki, maka semakin kecil peluang untuk terpaksa mengambil langkah terburuk. Semakin banyak *mobility* yang dimiliki, juga berarti semakin banyak pula peluang untuk memainkan langkah bagus [5].



Gambar 12. Pengecekan Nilai *Mobility*

• *Liberties*

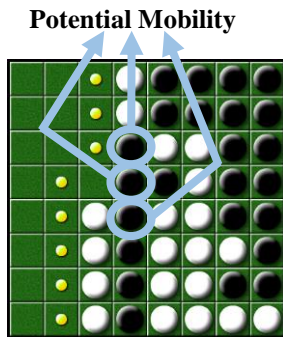
Liberties berfungsi untuk menampung jumlah kotak kosong di sekeliling tiap-tiap kotak pada papan. *Liberties* ini nantinya digunakan untuk memproses nilai *potential mobility* [5].



Gambar 13. Cara Kerja *Virtual Button*

• *Potential Mobility*

Konsep *potential mobility* atau dikenal dengan nama *frontier* merupakan konsep menjumlahkan pion yang berbatasan dengan kotak kosong. Semakin banyak *frontier* semakin jelek pula posisinya, karena semakin banyak *frontier* memungkinkan lawan untuk mendapatkan semakin banyak *mobility* tambahan pada beberapa langkah ke depan dan juga mengurangi *mobility*-nya sendiri [5].



Gambar 14. Pengecekan Nilai *Potential Mobility*

• Penguasaan *Corner*

Penguasaan *corner* (sudut-sudut pojok dari papan) disini tidak kalah penting, karena pion pada posisi tersebut tidak dapat dirubah atau dibalik. Caranya adalah dengan memberikan nilai yang tinggi pada daerah *corner* dan pada daerah-daerah yang berbatasan dengan daerah *corner* tersebut diberikan nilai yang lebih rendah dibandingkan dengan daerah *corner*. Karena daerah-daerah tersebut merupakan daerah rawan untuk mendapatkan daerah *corner* [5]. Berikut pemberian nilai *corner* pada permainan Othello.

75	-3	0	0	0	0	-3	75
-3	-60	0	0	0	0	-60	-3
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
-3	-60	0	0	0	0	-60	-3
75	-3	0	0	0	0	-3	75

Dipengaruhi oleh kondisi didalam 1 baris saja

Dipengaruhi oleh kondisi didalam daerah horizontal, vertikal dan diagonal

Gambar 15. Inisialisasi *Corner*

Metode-metode diatas digunakan untuk perhitungan nilai evaluasi setiap pion, baik pion hitam dan pion putih dengan rumus seperti berikut.

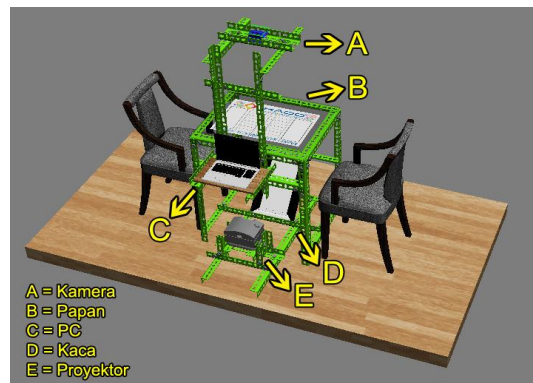
$$\text{Nilai evaluasi pion} = \text{mobility} - \text{potential mobility} + \text{penguasaan corner}$$

Perhitungan diatas digunakan untuk menghitung nilai evaluasi untuk pion hitam dan pion putih pada suatu cabang. Hasil perhitungan nilai evaluasi kedua pion tersebut yang kemudian dihitung kembali dan hasilnya sebagai nilai evaluasi total dengan perhitungan seperti berikut.

$$\text{Nilai evaluasi total} = \text{nilai evaluasi pion putih} - \text{nilai evaluasi pion hitam}$$

G. Implementasi Perangkat Keras Dan Perangkat Lunak

Pentuan jarak antar perangkat keras mempengaruhi kualitas sistem yang dihasilkan. Semakin optimal jarak tiap perangkat keras, semakin baik pula kinerja sistem. Berikut model 3D perancangan *bracket* yang akan digunakan untuk memainkan permainan Othello dengan memanfaatkan *Augmented Reality*.



Gambar 16. Model 3D *Bracket*

H. *User Interface*

Tampilan permainan Othello ini dapat dilihat dari hasil proyeksi proyektor di papan permainan. Berikut dapat dilihat tampilan-tampilan permainan Othello.

a) Menu Utama



Gambar 17. Tampilan Menu Utama

b) Menu Mulai Bermain

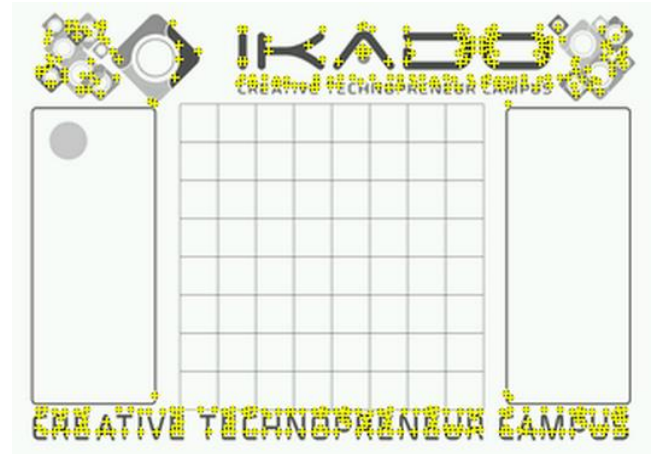


Gambar 18. Tampilan Menu Bermain

c) Menu *Credit*



Gambar 19. Tampilan Menu *Credit*



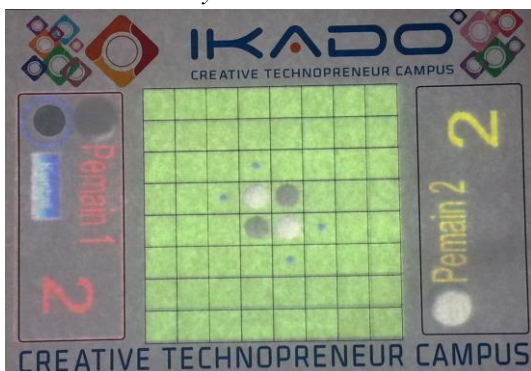
Gambar 22. Fitur *Image Target*

d) Permainan *Single Player*



Gambar 20. Tampilan Permainan *Single Player*

e) Permainan *Multi Player*



Gambar 21. Tampilan Permainan *Multi Player*

I. *Pengujian*

a) *Pengujian Pola Image Target*

Pengujian terhadap beberapa *image target* sangat diperlukan untuk mengetahui bagian penting mana saja yang wajib dideteksi oleh aplikasi melalui kamera sehingga dapat menjalankan aplikasi yang lebih optimal. Pengenalan *image target* berdasarkan fitur (*marker*) yang dikenali oleh aplikasi. Berikut fitur yang dikenali oleh aplikasi dari *image target* yang digunakan.

Dengan menguji beberapa model gambar yang menyerupai *image target* yang asli akan didapat analisa pola gambar yang dikenali aplikasi sebagai *image target*. Berikut hasil uji coba terhadap beberapa *image target* pada tabel berikut.

Tabel 1. Hasil Uji Coba Pola *Image Target*

Pola Gambar	Hasil
	Semua fungsi berjalan dengan baik
	Semua fungsi berjalan dengan baik. Tulisan tidak terlihat.
	<i>Image target</i> kurang stabil.
	<i>Image target</i> tidak stabil.

Pola Gambar	Hasil
	Semua fungsi berjalan dengan baik.
	Semua fungsi berjalan dengan baik.
	Semua <i>virtual button</i> tidak dapat berfungsi dengan baik.
	<i>Virtual button</i> kembali di label pemain pertama tidak berfungsi.
	<i>Image target</i> dikenali namun tidak dapat merepresentasikan obyek dengan baik. Respon <i>virtual button</i> buruk.

b) Pengujian Kualitas Kamera

Pengujian terhadap kamera yang berbeda dapat menghasilkan informasi yang berguna dalam pemilihan kamera yang digunakan pada permainan Othello yang memanfaatkan teknologi *Augmented Reality*. Pengujian kualitas kamera akan menguji 3 jenis kamera yang berbeda dengan spesifikasi kamera yang dapat dilihat pada tabel berikut.

Tabel 2. Spesifikasi Kamera

Jenis Kamera	Resolusi Kamera	Kualitas Video	Auto Focus
Lenovo Vibe Z K910 Secondary Camera	5.0 MP	480 pixel	X
Logitech HD Webcam C525	8.0 MP	720 pixel	✓
Asus Zenfone 6 A600CG Primary Camera	13.0 MP	1080 pixel	✓

c) Pengujian Jarak Kamera

Jarak kamera terhadap *image target* sangat mempengaruhi stabilitas pengenalan *image target* dan *virtual button*. Semakin dekat jarak kamera semakin sulit kamera mengenali *image target*, sebaliknya semakin jauh jarak kamera maka semakin sulit pula kamera mendeteksi penekanan pada *virtual button*. Maka dari hasil analisa tersebut, pencarian jarak optimal sangat diperlukan guna menjalankan permainan Othello berbasis *Augmented Reality* dengan baik. Dengan menggunakan papan permainan berukuran A4, berikut hasil uji coba jarak kamera terhadap *image target* dari ketiga kamera yang digunakan pada tabel berikut.

Tabel 3. Hasil Uji Coba Jarak Kamera

Jarak Kamera Terhadap Papan	Lenovo Vibe Z 5.0 MP		Logitech C525 8.0 MP		Asus Zenfone 6 13.0 MP	
	IT	VB	IT	VB	IT	VB
17 cm	✓	X	✓	X	✓	X
45 cm	✓	✓	✓	X	✓	✓
55 cm	✓	X	✓	✓	✓	✓
59 cm	✓	X	✓	✓	✓	X
66 cm	X	X	✓	✓	✓	X
68 cm	X	X	✓	✓	✓	X
71 cm	X	X	✓	✓	✓	X
77 cm	X	X	✓	✓	X	X
79 cm	X	X	✓	X	X	X
95 cm	X	X	X	X	X	X
108 cm	X	X	X	X	X	X

d) Pengujian Cahaya Pada Kamera

Pengujian cahaya yang ditangkap kamera bertujuan untuk mengetahui seberapa besar cahaya yang ditangkap oleh kamera untuk dapat mendeteksi *image target* dan *virtual button* dengan baik. Satuan tingkat cahaya yang digunakan yaitu *lux* (lx). *Lux* adalah satuan energi cahaya yang diterima oleh luas permukaan. Pengujian dilakukan dengan menempelkan pendeteksi nilai *lux* terhadap kamera yang digunakan. Uji coba dilakukan dengan memberi serangkaian pencahayaan yang berbeda dengan melakukan pengecekan terhadap nilai *lux* yang diperoleh. Berikut hasil uji coba tingkat cahaya dari ketiga kamera yang digunakan.

Tabel 4. Hasil Uji Coba Tingkat Cahaya Kamera

Tingkat Cahaya	Lenovo Vibe Z 5.0 MP		Logitech C525 8.0 MP		Asus Zenfone 6 13.0 MP	
	IT	VB	IT	VB	IT	VB
30-33 lx	✓	X	✓	X	✓	X
37-40 lx	✓	X	✓	X	✓	X

Tingkat Cahaya	Lenovo Vibe Z 5.0 MP		Logitech C525 8.0 MP		Asus Zenfone 6 13.0 MP	
	IT	VB	IT	VB	IT	VB
	50-54 lx	✓	✗	✓	✗	✓
64-68 lx	✓	✓	✓	✓	✓	✓
88-91 lx	✓	✓	✓	✓	✓	✓
105-108 lx	✓	✓	✓	✓	✓	✓
112-118 lx	✓	✓	✓	✓	✓	✓

e) Pengujian Sensitivitas *Virtual Button*

Pengujian terhadap sensitivitas *virtual button* dilakukan untuk mengetahui waktu respon yang diproses oleh aplikasi. Semakin kecil waktu respon yang diproses oleh aplikasi, maka semakin responsif aplikasi menerima inputan (berupa *hover*). Sebaliknya jika semakin kecil waktu respon yang diproses oleh aplikasi, maka kemampuan aplikasi memproses obyek yang terdeteksi oleh kamera semakin lambat. Pengaturan sensitivitas *virtual button* dilakukan pada *software* Unity yang dapat dilihat pada gambar berikut.

Tabel 5. Hasil Uji Coba Sensitivitas *Virtual Button*

Sensitivitas	Waktu Respon
High	0.142 detik
Medium	0.217 detik
Low	0.303 detik

f) Pengujian AI Komputer Terhadap Aplikasi Othello Serupa

Pengujian terhadap aplikasi Othello serupa yaitu dengan menandingkan komputer permainan Othello ini dengan komputer aplikasi penanding dimana metode AI yang digunakan pada aplikasi penanding tidak diketahui. Komputer aplikasi penanding akan memainkan pion hitam akan ditandingkan dengan komputer permainan Othello ini sebagai pion putih. Berikut hasil uji coba AI komputer terhadap aplikasi Othello serupa pada tabel berikut.

Tabel 6. Hasil Uji AI Komputer Terhadap Aplikasi Serupa

Nama Aplikasi Dan Keterangan	Uji Coba Ke-	Pion Aplikasi Serupa	Pion Komputer AR
Reversi Free v1.303 Level: Tidak Ada (Android)	1	16	48
	2	4	60
	3	4	59
Ultima Reversi v1.5.1 Level: Auto Matching (Android)	1	1	39
	2	39	25
	3	41	23

Nama Aplikasi Dan Keterangan	Uji Coba Ke-	Pion Aplikasi Serupa	Pion Komputer AR
KingOfReversi v150210 Level: 4 dari 6 (Android)	1	18	46
	2	30	34
	3	15	49
http://www.best-flash-games.net/puzzle-games/reversi.html Level: Tidak Ada	1	7	57
	2	28	36
	3	17	47
http://reversi.watype.net/ Level: 3 dari 4	1	37	27
	2	35	29
	3	15	49

g) Pengujian AI Komputer Terhadap *User*

Selain pengujian dilakukan dengan menandingkan permainan Othello ini dengan aplikasi serupa, pengujian AI komputer juga dilakukan terhadap pemain yaitu dengan mengajak *user* untuk melawan komputer dari aplikasi permainan Othello ini. *User* akan memainkan pion hitam dan komputer memainkan pion putih. Berikut hasil uji coba AI komputer terhadap *user* pada tabel berikut.

Tabel 7. Hasil Uji AI Komputer Terhadap *User*

Nama <i>User</i>	Uji Coba Ke-	Pion Pemain	Pion Komputer Othello AR
Jemmy	1	3	61
	2	21	43
Hindohan	1	3	60
	2	18	46
Hendri	1	22	42
	2	11	53
Fransisca	1	11	53
	2	31	33
Dandy	1	26	38
	2	43	21
Michael	1	49	15
	2	17	47
Joshua	1	28	36
	2	18	46
Adriel	1	1	58
	2	5	58
Gladys	1	33	31
	2	17	47
Claudia	1	33	31
	2	0	47
Lifan	1	22	42
	2	19	45
Reza	1	22	42
	2	16	48
Veronica	1	29	35
	2	20	44

Nama User	Uji Coba Ke-	Pion Pemain	Pion Komputer Othello AR
Ndaru	1	16	48
	1	23	41
Sathya	2	38	26
	3	30	34
Fergy	1	15	49
	2	15	49
Christhofer	1	50	14
	2	35	29
Guruh	1	0	42
	2	52	12
	3	1	58

IV. KESIMPULAN

Akan dijelaskan beberapa kesimpulan yang dapat diambil dari pembuatan aplikasi permainan Othello yang memanfaatkan teknologi *Augmented Reality*. Kesimpulan yang dapat diambil antara lain:

- Jika dibandingkan dengan permainan Othello konvensional, permainan Othello ini membutuhkan perangkat yang lebih banyak dan persiapan lebih lama namun permainan Othello ini lebih mempermudah *user* dalam memainkan permainan yaitu lebih praktis dalam membalikkan pion musuh (*reversi*) dan juga dapat memainkan mode melawan komputer. Sedangkan jika dibandingkan dengan permainan Othello berbasis aplikasi, permainan Othello ini tidak dapat dibawa kemana-mana dan memiliki tampilan yang kalah menarik dibandingkan versi aplikasi namun *user* dapat berinteraksi dengan *user* lain secara langsung.
- Agar pemanfaatan *Augmented Reality* pada permainan Othello dapat bekerja secara optimal, menggunakan pola papan permainan yang sesuai dengan fitur *image target*

yang digunakan dengan ukuran rasio papan yang sesuai yaitu ukuran A4 (21 cm x 29,7 cm), penggunaan jenis kamera yang baik yaitu *webcam* Logitech C525, jarak rata-rata kamera terhadap papan permainan yaitu 66 cm, tingkat cahaya rata-rata yang ditangkap kamera sekitar 91 lx serta menggunakan pengaturan sensitivitas *medium* pada *virtual button*.

- Dari 52 pengujian terhadap kemampuan AI komputer, menghasilkan persentase kemenangan mencapai 73,33% dari 15 pengujian dengan jumlah kemenangan 11 kali dan 78,34% dari 37 pengujian dengan jumlah kemenangan 29 kali.

REFERENSI

- [1] Hendrianto, D. (2010). *Implementasi Augmented Reality Memanfaatkan Sensor Akselerometer, Kompas dan GPS Pada Penentuan Lokasi Masjid Berbasis Android*, Surabaya: Institut Teknologi Sepuluh Noverber.
- [2] Heineman, G. T., Pollice, G., dan Selkow, S. (2009). *Algorithms In A Nutshell*, California: O'Reilly Media.
- [3] Azhar, N. F. (2011). *Pemanfaatan Augmented Reality Untuk Game Ranger Target FPS Berbasis Android Menggunakan Unity 3D Dan Vuforia SDK*. Malang: Universitas Muhammadiyah Malang.
- [4] Suhendar, A. (2002). *Visual Modelling Menggunakan UML Dan Rational Rose*, Bandung: Informatika Bandung.
- [5] Gunawan, Kristian, Y., Andika, H. (2009). *Game Playing Untuk Othello Dengan Menggunakan Algoritma Negascout Dan MDTF*. Surabaya: Institut Sains Terapan Dan Teknologi Surabaya.