



## Profiling Gaya Sutradara Berdasarkan Penempatan Posisi Kamera dengan Fuzzy Logic

Hartarto Junaedi<sup>#1</sup>, Mochamad Hariadi<sup>\*2</sup>, I Ketut Eddy Purnama<sup>\*3</sup>

<sup>#</sup>Departemen Teknik Elektro Fakultas Teknologi Elektro Institut Sepuluh Nopember Surabaya  
Jl. Teknik Mesin Gedung B,C, dan AJ. Kampus ITS, Sukolilo Surabaya 60111

<sup>1</sup>hartarto12@mhs.ee.its.ac.id

<sup>\*</sup>Departemen Teknik Komputer Fakultas Teknologi Elektro Institut Sepuluh Nopember Surabaya  
Jl. Teknik Mesin Gedung B dan C, Kampus ITS, Sukolilo Surabaya 60111

<sup>2</sup>mochar@ee.its.ac.id

<sup>3</sup>ketut@ee.its.ac.id

**Abstrak**—Teknologi komputer telah banyak digunakan di tidak hanya untuk tujuan penelitian dan pendidikan saja tetapi juga di dunia hiburan. Salah satu hiburan berbasis komputer adalah permainan komputer dan animasi. Salah satu komponen pendukungnya adalah *machinima*. *Machima* adalah teknologi yang akan menempatkan sebuah komponen sinematik dalam dunia *virtual*. Salah satu komponen yang dapat dikontrol adalah penempatan posisi kamera. Seorang sutradara bisa dibedakan gaya penyutradaraannya, salah satunya berdasarkan penempatan posisi kamera. Dengan menerapkan suatu gaya penyutradaraan pada sebuah permainan atau animasi bisa mendapatkan suasana yang berbeda. Penelitian ini akan mencoba melakukan profiling terhadap gaya seorang sutradara berdasarkan kebiasaan penempatan posisi kamera. Pendekatan yang dilakukan berbasis logika fuzzy. Penelitian ini akan menggunakan 19 variabel input yang berasal dari hasil ekstraksi data simulasi dan 5 variabel output untuk melakukan profiling terhadap dua buah gaya sutradara yang berbeda dengan pendekatan logika fuzzy. Akan dihasilkan diagram area dan histogram sehingga mempermudah pembacaan dalam membedakan gaya sutradara dan berhasil dibedakan berdasarkan modus hasil analisa terhadap diagram histogram.

**Kata kunci**— *Machinima*, *Fuzzy Logic*, *Gaya Sutradara*, *Profiling*

### I. PENDAHULUAN

*Machima* adalah sebuah teknik untuk membuat permainan komputer atau animasi dalam dunia *virtual* secara real time. Dalam dunia virtual kita dapat menciptakan lingkungan dan semua karakter yang diinginkan dan menganimasi semua aksi yang dibutuhkan. Saat ini pengembang aplikasi 3D dituntut untuk membuat animasi yang senatural mungkin sesuai dengan dunia nyata. Untuk mendapatkan hal ini maka para pengembang aplikasi 3D mengembangkan berbagai macam algoritma dan metode untuk menerapkan aturan sinematografi di dalam dunia virtual. *Machinima*

menggunakan teknologi grafis untuk merender 3D secara *real time*[1]. Dengan teknologi inilah sebuah produk sinematik akan dihasilkan. *Machinima* adalah sebuah alternative untuk menghasilkan sebuah animasi dengan biaya yang lebih murah [2], tetapi untuk menghasilkan produk sinematik yang bagus tentu saja dibutuhkan penelitian yang berkaitan dengan kontrol terhadap kamera *virtual*. Dalam dunia nyata seorang sutradara sering menggunakan sebuah storyboard untuk visualisasi ide mereka supaya dapat dipahami oleh animator atau juru kamera[3].



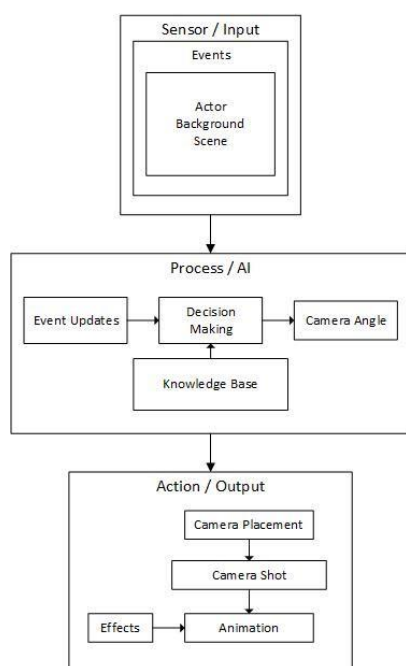
(a) (b)

Gambar. 1 Point of View

Walaupun saat ini telah banyak penelitian di area *machinima* termasuk penempatan posisi kamera, masih sedikit sekali penelitian yang berhubungan dengan gaya sutradara. Penelitian ini akan fokus terhadap profiling sebuah produk animasi apakah sesuai dengan gaya seorang sutradara. Sistem akan memeriksa apakah penempatan posisi kamera sudah sesuai dengan perilaku seorang sutradara dalam menempatkan posisi kameranya secara otomatis. Harapan ke depan setelah berhasil dilakukan profiling adalah melakukan reverse sehingga dapat ditemukan aturan yang dapat diterapkan dalam engine 3D sehingga gaya tersebut dapat diterapkan pada sebuah permainan atau animasi. Mungkin dapat dibayangkan permainan yang terkenal Mario Bros [4][5], permainan *side-scroll* sederhana kita pasang kamera engine yang berbeda misalkan kamera engine permainan

RPG semacam *Lufia* atau action semacam *Assasin Creed* [6][7] maka akan dihasilkan sebuah nuansa permainan yang benar-benar berbeda. Pada gambar 1 dapat dilihat sebuah adegan sama yang diambil dengan posisi kamera berbeda.

Banyak sekali penelitian di bidang animasi dan permainan computer. Perspektif permainan sekarang bergeser dari teknologi 2D ke teknologi 3D, dimana penelitian sekarang menunjukkan bahwa teknologi dunia virtual ini semakin mendekati kondisi sebenarnya di dunia nyata. Pada gambar 2 tampak blok sistem penempatan posisi kamera *virtual* secara umum. Disana dibutuhkan beberapa metode dan algoritma sehingga penempatan posisi kamera akan sesuai.



Gambar. 2 Blok Sistem Penempatan Posisi Kamera Secara Umum

Biasanya dalam produksi game atau animasi, penempatan kamera dilakukan oleh seorang animator atau direktur fotografi. Namun dalam lingkungan *virtual* penempatan posisi kamera ini membutuhkan banyak pemodelan dan perhitungan yang harus dilakukan berulang-ulang [8] tentu saja ini akan memakan biaya dan waktu. Penelitian [9] mengusulkan sebuah dasar aturan untuk kontrol kamera dalam komputer grafis. Tentu saja aturan ini sesuai dengan kaidah sinematografi, seperti pada penelitian [10] menggunakan metode PSO multi objektif untuk menerapkan aturan fotografi seperti *rule of third*, garis horizon dan POI (Point of Interest).

Beberapa penelitian yang berkaitan dengan penempatan posisi kamera antara lain penelitian [11] dimana digunakan PSO untuk menyelesaikan permasalahan *Virtual Camera Composition* (VCC). Pendekatan yang digunakan adalah hybrid dimana pertama dihitung posisi kamera berdasarkan batasan-batasan yang telah didefinisikan kemudian akan dihitung ulang dengan menggunakan PSO. Output dari sistem ini adalah posisi

kamera, orientasi dan *Field of View* (FOV). Penelitian [12] mengusulkan penempatan posisi virtual kamera berdasarkan batasan yang dibuat. Penelitian [13] mengusulkan penempatan multi kamera, tetapi kamera pertama adalah kamera statis posisinya jadi hanya kamera kedua yang dihitung penempatannya berdasarkan metode *static behaviour tree*. Penelitian [14] mengusulkan pendekatan *behaviour tree* ditambah dengan metode A\* untuk melakukan *follow* pada aktor utama. Paper [15][16] membuat sebuah bahasa semi otomatis untuk melakukan kontrol terhadap virtual kamera sedangkan paper [17] memperbaiki penempatan posisi kamera untuk *first person navigation view* berdasarkan beberapa parameter tambahan seperti tinggi dan berat badan.

Paper [18] menggunakan metode *Support Vector Machine* (SVM) dengan menganalisa empat karakteristik dalam sebuah shot, sedangkan paper [19] mengusulkan sebuah sistem yang disebut dengan *Intelligent Virtual Environment* (IVE) untuk melakukan simulasi agen *virtual* dalam sebuah lingkungan *virtual*. Empat modul yang dikembangkan yaitu framework IC, IVE, AgentSim dan Visualizer. Penelitian [20] juga mengusulkan multi agen cerdas dimana behaviornya berdasarkan metode PSO. Pendekatan ini juga dapat digunakan pada virtual kamera. Penelitian [21][22] menciptakan sebuah sistem yang disebut dengan *Darshak*. Sistem ini akan secara otomatis membangun wacana naratif dari cerita yang diinputkan dalam sistem. Ada Sembilan operator yang digunakan dalam kamera virtual ini. Hasilnya akan divisualisasikan dalam 3D game engine. Penelitian [23] mengusulkan sebuah direktur sinematografi cerdas untuk kontrol kamera berdasarkan *storytelling*. Peranan utama dari direktur ini adalah memilih scene pengambilan gambar yang sesuai secara *real time*. Metode yang digunakan adalah SVM.

Walaupun cukup banyak penelitian yang berkaitan dengan penempatan posisi kamera tetapi tidak banyak penelitian yang bicara khusus mengenai gaya sutradara dalam penempatan posisi kamera termasuk cara mengukur kemiripan sebuah animasi dengan gaya tersebut. Penelitian ini bukan bicara tentang bagaimana cara penempatan posisi kamera berdasarkan sebuah gaya tetapi cara melakukan profile animasi berdasarkan gaya tersebut. Banyak penelitian sejenis yang pengukurannya berdasarkan kuesioner, tetapi penelitian ini diharapkan dalam melakukannya secara otomatis.

Makalah ini disusun dalam lima bagian sebagai berikut. Bagian pertama akan menyajikan pendahuluan mengenai *state of the art* penelitian dan penelitian sebelumnya yang berkaitan. Bagian kedua akan membahas mengenai aturan sinematografi dan gaya sutradara. Metode yang diusulkan dalam penelitian ini akan dibahas pada bagian ketiga. Bagian keempat akan membahas hasil percobaan penelitian ini dan akan ditarik sebuah kesimpulan dan usulan penelitian lanjut untuk masa depan pada bagian kelima.

## II. SINEMATOGRAFI DAN GAYA SUTRADARA

Sebuah gambar bergerak terdiri dari beberapa pengambilan gambar. Setiap pengambilan gambar tersebut membutuhkan penempatan posisi kamera terbaik sesuai dengan aturan sinematografi. Sinematografi [24] berarti pencahayaan dan penempatan kamera. Salah satu produk sinematik adalah film. Film [25] sendiri adalah kumpulan dari bahasa dan estetika, estetika dalam sinematografi adalah penempatan kamera dan pengaturan semua properti untuk mendukung jalan sebuah cerita.

Beberapa faktor [26] yang harus dipertimbangkan untuk menghasilkan produk sinematik yang bagus yaitu :

- *Camera Angle*  
Menunjukkan posisi kamera ditempatkan pada saat pengambilan gambar. Sebuah scene bisa diambil secara bersamaan dari berbagai macam sudut. Beberapa macam *camera angle* antara lain *objective shot*, *subjective shot* dan *point of view shot*. Shot [27] ini juga dapat dikategorisasikan menjadi *close up shot*, *medium shot* dan *long shot*.
- *Continuity*  
Adalah sebuah kondisi pergantian dari sebuah frame ke frame berikut. Tanpa adanya continuity [28] frame tidak akan sinambung.
- *Cutting*  
Adalah proses perubahan point of view [29]. Tanpa adanya proses cutting yang baik maka plot film akan terganggu.
- *Close Up*  
Adalah Teknik pengambilan gambar dari jarak yang dekat. Jadi kamera dan objek yang diambil gambarnya dekat sekali.
- *Composition*  
Adalah proses penyusunan berbagai macam elemen gambar menjadi satu kesatuan yang harmoni. Komposisi adalah bagaimana seorang sutradara mengarahkan pemain, menempatkan background dan properti semuanya menjadi satu elemen utuh.

Setiap sutradara memiliki gaya nya sendiri yang unik dalam mengarahkan dan mengambil gambar sebuah scene. Gaya inilah yang membedakan seorang sutradara dengan sutradara lainnya. Salah satu sutradara yang terkenal adalah *James Cameron* yang mengarahkan film box office *Avatar*. Film ini [30] dapat dikatakan sebagai tonggak sejarah kelahiran film dengan menggunakan lingkungan *virtual*. Bahkan dalam pembuatan film ini sang sutradara membuat teknologi kamera *virtual* yang memiliki fungsi layaknya kamera normal tetapi hanya dapat digunakan dalam lingkungan *virtual*. *James Cameron* amat terkenal dengan gaya pengambilan gambar nya yang detail. Dia suka menekankan pada background dan detail komponen. Seperti pada film *Titanic* penonton diajak melihat detail kemegahan dari kapal *Titanic*. *Christopher Nolan* seorang sutradara terkenal lainnya suka menekankan elemen realistic pada film-filmnya seperti pada film *Dark Knight* dan *Man of Steel*.



Gambar. 3 Beberapa Contoh Pengambilan Gambar Quentin Tarantino (a) Trunk and Hood POV (b) The Corpse POV (c) God's Eye POV (d) Black and White Shot (e) Lip Shot (f) Violent Awakening

Sutradara terkenal lainnya yaitu *Quentin Tarantino* [31][32][33] dengan beberapa film box officenya antara lain *Kill Bill*, *Pulp Fiction*, *From Dusk Till Dawn* dan banyak lainnya. *Quentin Tarantino* diakui sebagai salah satu sineas sukses dan jago dalam mengekspresikan idenya. Setiap pengemar film pasti tahu style dari *Quentin Tarantino* ini. Hampir semua filmnya menggunakan gaya *action-thriller*, dengan menggunakan kegelapan dan elemen sadis. Gambar 3 menunjukkan beberapa style *Quentin Tarantino* dalam pengambilan gambar.

Beberapa gaya pengambilan gambar (point of view) dari *Quentin Tarantino* yang digunakan dalam filmnya :

- *The Trunk and hood POV*  
Merupakan teknik pengambilan gambar seolah-olah dari bagasi mobil. Banyak sekali film *Quentin* yang menggunakan gaya ini.
- *Corpse POV*  
Merupakan teknik pengambilan gambar seolah-olah diambil dari sudut mata korban atau mayat. Teknik ini variasi dari *low angle shot*.
- *Tracking Shot*  
Merupakan gaya pengambilan yang mengikuti pemain utama. Pengambilan gambar seolah-olah diambil dari sudut mata seorang *stalker* yang sedang mengikuti seseorang.
- *God's Eye Shot*  
Merupakan gaya pengambilan dari atas kepala pemain sehingga penonton tampak apa yang dikerjakan oleh pemain. Ini merupakan variasi teknik *high angle shot*.
- *Black and White Shot*  
Merupakan gaya pengambilan dalam suasana hitam putih. Teknik ini biasanya digunakan untuk penekanan sesuatu atau dalam kondisi *flash back*.
- *Close up on Lip*



Merupakan gaya pengambilan gambar *close up* dimana pengambilan gambar di sekitar area bibir. Teknik ini memberikan efek misterius terhadap sebuah scene.

- *Violent Awakening*

Merupakan gaya pengambilan dari sudut pandang seseorang yang tiba-tiba terbangun. Teknik ini menunjukkan kejutan yang terjadi.

Selain beberapa gaya diatas Quentin Tarantino juga suka menambahkan efek semacam semburan darah, penggunaan mobil dll. Tetapi untuk penelitian ini hanya akan digunakan lima gaya saja yang berhubungan dengan penempatan posisi kamera sedangkan gaya *black and white* tidak digunakan karena tidak berhubungan dengan penempatan posisi kamera.

### III. METODE PROFILING

#### A. Desain Simulasi

Dalam simulasi ini ada beberapa input yang akan diproses oleh sistem. Dalam simulasi ini akan digunakan 3 *scene* yang berbeda dan untuk masing-masing *scene* terdapat dua gaya sutradara yang berbeda. Style pertama merupakan adopsi dari gaya Quentin Tarantino sedangkan style kedua merupakan adopsi aturan sinematografi secara umum. Setiap *scene* akan memiliki komponen sebuah kamera bergerak, aktor utama, aktor pendukung jika dibutuhkan. Jadi untuk setiap *scene* bisa memiliki 3 set koordinat dan *timestamp* yang menunjukkan urutan frame.



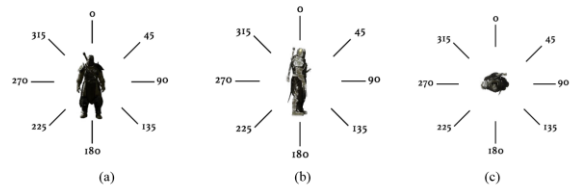
Gambar. 4 Karakter untuk simulasi

Tampak pada gambar 4 adalah karakter untuk simulasi. Karakter utama yang terletak di sebelah kiri. Kompleksitas simulasi yang berkaitan dengan karakter dan background dapat dilihat pada tabel 1.

TABLE I  
KOMPLEKSITAS KARAKTER DAN SCENE

Character / Scene	Objects	Triangle	Vertices
Background	720	629K	451K
Main Character	1	35K	21K
Second Character	1	8970	6154

Ada 19 input yang diekstraksi dari simulasi. Karena kita menggunakan engine 3D maka setiap karakter akan memiliki 3 koordinat  $x, y$  dan  $z$  dan juga ada sudut berdasarkan axis yaitu  $rx, ry$  dan  $rz$ . Tampak pada gambar 5 adalah sudut angle berdasarkan axis. Sedangkan area untuk simulasi dapat dilihat pada gambar 6.

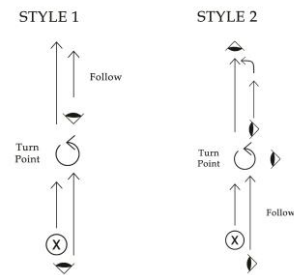


Gambar. 5 Shot Direction Angle (a) Z-Axis (b) X-Axis (c) Y-Axis



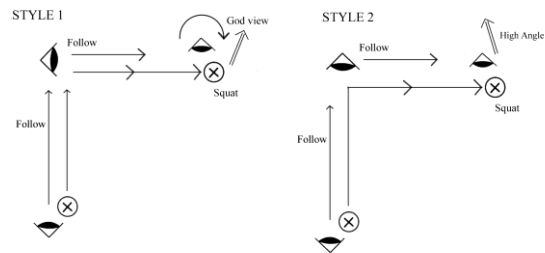
Gambar. 6 Desain Area untuk Simulasi (a) Top View (b) Perspektif View

Untuk membuat animasi pendek yang hasilnya akan diperiksa menggunakan sistem ini, kami menggunakan *storyboard* untuk menggambarkan pergerakan kamera sehingga animator dapat membuat animasinya. Style 1 merupakan adopsi dari gaya Quentin dan style 2 merupakan adopsi gaya generik.



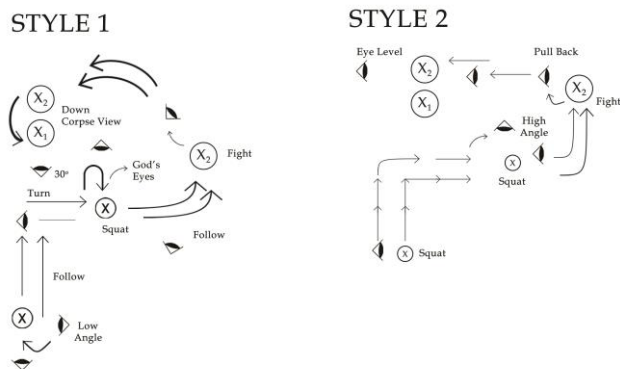
Gambar. 7 Story Board untuk scene 1

Tampak pada gambar 7 adalah storyboard untuk scene 1. Disana terlihat 2 style yang berbeda walaupun aksi yang dilakukan adalah sama yaitu karakter pertama berjalan dan ditengah perjalanan akan berhenti untuk berputar karena merasa diikuti dan kemudian akan berjalan kembali meneruskan perjalanan. Scene 1 ini merupakan scene yang paling sederhana dimana animasi sekitar 24 detik.



Gambar. 8 Story Board Untuk Scene 2

Gambar 8 merupakan story board untuk scene 3. Pada scene ini karakter utama tampak berbelok ke kanan dan pada akhir scene karakter akan berjongkok untuk mengambil sesuatu. Animasi untuk scene 3 ini kurang lebih 27 detik.



Gambar. 9 Story Board Untuk Scene 3

Gambar 9 merupakan story board untuk scene 3. Dimana scene ini paling kompleks dibandingkan dengan scene yang lain. Tampak di story board pada awal scene karakter berjongkok untuk mengambil sesuatu kemudian berjalan menuju ke sebuah ruangan untuk mengambil sebuah barang. Dan pada akhirnya menuju ke sebuah ruangan dimana pada ruangan tersebut ada karakter pendukung dan karakter utama akan berkelahi dengan karakter pendukung dan pada akhirnya karakter utama akan kalah berkelahi dan pingsan. Scene terakhir ini animasinya memakan waktu sekitar 1 menit lebih.

**B. Logika Fuzzy untuk profiling**

Pada bagian ini akan dibahas mengenai pendekatan fuzzy yang digunakan. Untuk penelitian ini akan dibuat 5 jenis fuzzy output untuk 5 aturan sinematografi yaitu :

- *Tracking / Following Shot*  
Digunakan untuk menentukan apakah penempatan posisi kamera merupakan tracking shot. Variabel yang digunakan diberi nama Follow Shot.
- *Close Up Shot*  
Digunakan untuk menentukan jenis close up shot apakah termasuk choker shot atau sekadar close up shot biasa atau bukan close up shot. Variabel yang digunakan diberi nama Lip Shot.
- *High Angle Shot*  
Digunakan untuk menentukan jenis high angle shot. Apakah termasuk god view shot atau sekadar high shot atau bukan high shot. Variabel yang digunakan diberi nama God View.
- *Low Angle Shot*  
Digunakan untuk menentukan apakah termasuk low angle shot dari first person view. Variabelnya adalah Low First Player.
- *Trunk Shot*

Digunakan untuk menentukan apakah pengambilan dari bagasi atau low angle. Variabel yang digunakan diberi nama Trunk Player.

TABLE II  
FUZZY OUTPUT MEMBERSHIP

Output Variable	Membership Function		
	MF	Type	Control
Follow Shot (O1)	Unfollow	trapmf	[0,0,2,3]
	Pseudo	trimf	[2,4,6]
	Follow	trapmf	[5,7,10,10]
Lip Shot (O2)	Unlip Shot	trapmf	[0,0,2,4]
	Pseudo	trimf	[2,5,8]
	Lip Shot	trapmf	[6,8,10,10]
God View (O3)	Not High Angle	trapmf	[0,0,2,4]
	High Angle	trimf	[2,5,8]
	God View	trapmf	[6,8,10,10]
Low First Player (O4)	Unlow	trapmf	[0,0,2,4]
	Middle Low	trimf	[2,5,8]
	High Low	trapmf	[6,8,10,10]
Trunk Player (O5)	Untrunk	trapmf	[0,0,2,4]
	Semi	trimf	[2,5,8]
	Trunk	trapmf	[6,8,10,10]

Tabel 2 menunjukkan fuzzy output membership yang digunakan dalam simulasi dalam penelitian ini. Terdapat 5 output variabel yang merupakan aturan dalam sinematografi dan masing-masing variabel memiliki 3 membership function. Trimf menunjukkan membership function typenya adalah triangle. Control yang dibutuhkan adalah 3 buah titik. Sedangkan tramf atau trapesium yang membutuhkan 4 buah titik.

Dari 34 input sistem yang digunakan dalam sistem ini adalah 8 input tetapi pada penelitian selanjutnya akan kami optimasi semua input yang ada. Berikut adalah membership untuk input :

- *Distance\_P1*  
Jarak antara kamera dengan karakter utama.
- *Different\_P1*  
Perbedaan jarak antara sebuah frame dengan frame berikutnya.
- *Angle\_Y\_P1*  
Sudut y antara karakter utama dan kamera.
- *Distance\_P2*  
Jarak antara kamera dan karakter pendukung
- *Angle\_Y\_P2*  
Sudut y antara kamera dan karakter pendukung.
- *Angle\_X\_P1*  
Sudut x antara kamera dan karakter utama.
- *Coordinat\_Y*  
Ketinggian kamera berdasarkan sudut y kamera.
- *Angle\_X\_P2*  
Sudut x antara kamera dan karakter pendukung.

TABLE III  
FUZZY INPUT MEMBERSHIP

Input Variable	Membership Function		
	MF	Type	Control
Distance_P1 (I1)	Near	trapmf	[0,0,1,7,2]
	Medium	trimf	[1,7,2,1,2,5]
	Far	trapmf	[2,3,3,20,20]
Different_P1 (I2)	Short	trapmf	[0,0,20,40]
	Medium	trimf	[20,50,80]
	Long	trapmf	[60,80,100,100]
Angle_Y_P1 (I3)	Front Left	trapmf	[-180,-180,-160,-110]
	Left	trimf	[-160,-90,-20]
	Rear	trimf	[-70,0,70]
	Right	trimf	[20,90,160]
	Front Right	trapmf	[110,160,180,180]
Distance_P2 (I4)	Near	trapmf	[0,0,20,40]
	Medium	trimf	[20,50,80]
	Far	trapmf	[60,80,100,100]
Angle_Y_P2 (I5)	Front Left	trapmf	[-180,-180,-160,-110]
	Left	trimf	[-160,-90,-20]
	Rear	trimf	[-70,0,70]
	Right	trimf	[20,90,160]
	Front Right	trapmf	[110,160,180,180]
Angle_X_P1 (I6)	Rear Upper	trapmf	[-180,-180,-160,-110]
	Upper	trimf	[-160,-90,-20]
	Front	trimf	[-70,0,70]
	Below	trimf	[20,90,160]
	Rear Below	trapmf	[110,160,180,180]
Coordinat_Y (I7)	Low	trapmf	[0,0,20,40]
	Eye View	trimf	[20,50,80]
	High	trapmf	[60,80,100,100]
Angle_X_P2 (I8)	Front Upper	trapmf	[0,0,45,110]
	Rear Upper	trimf	[70,135,200]
	Rear Below	trimf	[160,225,290]
	Front Below	trapmf	[250,315,360,360]

Tabel 3 menunjukkan input membership function yang digunakan berupa 8 input. Rule Fuzzy yang digunakan merupakan kombinasi dari input fuzzy dimana didapatkan sekitar 40k rule yang kemudian direduksi menjadi 47 rule. Beberapa rule yang digunakan dapat dilihat pada tabel 4. Tabel 5 menunjukkan representasi rule pada tabel 4 menjadi bentuk rule IF THEN.

Dari tabel 5, rule No. 1 dapat dijelaskan sebagai berikut : IF *distance\_p1=medium* AND *different\_p1 = short* AND *angle\_Y\_P1 = rear* THEN *follow\_shot =follow*. Aturan ini artinya bilamana jarak antara kamera *virtual* dan player adalah medium, perbedaan antara posisi sekarang dan sebelumnya pendek, dan sudut y terletak di belakang maka jenis shot ini disebut *Follow Shot*.

TABLE IV  
REDUCED FUZZY RULE SAMPLE

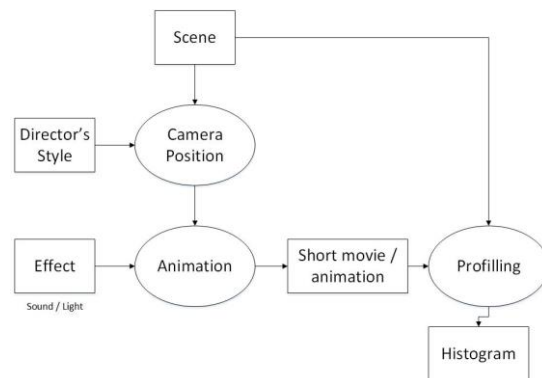
	Input Fuzzy						Output Fuzzy		
	I1	I2	I3	I4	I5	I6, I7, I8	O1	O2	O3, O4, O5
1	MED	Short	Rear				Follow		
2	MED	Short	Right				Pseudo		
3	MED	Short	Left				Pseudo		
4	MED	Short	FR				UNFOL		
5	MED	Short	FL				UNFOL		
6		Long					UNFOL		
7		MED					UNFOL		
8	Near						UNFOL		
9	Far						UNFOL		
10				Near	FL			Lip Shot	
11				Near	FR			Lip Shot	
12				Near	Right			Pseudo	

TABLE V  
IF THEN RULE SAMPLE

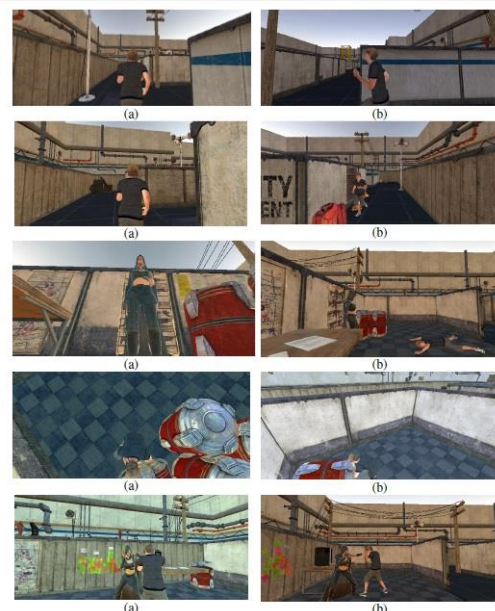
Rule	IF THEN RULE
1	IF distance_p1=Medium AND different_p1=Short AND angle_Y_P1=Rear THEN follow_shot=follow
2	IF distance_p1=Medium AND different_p1=Short AND angle_Y_P1=Right THEN follow_shot=pseudo
3	IF distance_p1=Medium AND different_p1=Short AND angle_Y_P1=Left THEN follow_shot=pseudo
4	IF distance_p1=Medium AND different_p1=Short AND angle_Y_P1=Front Right THEN follow_shot=unfollow
5	IF distance_p1=Medium AND different_p1=Short AND angle_Y_P1=Front Left THEN follow_shot=unfollow
6	IF different_p1=Long THEN follow_shot=unfollow
7	IF different_p1=Medium THEN follow_shot=unfollow
8	IF distance_p1=Near THEN follow_shot=unfollow
9	IF distance_p1=Far THEN follow_shot=unfollow
10	IF distance_p2=Near AND angle_Y_P2=Front Left THEN lip_shot=lip shot
11	IF distance_p2=Near AND angle_Y_P2=Front Right THEN lip_shot=lip shot
12	IF distance_p2=Near AND angle_Y_P2=Right THEN lip_shot=pseudo

IV. HASIL PERCOBAAN

Dalam penelitian ini kami hasilkan beberapa animasi hasil render dari 3D engine. Tampak pada gambar 10 adalah rancangan sistem secara keseluruhan. Tetapi fokus penelitian ini adalah bagian profiling. Sedangkan bagian penempatan posisi kamera dilakukan secara manual berdasarkan story board yang dibuat.



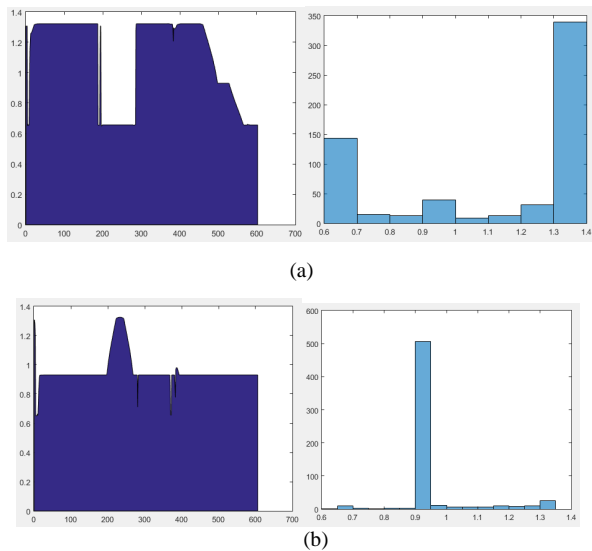
Gambar. 10 Blok sistem yang diusulkan dalam percobaan



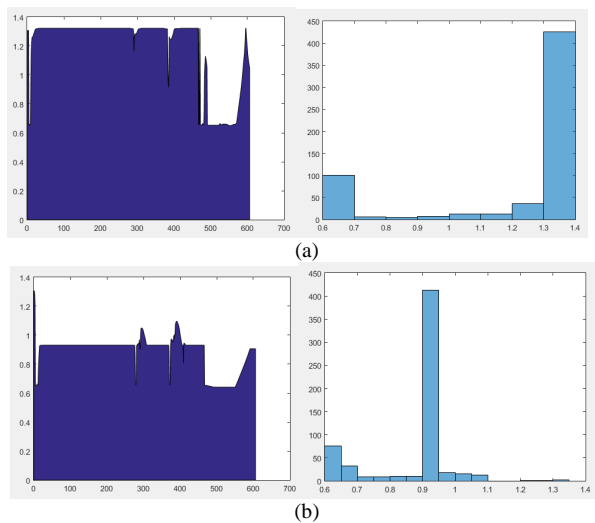
Gambar. 11 Scene sama aksi sama tetapi gaya penempatan posisi kamera yang berbeda.

Hasil animasi merupakan frame yang akan diekstrak nilainya untuk diumpungkan ke dalam sistem. Animasi yang dihasilkan menggunakan Unity 3D engine dan hasilnya dapat dilihat pada gambar 11. Tampak di gambar tersebut aksi yang sama tetapi style yang berbeda. Gambar dengan label (a) merupakan adopsi gaya dari Quentin Tarantino sedangkan yang label (b) merupakan adopsi gaya umum.

Untuk setiap scene dan setiap gaya sutradara akan dihasilkan dua buah diagram yaitu area plot dan histogram diagram. Gambar 12 merupakan area plot dan histogram diagram untuk scene 1. Untuk area plot diagram sumbu x merupakan nomor frame dan sumbu y adalah nilai fuzzy yang dihasilkan. Sedangkan untuk histogram diagram yang merupakan representasi lain dari area plot diagram, sumbu x adalah nilai logika fuzzy dan sumbu y adalah frekuensi kemunculan nilai tersebut.

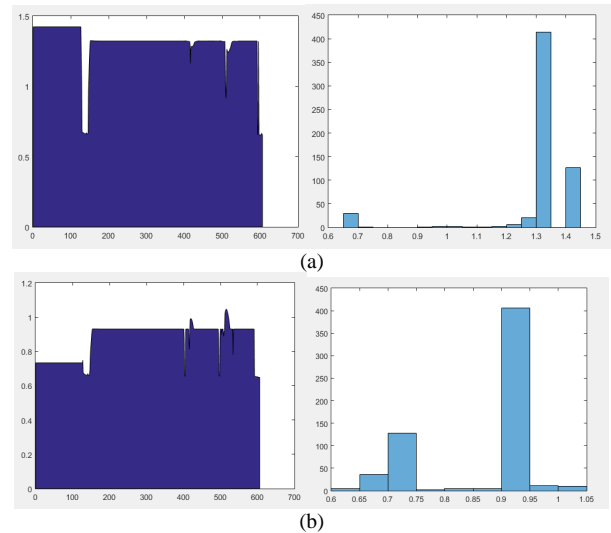


Gambar. 12 Area Plot dan Histogram Diagram Scene 1 (a) Style 1 (b) Style 2



Gambar.13 Area Plot dan Histogram Diagram Scene 2 (a) Style 1 (b) Style 2

Gambar 13 merupakan diagram untuk scene 2 dan gambar 14 merupakan diagram untuk scene 3. Terlihat pada histogram diagram yang merupakan frekuensi kemunculan nilai logika fuzzy tersebut adanya perbedaan visualisasi diagram. Untuk style pertama yang merupakan adopsi gaya dari Quentin Tarantino tampak bahwa modus berada di area kanan angka 1 sedangkan untuk gaya umum berada di sebelah kiri angka 1.



Gambar. 14 Area Plot dan Histogram Diagram Scene 3 (a) Style 1 (b) Style 2

### V. KESIMPULAN

Dalam makalah ini telah dijelaskan pendekatan baru untuk melakukan profil terhadap gaya seorang sutradara berdasarkan penempatan posisi kamera secara otomatis. Berbeda dengan penelitian lain yang menggunakan pendekatan berbasis kuesioner. Digunakan pendekatan berbasis logika fuzzy dikarenakan ada kemiripan antara bahasa dalam dunia sinematografi dan fungsi keanggotaan fuzzy. Selain itu keuntungan menggunakan logika fuzzy karena sifat cepat dan real time dibandingkan dengan pendekatan lainnya, misalkan metode evolusioner semacam swarm yang membutuhkan proses perhitungan berulang-ulang yang tentu saja tidak cocok untuk hal yang membutuhkan real time. Dibandingkan dengan pendekatan berbasis machine learning semacam SVM yang membutuhkan pelatihan dan data training dalam jumlah yang besar. Tetapi tentu saja setiap pendekatan juga memiliki kekurangan yaitu proses akuisisi data membutuhkan waktu dan usaha yang tidak sedikit.

Dalam penelitian ini digunakan lima aturan sinematografi dan 34 variabel yang diambil dari setiap frame pada animasi. Dengan menggunakan visualisasi berbasis diagram area dan histogram berhasil dilakukan analisa bahwa untuk setiap gaya sutradara akan memiliki modus yang tidak sama. Tidak menutup kemungkinan jika akan mengenali gaya sutradara yang lain maka dibutuhkan sebuah aturan sinematografi baru yang belum dibuat pada penelitian ini. Visualisasi hasil dalam bentuk diagram



membuat lebih mudah untuk melakukan profile gaya sutradara secara otomatis.

Kedepan kami merencanakan untuk mengekstrak informasi yang lebih banyak dari frame dan menambahkan beberapa aturan sinematografi sehingga lebih akurat dalam melakukan profiling. Dan kami berharap dapat dilakukan reverse terhadap rule tersebut sehingga kami dapat membuat engine penempatan posisi kamera berdasarkan gaya seorang sutradara.

#### UCAPAN TERIMA KASIH

Ucapan terima kasih diberikan kepada kawan-kawan mahasiswa di Departemen Teknik Elektro Fakultas Teknologi Elektro, Institut Teknologi Sepuluh Nopember Surabaya atas bantuan dan saran-saran dalam penyelesaian makalah penelitian ini dan juga kepada rekan-rekan fotografer dalam bantuan saran aturan sinematik.

#### REFERENSI

- [1] H. Hancock and J. Ingram, *Machinima For Dummies*. Wiley, 2007.
- [2] D. K. Elson and M. O. Riedl, "A lightweight intelligent virtual cinematography system for machinima production," in *Proceedings of the Third AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, Stanford, California, 2007, pp. 8–13.
- [3] J. Hart, *The Art of the Storyboard: A Filmmaker's Introduction*. Elsevier/Focal Press, 2008.
- [4] A. Summerville, J. R. H. Mariño, S. Snodgrass, S. Ontañón, and L. H. S. Lelis, "Understanding Mario: An Evaluation of Design Metrics for Platformers," in *Proceedings of the 12th International Conference on the Foundations of Digital Games*, New York, NY, USA, 2017, pp. 8:1–8:10.
- [5] S. Karakovskiy and J. Togelius, "The Mario AI Benchmark and Competitions," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 4, no. 1, pp. 55–67, Mar. 2012.
- [6] M. Miller, *Assassin's Creed: The Complete Visual History*. Insight Editions, 2015.
- [7] P. Davies, *The Art of Assassin's Creed Unity*. Titan Books, 2014.
- [8] R. Ranon, L. Chittaro, and F. Buttussi, "Automatic camera control meets emergency simulations," *Comput. Graph.*, vol. 48, no. C, pp. 23–34, 2015.
- [9] M. Christie and P. Olivier, "Camera control in computer graphics: models, techniques and applications," in *ACM SIGGRAPH ASIA 2009 Courses*, Yokohama, Japan, 2009, pp. 1–197.
- [10] W. Barry and B. J. Ross, "Virtual photography using multi-objective particle swarm optimization," in *Proceedings of the 2014 Annual Conference on Genetic and Evolutionary Computation*, Vancouver, BC, Canada, 2014, pp. 285–292.
- [11] P. Burelli, L. Di Gaspero, A. Ermetici, and R. Ranon, "Virtual Camera Composition with Particle Swarm Optimization," in *Smart Graphics*, 2008, pp. 130–141.
- [12] S. M. Drucker and D. Zeltzer, "Intelligent Camera Control in a Virtual Environment," in *In Proceedings of Graphics Interface '94*, 1994, pp. 190–199.
- [13] D. A. Prima, B. B. Ferial Java, E. Suryapto, and M. Hariadi, "Secondary camera placement in Machinima using behavior trees," in *2013 International Conference on QiR*, 2013, pp. 94–98.
- [14] A. Z. Fanani, D. A. Prima, B. B. F. Java, E. Suryapto, M. Hariadi, and I. K. E. Purnama, "Secondary camera movement in machinima using path finding," in *2013 International Conference on Technology, Informatics, Management, Engineering and Environment*, 2013, pp. 136–139.
- [15] W. Hu and X. Zhang, "A Semiautomatic Control Technique for Machinima Virtual Camera," in *2012 International Conference on Computer Science and Electronics Engineering*, 2012, vol. 1, pp. 112–115.
- [16] D. B. Christianson, S. E. Anderson, L. He, D. H. Salesin, D. S. Weld, and M. F. Cohen, "Declarative camera control for automatic cinematography," in *Proceedings of the thirteenth national conference on Artificial intelligence - Volume 1*, Portland, Oregon, 1996, pp. 148–155.
- [17] L. Terziman, M. Marchal, F. Multon, B. Arnaldi, and A. Lécuyer, "Personified and Multistate Camera Motions for First-Person Navigation in Desktop Virtual Reality," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 4, pp. 652–661, Apr. 2013.
- [18] S. Benini, L. Canini, and R. Leonardi, "Estimating cinematographic scene depth in movie shots," in *2010 IEEE International Conference on Multimedia and Expo*, 2010, pp. 855–860.
- [19] F. Pereira Ferreira, G. Gelatti, and S. Raupp Musse, "Intelligent Virtual Environment and Camera Control in behavioural simulation," in *Proceedings. XV Brazilian Symposium on Computer Graphics and Image Processing*, 2002, pp. 365–372.
- [20] H. Junaedi, M. Hariadi, and I. K. E. Purnama, "Multi agent with multi behavior based on particle swarm optimization (PSO) for crowd movement in fire evacuation," in *2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP)*, 2013, pp. 366–372.
- [21] A. Jhala and R. M. Young, "Cinematic Visual Discourse: Representation, Generation, and Evaluation," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 2, no. 2, pp. 69–81, Jun. 2010.
- [22] A. Jhala and R. M. Young, "Intelligent Machinima Generation for Visual Storytelling," in *Artificial Intelligence for Computer Games*, P. A. González-Calero and M. A. Gómez-Martín, Eds. New York, NY: Springer New York, 2011, pp. 151–170.
- [23] E. E. S. d. Lima, C. T. Pozzer, M. C. d'Ornellas, A. E. M. Ciarlini, B. Feijó, and A. L. Furtado, "Support Vector Machines for Cinematography Real-Time Camera Control in Storytelling Environments," in *2009 VIII Brazilian Symposium on Games and Digital Entertainment*, 2009, pp. 44–51.
- [24] J. V. Mascelli, *The Five C's of Cinematography: Motion Picture Filming Techniques*. Silman-James Press, 1998.
- [25] D. Bordwell and K. Thompson, *Film art: an introduction*. McGraw Hill, 2008.
- [26] D. Arijon, *Grammar of the Film Language*. Silman-James Press, 1991.
- [27] L. Canini, S. Benini, and R. Leonardi, "Classifying cinematographic shot types," *Multimedia Tools and Applications*, vol. 62, no. 1, pp. 51–73, Jan. 2013.
- [28] Q. Galvane, R. Ronfard, C. Lino, and M. Christie, "Continuity editing for 3D animation," in *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence*, Austin, Texas, 2015, pp. 753–761.
- [29] B. Brown, *Cinematography: Theory and Practice: Imagemaking for Cinematographers, Directors & Videographers*. Focal Press, 2002.
- [30] J. Bennett and C. P. Carter, "Adopting virtual production for animated filmmaking," in *Creative Industries Faculty*, Singapore, 2014.
- [31] M. K. Pratt, *How to Analyze the Films of Quentin Tarantino*. ABDO Publishing Company, 2011.
- [32] Q. Tarantino and G. Peary, *Quentin Tarantino: Interviews*. University Press of Mississippi, 1998.
- [33] P. Woods, *Quentin Tarantino: the film geek files*. Plexus, 2000.