

FACIAL MOTION CAPTURE MENGGUNAKAN ACTIVE APPEARANCE MODEL BERBASIS BLENDER

Tri Afirianto¹⁾, Mochamad Hariadi²⁾

¹⁾ Mahasiswa Program Studi Game Tech Teknik Elektro FTI ITS
Keputih, Sekolilo, Surabaya 60111

²⁾ Asosiasi Profesor, Jurusan Teknik Elektro FTI ITS
email : tri.afirianto09@mhs.ee.its.ac.id¹⁾, mochar@ee.its.ac.id²⁾

Abstrak

Teknologi motion capture dibutuhkan dalam berbagai aplikasi, khususnya animasi yang terus berkembang pesat. Teknik yang digunakan dapat menggunakan penanda maupun tanpa penanda (*markerless*). Metode Active Appearance Model (AAM) mampu melakukan capture titik-titik landmark pada wajah dengan baik. Penelitian ini diarahkan untuk mengembangkan teknik *markerless motion capture* dengan menggunakan AAM pada wajah.

Penelitian ini membandingkan dua metode AAM, yaitu IAIA (*Inverse Additive Image Alignment*) dan ICIA (*Inverse Compositional Image Alignment*), kemudian hasil dari AAM ditampilkan dalam karakter 3D yang di-render dengan menggunakan blender. Sumber data menggunakan tiga dataset, yaitu FRANCK, IMM, dan JIAPEI. Metode evaluasi menggunakan tiga kriteria, yaitu waktu *fitting*, jumlah iterasi, dan jarak penyimpangan yang menunjukkan penyimpangan titik landmark hasil *fitting* terhadap posisi titik landmark yang seharusnya.

Hasil uji coba menunjukkan bahwa rata-rata waktu *fitting* IAIA 33,061844 ms lebih baik daripada ICIA yang memiliki rata-rata 143,295766 ms. Dari segi iterasi, rata-rata IAIA adalah 6,00573, sedangkan ICIA membutuhkan 5,663868 iterasi. Rata-rata jarak penyimpangan IAIA adalah 0,53734 sedangkan ICIA 0,531054. Sedangkan waktu rata-rata proses rendering adalah 21,329753 ms, baik ICIA maupun IAIA.

Kata kunci :

AAM, blender, Facial motion capture, Inverse Additive Image Alignment, Inverse Compositional Image Alignment

1. Pendahuluan

Facial motion capture mempunyai ruang lingkup dan aplikasi yang lebih luas meliputi pergerakan, pengenalan, serta ekspresi wajah. Teknik yang digunakan dapat digolongkan menjadi dua, yaitu dengan penanda dan tanpa penanda (*markerless*). Makalah ini menampilkan *facial motion capture* pada karakter 3D berbasis blender dengan menggunakan AAM yang merupakan salah satu teknik *markerless*. Tujuan dari penelitian ini adalah memproses wajah guna

mendapatkan data wajah dengan AAM kemudian menampilkan ekspresi wajah tersebut dengan menggunakan karakter 3D yang di-render dari blender. Penelitian ini difokuskan pada metode penyesuaian titik-titik (*fitting method*) pada pergerakan wajah dan karakter wajah 3D terlihat tampak depan dan tetap menghadap ke depan. Penelitian [4] menggunakan metode AAM Basic dan *Inverse Compositional Image Alignment* (ICIA), dan disimpulkan bahwa metode ICIA menghasilkan kesesuaian titik yang lebih tinggi, dan hasil penelitian tersebut masih berupa kumpulan data titik-titik wajah. Pada makalah ini, akan digunakan dua metode, yaitu *Inverse Additive Image Alignment* (IAIA) dan ICIA, untuk memperkuat hasil penelitian sebelumnya, dan hasil dari proses AAM akan ditampilkan dalam bentuk karakter 3D.

2. Tinjauan Pustaka

Dalam penelitiannya, Suardinata (2011) melakukan penelitian mengenai penyesuaian titik landmark (*fitting*) dengan menerapkan metode AAM Basic dan ICIA berbasis AAM. Data yang digunakan adalah empat dataset yang diunduh dari internet, yaitu: FRANCK, AGING, BIOD, dan JIAPEI. Penelitian ini diperoleh hasil bahwa ICIA memiliki kecepatan *fitting* 88% - 100% dibanding AAM Basic. Sehingga dianjurkan pengembangan *facial motion capture* menggunakan metode ICIA.

Namun, pada penelitian tersebut hanya membandingkan AAM Basic dan ICIA yang mana AAM Basic merupakan AAM dasar yang belum mengalami perbaikan performa. Sehingga dalam penelitian ini akan dilakukan pengujian *facial motion capture* dengan menggunakan metode IAIA dan ICIA untuk menguji apakah ICIA memiliki kecepatan *fitting* lebih baik daripada IAIA.

2.1 Active Appearance Model (AAM)

Active Appearance Model merupakan metode yang sering digunakan pada pemodelan wajah (*face modeling*) [1]. Namun, AAM dapat juga bermanfaat untuk fenomena lainnya. Dalam aplikasi tertentu, langkah pertama adalah mencocokkan AAM dengan gambar, yakni parameter model ditemukan terlebih dahulu untuk memaksimalkan kecocokan antara contoh model (*model instance*) dengan gambar *input*. Parameter model kemudian dapat

digunakan dalam berbagai aplikasi. Sebagai contoh, parameter ini dapat digunakan sebagai *classifier* untuk menghasilkan algoritma pengenalan wajah. Pada [1], model yang sama dapat digunakan untuk pengenalan wajah, estimasi pose, dan pengenalan ekspresi.

Pencocokan AAM pada gambar merupakan permasalahan non-linear. Pendekatan yang umum dengan cara berulang-ulang memecahkan *incremental additive update* pada parameter (koefisien *shape* dan *appearance*). Dengan nilai parameter *shape* saat ini, dimungkinkan untuk mengubah gambar *input* ke dalam *frame* model koordinat kemudian menghitung *error image* antara contoh model saat ini dengan gambar AAM yang cocok. Pada kebanyakan algoritma sebelumnya, dapat diasumsikan bahwa terdapat linear konstan antara *error image* dengan *additive incremental update* pada parameter. Asumsi tersebut pada umumnya kurang tepat, karena akan menghasilkan *fitting* AAM yang kurang bagus performanya, baik dalam jumlah iterasi untuk konvergen maupun akurasi hasil akhir.

Algoritma *inverse compositional* ditemukan untuk mengatasi hal tersebut. Untuk menggunakan algoritma *inverse compositional*, pertama kita harus mendapat *first order approximation* pada kumpulan operator dari *composition* dan *inversion*. Algoritma *inverse compositional* juga memungkinkan tindakan yang berbeda pada variasi *appearance*. Menggunakan algoritma yang diajukan dalam [2], kita dapat melakukan variasi *appearance* pada langkah *precomputation* dan mengeliminasi besarnya perhitungan dalam komputasi.

Pada dasarnya hanya ada dua jenis linear model *shape* dan *appearance* yaitu model yang *shape* dan *appearance*-nya independen serta model yang memparameterkan *shape* dan *appearance* dengan satu himpunan parameter linear. Kita menyebut himpunan yang pertama sebagai *independent shape and appearance models (independent AAMs)* dan yang kedua adalah *combined shape and appearance (combined AAMs)*.

2.2 Shape

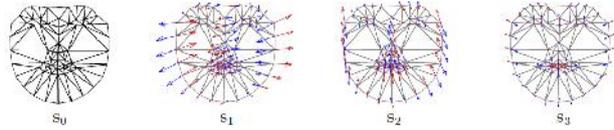
Shape dari *independent AAM* didefinisikan sebagai sebuah *mesh* dan khususnya lokasi verteks dari suatu *mesh*. Secara matematis, *shape* s dari AAM dapat didefinisikan sebagai koordinat dari sebuah verteks v yang membuat *mesh*:

$$s = (x_1, x_2, \dots, x_v, y_1, y_2, \dots, y_v)^T \quad (1)$$

Pada Gambar 1 merupakan contoh *mesh* yang terdiri dari 68 verteks. AAM memungkinkan variasi *shape* linear. Ini berarti *shape* s dapat diekspresikan sebagai *base shape* s_0 ditambah kombinasi linear dari n vektor *shape* s_i :

$$s = s_0 + \sum_{i=1}^n p_i s_i \quad (2)$$

Dalam persamaan 2, koefisien p_i adalah parameter *shape*. Karena re-parameterisasi linear dapat dilakukan kapanpun, maka dapat diasumsikan bahwa vektor s_i adalah orthonormal.



Gambar 1 Linear shape model dari independent AAM

2.3 Appearance

Appearance dari *independent AAM* didefinisikan dalam *base mesh* s_0 . Misalkan s_0 merupakan himpunan piksel $x = (x, y)^T$ yang terletak di dalam *base mesh* s_0 . Sehingga *appearance* AAM merupakan gambar $A(x)$ yang mendefinisikan semua piksel $x \in s_0$. AAM memungkinkan variasi *appearance* linear. Ini berarti *appearance* $A(x)$ dapat diekspresikan sebagai *base appearance* $A_0(x)$ ditambah kombinasi linear dari m *appearance* gambar $A_i(x)$:

$$A(x) = A_0(x) + \sum_{i=1}^m \lambda_i A_i(x) \quad \forall x \in s_0 \quad (3)$$



Gambar 2 Linear appearance model dari independent AAM

2.4 Combined AAM

Jika *independent AAM* mempunyai parameter terpisah untuk *shape* p dan parameter *appearance*, *combined AAM* hanya menggunakan sekumpulan himpunan parameter $c = (c_1, c_2, \dots, c_l)^T$ untuk parameterisasi *shape*:

$$s = s_0 + \sum_{i=1}^l c_i s_i \quad (4)$$

dan *appearance*:

$$A(x) = A_0(x) + \sum_{i=1}^l c_i A_i(x) \quad (5)$$

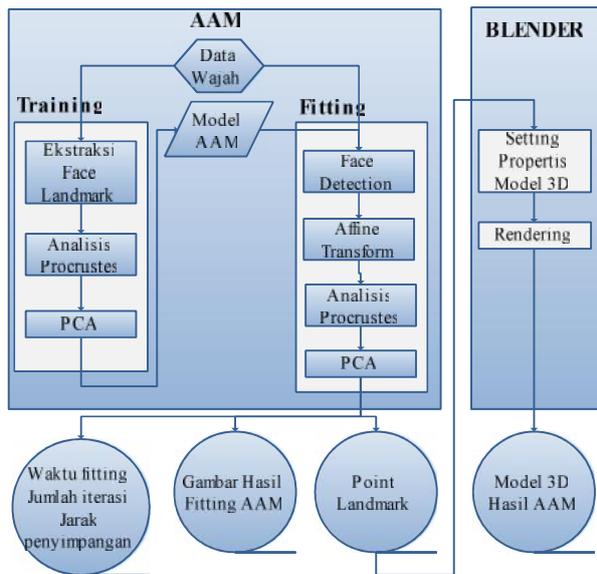
Karena itu bagian *shape* dan *appearance* dari model saling berpasangan. Adanya pasangan ini mempunyai kerugian. Sebagai contoh, pada kasus ini berarti diasumsikan bahwa vektor s_i dan $A_i(x)$ adalah orthonormal. Hal ini akan membatasi pemilihan dalam algoritma *fitting*. Di sisi lain, AAM mempunyai beberapa keuntungan, yaitu formulasi *combined* lebih umum dan membutuhkan parameter yang lebih sedikit.

2.5 Blender

Blender merupakan aplikasi komputer grafik 3D gratis dan *open-source* yang digunakan untuk membuat animasi film, visual efek, aplikasi 3D yang interaktif, maupun *video game*. Oleh karena itu, banyak orang yang menggunakan aplikasi ini untuk membuat karakter 3D beserta animasinya. Salah satu karakter yang dibuat dengan menggunakan blender adalah Ludwig, yang dibuat oleh Jason Pierce.

Ludwig merupakan karakter blender yang memiliki kelengkapan animasi penuh. Memiliki fitur lengan, pergerakan kepala dan tulang belakang, dan pergerakan bibir. Ludwig disediakan untuk memberikan kemudahan bereksperimen dalam hal *humanoid rig* dan animasi.

3. Metode Penelitian



Gambar 3 Metode penelitian

Penelitian yang dilakukan adalah dengan menggunakan tiga dataset yang dapat diunduh secara gratis dari internet. Detil tentang dataset dapat dilihat pada Tabel 1. Data ini terdiri atas gambar dan *annotation*. *Annotation* adalah pemetaan titik *landmark* pada masing-masing gambar. Jumlah gambar adalah jumlah seluruh gambar beserta *annotation*-nya. Jumlah subjek adalah jumlah orang yang digunakan sebagai objek foto. Jumlah titik adalah jumlah titik *landmark* wajah dataset tersebut.

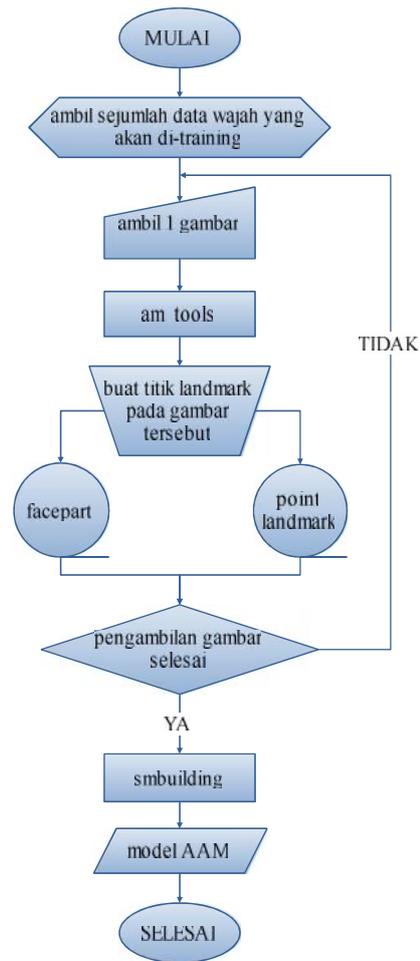
Selanjutnya sebagian data wajah diambil untuk dilakukan proses *training*, dan sebagian untuk proses *fitting*. Dari hasil *fitting* AAM, akan dianalisis mengenai waktu *fitting*, jumlah iterasi, dan jarak penyimpangan. Sedangkan data *point distribution* diproses secara matematis untuk mendapatkan nilai sesuai dengan properti model 3D kemudian di-*render* untuk menghasilkan model 3D.

Tabel 1 Sumber data wajah

Dataset	Franck	IMM	JIAPEI
Jumlah gambar	5000	240	67
Training	1000	60	33
Testing	300	180	34
Jumlah subjek	1	40	1
Jumlah titik	68	58	109
Resolusi	720*576	640*480	640*480
RGB	RGB	RGB	RGB

3.1 Proses Training

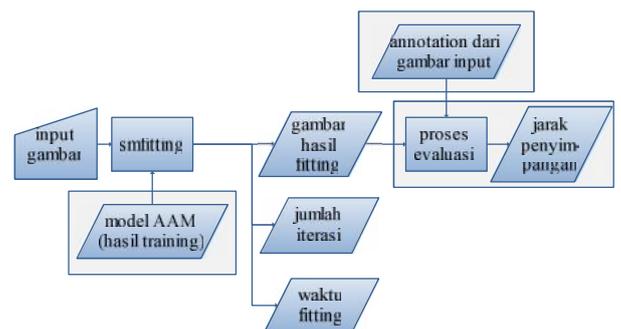
Proses *training* secara detil dapat dilihat pada Gambar 4.



Gambar 4 Flowchart proses training

3.2 Proses Fitting

Detil langkah proses *fitting* dapat dilihat pada Gambar 5.



Gambar 5 Alur proses fitting

3.3 Proses Rendering

Langkah-langkah yang dilakukan dalam proses *rendering* adalah sebagai berikut.

1. Siapkan sejumlah data titik *landmark* hasil fitting.
2. Membuat suatu fungsi untuk mengubah titik *landmark* menjadi suatu nilai yang sesuai dengan properti model 3D dan masukkan nilai tersebut ke dalam properti model 3D.

3. Lakukan proses *rendering* model 3D dengan menggunakan aplikasi blender.
4. Ulangi langkah 1 hingga semua data hasil *fitting* selesai dimodelkan.

3.4 Proses Evaluasi

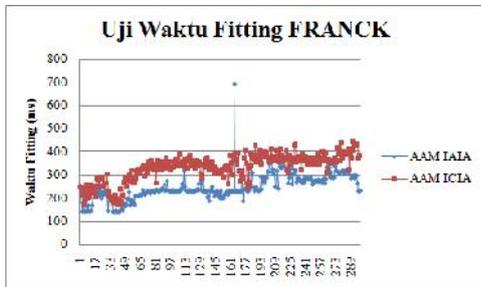
Sesuai dengan Gambar 5, untuk melakukan evaluasi jarak penyimpangan, maka diperlukan *annotation* sebagai pembanding. Terdapat 3 nilai evaluasi yang dihasilkan, yaitu:

1. Waktu *fitting*, yaitu waktu yang diperlukan untuk melakukan proses *fitting* hingga konvergen pada setiap gambar.
2. Jumlah iterasi, yaitu jumlah perulangan yang dilakukan selama proses *fitting* hingga konvergen.
3. Jarak penyimpangan, yaitu selisih jarak titik *landmark* antara hasil *fitting* dengan titik *landmark* yang seharusnya, semakin besar jarak penyimpangan maka semakin besar *error* konvergen yang dihasilkan.

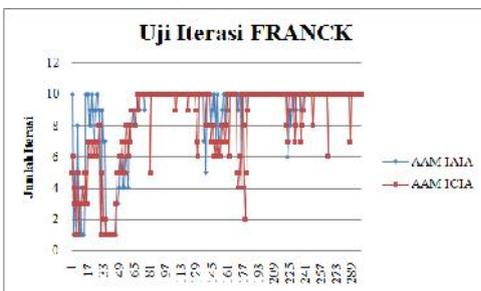
4. Hasil dan Pembahasan

Uji coba dilakukan dengan menggunakan komputer Core i5 2.30GHz (4 CPU), memori 4GB. Metode AAM IAIA dan ICIA diuji dengan menggunakan tiga *dataset* (Tabel 1) dengan kriteria evaluasi berupa waktu *fitting*, jumlah iterasi, dan jarak penyimpangan.

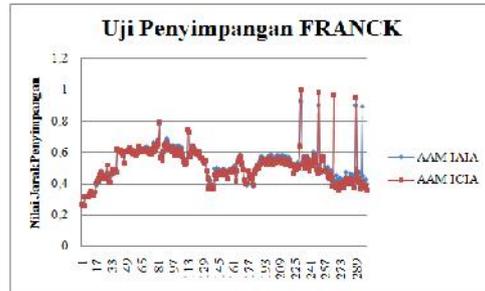
4.1 Dataset FRANCK



Gambar 6 Hasil uji waktu *fitting* dataset FRANCK



Gambar 7 Hasil uji jumlah iterasi dataset FRANCK

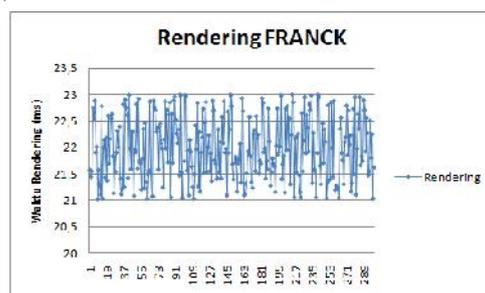


Gambar 8 Hasil uji jarak penyimpangan dataset FRANCK

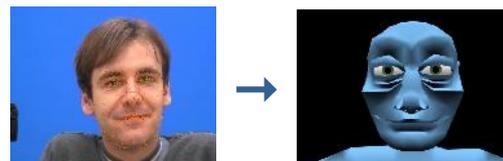
Tabel 2 Rekapitulasi data pengujian dataset FRANCK

		FRANCK	
		AAM IAIA	AAM ICIA
rata-rata	waktu fitting	244,4265567	332,86113
	iterasi	8,744966443	8,32885906
	jarak	0,530723289	0,513835151
prosentase performa lebih baik	waktu fitting	57,7%	42,3%
	iterasi	48,8%	51,2%
	jarak	49,2%	50,8%

Berdasarkan hasil pengujian Gambar 6 didapatkan hasil bahwa waktu *fitting* yang dibutuhkan oleh dataset FRANCK dengan metode IAIA adalah 57,7%, dengan rata-rata 244,4265567 ms, sedangkan dari ICIA memiliki prosentase 42,3% (Tabel 2), dengan rata-rata 332,2612709 ms. Pada pengujian jumlah iterasi (Gambar 7), IAIA memiliki nilai lebih rendah yaitu 48,8%, dengan rata-rata 8,744966443, sedangkan ICIA memiliki nilai 51,2%, dengan rata-rata 8,32885906. Dan pada pengujian jarak penyimpangan (Gambar 8), IAIA memiliki jarak penyimpangan lebih besar dengan nilai keunggulan 49,2%, dengan rata-rata 0,530723289, dibandingkan ICIA yang bernilai 50,8%, dengan rata-rata 0,513835151.



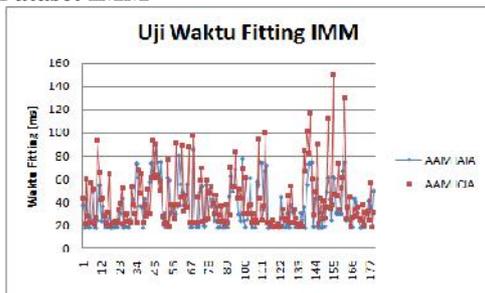
Gambar 9 Hasil uji waktu *rendering* dataset FRANCK



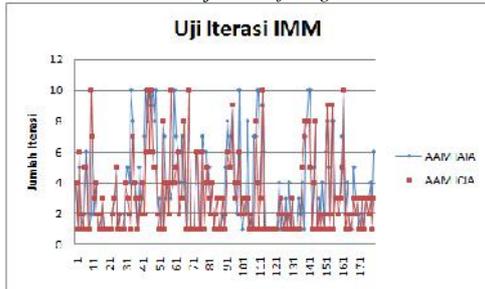
Gambar 10 Salah satu hasil AAM disertai dengan *rendering* model 3D data FRANCK

Berdasarkan Gambar 9, bahwa waktu rata-rata yang diperlukan untuk me-render model 3D adalah 21,9717 ms, dan hasil model 3D-nya dapat dilihat pada Gambar 10.

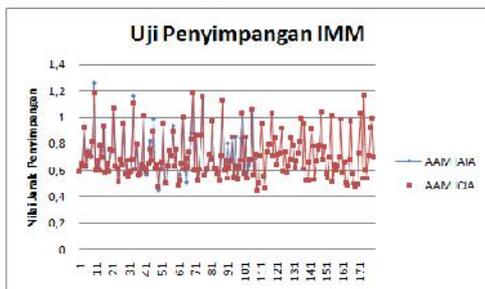
4.2 Dataset IMM



Gambar 11 Hasil uji waktu fitting dataset IMM



Gambar 12 Hasil uji jumlah iterasi dataset IMM



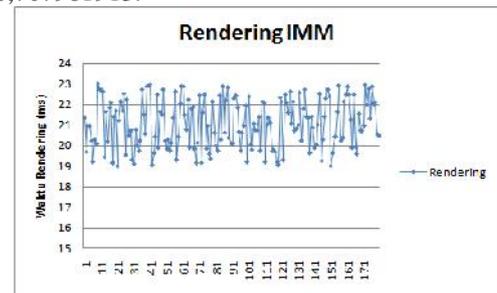
Gambar 13 Hasil uji jarak penyimpangan dataset IMM

Tabel 3 Rekapitulasi data pengujian dataset IMM

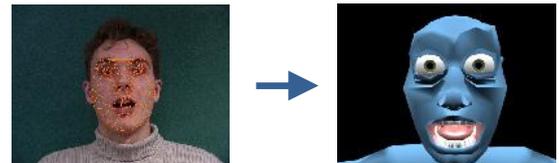
		IMM	
		AAM IAIA	AAM ICIA
rata-rata	waktu fitting	36,27479556	41,91208111
	iterasi	3,772222222	3,133333333
	jarak	0,710513775	0,707981913
prosentase performa lebih baik	waktu fitting	53,6%	46,4%
	iterasi	45,4%	54,6%
	jarak	49,9%	50,1%

Berdasarkan hasil pengujian Gambar 11 diperoleh hasil waktu fitting yang dibutuhkan oleh dataset IMM dengan metode IAIA memiliki prosentase 53,6%, dengan rata-rata 36,27479556 ms, sedangkan dari ICIA hanya memiliki prosentase 46,4% (Tabel 3), dengan rata-rata 41,91208111 ms. Pada pengujian jumlah iterasi (Gambar 12), IAIA memiliki nilai lebih rendah yaitu 45,4%, dengan rata-rata 3,772222222, sedangkan ICIA memiliki nilai 54,6%, dengan rata-rata 3,133333333. Sedangkan pada pengujian jarak penyimpangan (Gambar 13), IAIA memiliki jarak penyimpangan lebih besar dengan nilai keunggulan 49,9%, dengan rata-rata 0,710513775,

dibandingkan ICIA yang bernilai 50,1%, dengan rata-rata 0,707981913.



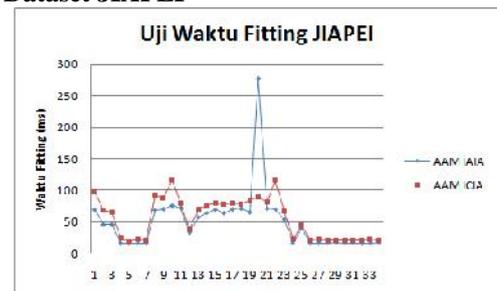
Gambar 14 Hasil uji waktu rendering dataset IMM



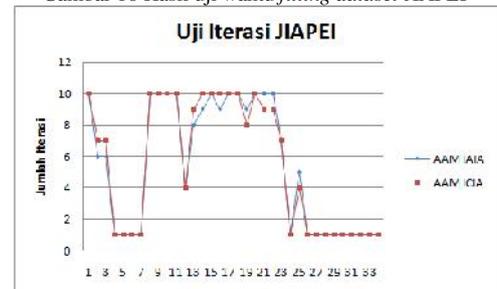
Gambar 15 Salah satu hasil AAM disertai dengan rendering model 3D data IMM

Untuk dataset IMM, waktu rata-rata yang diperlukan untuk me-render model 3D adalah 21,0505 ms (Gambar 14), dan hasil model 3D-nya dapat dilihat pada Gambar 15.

4.3 Dataset JIAPEI

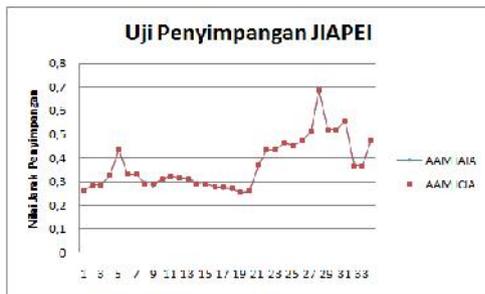


Gambar 16 Hasil uji waktu fitting dataset JIAPEI



Gambar 17 Hasil uji jumlah iterasi dataset JIAPEI

Berdasarkan hasil pengujian Gambar 16 didapatkan hasil bahwa waktu fitting yang dibutuhkan oleh dataset JIAPEI dengan metode IAIA sebesar 52,4%, dengan rata-rata 50,00041471 ms, sedangkan dari ICIA hanya sebesar 47,6% (Tabel 4), dengan rata-rata 55,11408824 ms. Pada pengujian jumlah iterasi (Gambar 17), IAIA memiliki nilai lebih tinggi yaitu 50,1%, dengan rata-rata 5,5, sedangkan ICIA memiliki nilai 49,9%, dengan rata-rata 5,529411765. Dan pada pengujian jarak penyimpangan (Gambar 18), IAIA memiliki jarak penyimpangan hampir sama dengan nilai keunggulan 50%, dengan rata-rata 0,370782003, dibandingkan ICIA yang hanya bernilai 50%, dengan rata-rata 0,37134525.

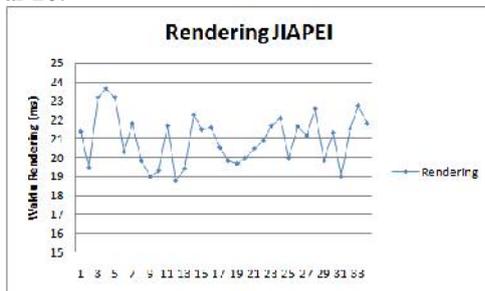


Gambar 18 Hasil uji jarak penyimpangan dataset JIAPEI

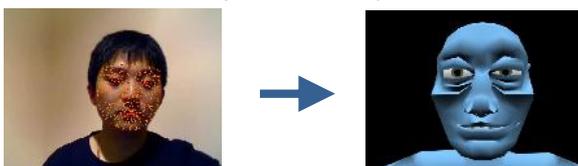
Tabel 4 Rekapitulasi data pengujian dataset JIAPEI

		JIAPEI	
		AAM IAIA	AAM ICIA
rata-rata	waktu fitting	50,00041471	55,11408824
	iterasi	5,5	5,529411765
	jarak	0,370782003	0,37134525
prosentase performa lebih baik	waktu fitting	52,4%	47,6%
	iterasi	50,1%	49,9%
	jarak	50%	50%

Untuk dataset JIAPEI, waktu rata-rata yang diperlukan untuk me-render model 3D adalah 20,96706 ms (Gambar 19), dan hasil model 3D-nya dapat dilihat pada Gambar 20.



Gambar 19 Hasil uji waktu rendering dataset JIAPEI



Gambar 20 Salah satu hasil AAM disertai dengan rendering model 3D data JIAPEI

3. Dari parameter jarak penyimpangan, IAIA dan ICIA memiliki rata-rata hampir sama, yaitu 49,7%:50,3%.
4. Waktu rata-rata proses *rendering* dibawah 30 ms, jumlah waktu tersebut cukup cepat apabila digunakan ketika proses *realtime*.

Penelitian-penelitian selanjutnya diharapkan menguji-coba metode IAIA maupun ICIA untuk *facial motion capture* secara *realtime*, disertai dengan model 3D dimulai dari sedikit polygon hingga banyak polygon. Namun demikian, model 3D Ludwig juga sekiranya diujicoba. Selain metode ICIA, dapat diteliti pula untuk metode-metode lainnya yang mampu memberikan waktu *fitting* yang cepat disertai dengan jarak penyimpangan minimal.

Daftar Pustaka

- [1] Lanitis, Andreas., Taylor, Chris J., Cootes, Timothy F., 1997, *Automatic Interpretation and Coding of Face Image Using Flexible Models*, Vol 19(7): 743-756, IEEE Transactions on Pattern Analysis and Machine Intelligence.
- [2] Matthews, Ian., Baker, Simon., 2004, *Active Appearance Models Revisited*, Second Edition, Addison Wiley, Canada.
- [3] Pe, Jia., 2010, *2D Statistical Models*, Computer Science, The University of Essex.
- [4] Suardinata, I Wayan., 2011, *Facial Motion Capture Menggunakan Algoritma Inverse Compositional pada AAM*, Tesis, Jurusan Teknik Elektro, Institut Teknologi Sepuluh Nopember.

Biodata Penulis

Tri Afirianto, memperoleh gelar Sarjana Teknik (S.T.), Jurusan Teknik Elektro FTI ITS pada tahun 2008. Tahun 2009 melanjutkan studinya di Program Studi Game Tech – Jaringan Multi Media Jurusan Teknik Elektro FTI ITS hingga saat ini.

Mochamad Hariadi, memperoleh gelar Sarjana Teknik (S.T.), Jurusan Teknik Elektro FTI ITS pada tahun 1995. Penulis meperoleh gelar M.Sc. dan Ph.D. dari Graduate School of Information Science Universitas Tohoku Jepang pada tahun 2003 dan 2006. Saat ini, penulis mengajar sebagai dosen di Jurusan Teknik Elektro FTI ITS. Penelitian penulis tertarik pada Video dan Image Processing, Data Mining dan Intelligent System. Penulis merupakan anggota dari IEEE, dan juga anggota dari IEICE.

5. Kesimpulan dan Saran

Dari penelitian dan percobaan yang telah dilakukan, maka penggunaan AAM dalam *facial motioin capture* adalah sebagai berikut.

1. Berdasarkan kecepatan waktu *fitting*, IAIA lebih baik dibandingkan dengan ICIA, yaitu 56,5%:43,5%.
2. Dari segi jumlah iterasi, metode IAIA lebih rendah daripada ICIA, yaitu 48,5%:51,5%.