

# Kontrol Gerakan Objek 3D *Augmented Reality* Berbasis Titik Fitur Wajah dengan POSIT

Heri Pratikno<sup>1</sup>

**Abstract**— *Augmented Reality (AR) is a technique in computer science that combines real world conditions with the computer computation results in the form of 2D or 3D graphics. In this study, a method and application implementation in Augmented Reality environment with markerless was discussed. In the markerless technique, interaction process between humans and computers becomes more natural and intuitive than the marker based techniques. Markerless techniques applied in this study used facial feature points so that the result is more robust for the head object does not produce light. The main problem in this area of research is how the process of controlling the 3D object movement does not experience anomalies such as railroad phenomenon, where the eyes catch as if the distance between the two railroads narrower at the farther view. This study used POSIT (Pose from Orthography and Scale with Iteration), where the position and orientation of the 3D objects were projected in orthographic from facial feature points with scaling, so the change in the distance between the face and the webcam were proportional to the big - small changes of 3D objects. The next step was iteration process carried out four to five times to look for the smallest error factor to obtain the best pose.*

**Intisari**— *Augmented Reality (AR) adalah sebuah teknik dalam bidang ilmu komputer yang mengkombinasikan antara kondisi dunia nyata dengan data hasil komputasi dari sebuah komputer dalam bentuk grafis 2D maupun 3D. Pada penelitian ini, dibahas sebuah metode dan implementasi aplikasi di lingkungan Augmented Reality secara markerless. Pada metode markerless proses interaksi antara manusia dan komputer lebih alami dan intuitif dibandingkan dengan metode berbasis marker. Metode markerless yang diterapkan pada penelitian ini menggunakan titik-titik fitur wajah supaya hasilnya lebih robust karena objek kepala tidak menghasilkan cahaya sehingga mengandalkan titik-titik fitur wajah. Permasalahan utama pada bidang penelitian ini adalah bagaimana proses pengontrolan pergerakan objek 3D tersebut tidak mengalami anomali seperti rel kereta api, semakin jauh mata memandang seakan-akan jarak antara kedua rel kereta api semakin menyempit. Penelitian ini menggunakan metode POSIT (Pose from Orthography and Scale with Iteration), posisi dan orientasi objek 3D diproyeksikan secara ortografi dari titik-titik fitur wajah dengan pen-skalaan, sehingga perubahan jarak antara wajah dan webcam berbanding lurus dengan perubahan besar dan kecilnya objek 3D. Langkah berikutnya dilakukan proses iterasi empat sampai lima kali untuk mencari faktor kesalahan terkecil sehingga didapatkan pose terbaik.*

**Kata Kunci**— *Augmented Reality, Titik Fitur Wajah, POSIT, Objek 3D*

## I. PENDAHULUAN

*Markerless* dalam *Augmented Reality (AR)* bertujuan untuk menghasilkan interaksi antara komputer dan pengguna lebih alami dan intuitif dibandingkan dengan model *marker*.

Jurusan Sistem Komputer, Institut Bisnis dan Informatika Stikom Surabaya, Jln. Raya Kedung Baruk 98 Surabaya 60298 INDONESIA (tlp: 031-8721731; e-mail: heri@stikom.edu)

Deteksi yang berbasis titik fitur wajah merupakan salah satu model *markerless* yang paling banyak menjadi perhatian oleh para praktisi dan peneliti. Dimana proses deteksi dan pengenalan yang berbasis titik fitur wajah dapat diterapkan dalam berbagai bidang kehidupan.

Dalam dunia *computer vision* maupun *Augmented Reality*, proses deteksi dan pengenalan wajah dapat dilakukan dalam berbagai bidang, diantaranya: deteksi dan pengenalan bentuk fitur wajah, gerak wajah, gerak mata, gerak dan bentuk mulut sampai dengan proses deteksi dan pengenalan ekspresi dari wajah seseorang. Masalah utama dalam proses deteksi dan pengenalan berbasis wajah manusia diantaranya adalah kebutuhan sistem untuk mendekati waktu nyata (*real time*), pelacakan (*tracking*) gerak titik fitur wajah serta bagaimana melakukan kontrol dan menampilkan *image* 2D maupun objek 3D pada sumbu koordinat [1].

Proses estimasi *pose* (*pose estimation*) mempunyai tingkatan komputasi yang lebih sulit dan krusial dalam menentukan jumlah dan penempatan titik fitur wajah, transformasi model objek 3D serta akurasi, dibandingkan dengan proses deteksi wajah (*face detection*) maupun pengenalan wajah (*face recognition*). Selain akurasi, proses *recovery fitting* antara *image* wajah 2D dan model objek 3D juga perlu diperhitungkan karena pergerakan terlalu cepat dari *image* wajah 2D aktor yang berfungsi sebagai *landmark* dari model objek 3D dapat menyebabkan kehilangan *frame* pelacakan dalam suatu *scene*.

Berdasarkan penelitian dari Daniel F. DeMenthon [2], metode POSIT (*Pose from Orthography and Scaling with Iterations*) mempunyai komputasi sistem estimasi *pose* bisa sampai dua puluh lima kali lebih cepat dari pada metode Yuan [3] maupun metode Lowe [4] karena pada metode POSIT sudah tidak diperlukan lagi proses inialisasi estimasi *pose* awal yang dilakukan secara manual dan *inverse* matrik dalam *looping* iterasinya. Metode POSIT akan mengekstraksi *pose* objek dengan pendekatan model 3D secara antropometrik (*rigid anthropometric*) yang berkorespondensi dengan titik fitur *image* wajah.

Komputasi posisi dan orientasi dari sebuah model objek dengan menggunakan konfigurasi geometrik titik fitur wajah mempunyai tahapan penting seperti proses kalibrasi, kartografi, pelacakan dan pengenalan objek. Beberapa metode sistem komputasi estimasi *pose* telah dikenalkan oleh Tsai [5], Lowe dan Yuan.

Metode yang dikenalkan oleh Tsai sangat berguna ketika panjang fokus (*focal length*) dari kamera, distorsi lensa dan pusat gambar tidak diketahui. Pada metode Lowe dan Yuan mempunyai dua kelemahan yang cukup signifikan, yaitu: pertama, sebuah perkiraan atau pendekatan *pose* harus dimulai dengan proses iterasi. Kelemahan kedua adalah setiap langkah proses iterasi diperlukan *pseudoinverse matrix* Jacobian dengan dimensi  $2N \times 6$  pada metode Lowe dan  $N \times 6$  untuk Yuan.

Dengan  $N$  adalah jumlah titik fitur yang harus ditemukan, hal ini menjadikan sebuah operasi komputasional yang lebih rumit dan waktunya lebih lama. Metode *Fully Projective* [6] merupakan pengembangan dari metode Lowe dan berhasil diimplementasikan dalam penelitian [7] dengan hasil penelitian sebagai berikut: tingkat kesalahan rotasi rata-rata sebesar  $4^\circ$  dan tingkat kesalahan translasinya rata-rata sebesar 5 cm.

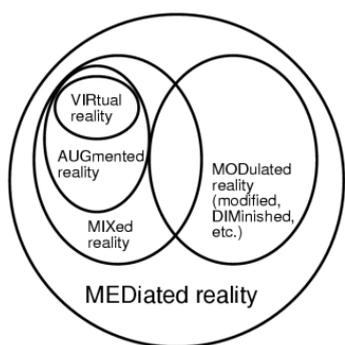
Penelitian ini bertujuan untuk memberikan metode alternatif dalam penerapan sistem interaksi antara manusia dan komputer dalam lingkungan *Augmented Reality* secara *markerless*. Dimana proses komputasinya tidak diperlukan kontak fisik secara langsung antara pengguna dan input sensor komputer sehingga interaksi antara manusia dan komputer lebih alami dan intuitif.

II. KONTROL GERAKAN OBJEK 3D

A. *Augmented Reality*

*Augmented Reality* (AR) merupakan kombinasi teknologi bidang komunikasi dan informasi yang meletakkan (*overlay*) benda maya dua dimensi maupun tiga dimensi ke dalam dunia nyata tiga dimensi. Sehingga suatu benda yang sebelumnya hanya dapat dilihat secara dua dimensi, dapat muncul sebagai objek virtual yang digabungkan dalam lingkungan nyata secara *real time*.

Teknologi *Augmented Reality* menambah, melengkapi atau meningkatkan realitas yang ada, dengan menambahkan elemen-elemen hasil komputasi yang didapatkan dari masukan data yang bisa berupa audio, video, grafis maupun data GPS. Sedangkan *Virtual Reality* (VR) akan menggantikan dunia nyata atau realitas yang ada untuk disimulasikan secara penuh di komputer dalam bentuk grafis, sehingga pengguna akan merasakan dalam lingkungan yang sintetik. Dalam relasi konsep yang lebih umum *Augmented Reality* dan *Virtual Reality* merupakan suatu realitas yang termediasi (*mediated reality*), sebagaimana ditunjukkan seperti pada Gbr. 1.



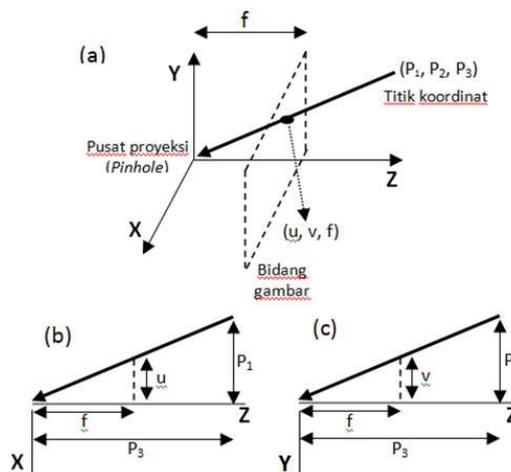
Gbr. 1 Lingkungan realitas.

B. *Proyeksi 3D ke 2D*

Hubungan antara titik koordinat pada ruang nyata (3D) dan titik koordinat korespondensinya pada *image* (2D) merupakan hasil proyeksi dari ruang nyata ke bidang *image*. Gbr. 2 menunjukkan model proyeksi ortografik (*weak perspective*), merupakan model proyeksi 3D ke 2D yang banyak digunakan

karena kesederhanaan dalam penggunaannya. Titik-titik di ruang nyata yang diproyeksikan ke bidang *image* ini tergantung pada jarak dari pusat proyeksi  $f$  (*focal length*). Hubungan antara titik koordinat bidang *image* ( $u, v$ ) dan titik koordinat nyata  $P = (p_1, p_2, p_3)$  dapat dinyatakan seperti pada persamaan (1) :

$$u = (p_1/p_3)f, \quad v = (p_2/p_3)f \tag{1}$$



Gbr. 2 *Proyeksi perspective geometry*, a. Sinar pada Titik p, b. View dari Sumbu Y dan c. View dari Sumbu X

*Proyeksi weak perspective* ini *valid* jika titik – titik koordinat dalam ruang nyata itu harus cukup jauh dari kamera, model objek kepala 3D ditransformasikan terlebih dulu dengan nilai rotasi dan translasi tertentu setelah itu baru diproyeksikan ke bidang *image* 2D.

C. *Metode Viola-Jones*

Algoritma Viola-Jones ditemukan oleh Paul Viola dan Michael Jones [8], algoritma Viola-Jones merupakan algoritma yang paling banyak digunakan untuk mendeteksi wajah. Proses pendeteksian wajah dilakukan dengan mengklasifikasikan sebuah gambar melalui sebuah pengklasifikasi yang dibentuk dari data latih. Data latih yang digunakan oleh algoritma ini berjumlah 5000 *image* wajah dan 9400 *image* bukan wajah sehingga menghasilkan akurasi sistem sebesar 95% dengan data positif salah sebesar 1 : 14.084.

Deteksi wajah, merupakan langkah pertama dalam proses identifikasi, klasifikasi *image* dilakukan berdasarkan nilai dari sebuah fitur, penggunaan fitur dilakukan karena pemrosesan fitur berlangsung lebih cepat dibandingkan pemrosesan *image* per piksel. Tiga tujuan utama dari algoritma Viola-Jones adalah sebagai berikut: komputasi fitur (*feature computation*), seleksi fitur (*feature selection*) dan ketepatan waktu nyata (*real timeliness*). Kelebihan deteksi wajah secara *real time* dengan menggunakan algoritma Viola-Jones diantaranya:

- 1) *Robust*, mempunyai tingkat deteksi tinggi untuk pelacakan gambar dalam *image* (*True Positive*) dengan tingkat kesalahan (*False Positive*) yang rendah.

2) *Real Time*, mempunyai komputasi fitur yang cepat dimana dalam prakteknya aplikasi seharusnya memproses sekurang-kurangnya 4 *frame* per detik.

D. Lucas-Kanade Optical Flow

*Optical flow* [9] didefinisikan sebagai suatu gerakan yang terlihat dikarenakan adanya *image brightness* (terang gelapnya *image*). misal  $I(x,y,t)$  adalah *image brightness* yang berubah terhadap waktu sebagai gambaran dari urutan beberapa *image* (*image sequences*), ada dua asumsi :

1) *Brightness*  $I(x,y,t)$  bergantung pada koordinat  $x, y$  yang ada dalam bagian yang lebih besar dari *image*.

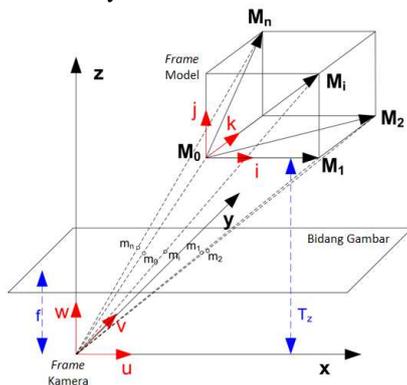
2) *Brightness* dari setiap titik dari suatu objek yang bergerak tidak berubah terhadap waktu.

Misal ada beberapa objek di dalam *image* atau beberapa titik dari suatu objek bergerak maka setelah waktu  $dt$  perubahan letak objek tersebut menjadi  $(dx, dy)$ .

E. Metode POSIT

Dalam penelitian ini dijelaskan metode untuk mencari *pose* sebuah objek dari sebuah gambar tunggal, metode ini merupakan gabungan dari dua algoritma, pertama adalah POS (*Pose Orthography and Scalling*), yang merupakan pendekatan proyeksi perspektif dengan proyeksi skala ortografi serta pencarian rotasi matrik, translasi vektor dari objek. Algoritma kedua POSIT (*POS with Iterations*) menggunakan looping iterasi pendekatan *pose* yang didapatkan dari POS dalam rangka untuk menghitung proyeksi skala ortografi lebih baik dari titik-titik fitur [10].

POSIT merupakan sebuah metode yang cepat dan algoritma iteratif yang akurat untuk menemukan *pose* 6DOF – enam derajat kebebasan bergerak (orientasi dan translasi) dari sebuah model 3D atau *scene* yang terkait atau berhubungan dengan kamera yang memberikan korespondensi titik objek 3D dan *image* 2D. Pada Gbr. 3 memperlihatkan model *pinhole* kamera dengan pusat proyeksi  $O$ , dan *image plane* di *focal length*  $f$  dengan asumsi nilai dari *focal length* dan pusat *image* telah diketahui besarnya.



Gbr. 3 Proyeksi perspektif  $m_i$  untuk titik model  $M_i$  [2]

Sebuah model 3D dengan titik-titik fitur  $M_0, M_1, \dots, M_i, \dots, M_n$  berada di posisi *frustum* kamera. *Frame* dari koordinat model berpusat di  $M_0$ , dengan titik  $M_i$  koordinatnya telah diketahui  $(X_i, Y_i, Z_i)$  dalam *frame* model dan koordinat yang

belum diketahui dalam *frame* kamera. Proyeksi *image* dari  $M_i$  telah diketahui dan disimpan dalam parameter  $m_i$  yang mempunyai koordinat *image*  $(u'_i, v'_i)$ . Pada proyeksi perspektif sebuah titik model 3D  $(X, Y, Z)$  akan diproyeksikan pada *image plane* dengan rumus (2) :

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = K \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ \mathbf{0}_{1 \times 3} & 1 \end{bmatrix} + \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (2)$$

Dimana  $\mathbf{R}$  adalah merepresentasikan orientasi rotasi matrik dari *frame* kamera dengan keterkaitan *frame world*. Sedangkan  $\mathbf{T}$  merupakan translasi vektor dari titik pusat kamera  $O$  ke  $M_0$  yang dinyatakan dalam *frame* kamera.  $\mathbf{o}$  adalah sebuah matrik nol dan  $\mathbf{K}$  adalah matrik kamera dengan jarak fokus  $f$  dan  $(c_x, c_y)$  merupakan titik prinsipal sebagaimana persamaan (3), dimana nilai  $\mathbf{K}$  seharusnya diketahui.

$$R = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \text{ dan } K = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

Agar dapat menggunakan normalisasi koordinat *image* maka transformasi pada persamaan (4) perlu diterapkan.

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = K^{-1} \begin{bmatrix} u' \\ v' \\ w' \end{bmatrix} \quad (4)$$

Sehingga dari persamaan (4) akan menjadi :

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = [R \quad T] \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (5)$$

dan matrik proyeksi sekarang diberikan oleh  $P = [R | T]$ , untuk mencari solusi dalam permasalahan *pose* adalah bagaimana menemukan matrik  $R$  dan  $T$  yang sepenuhnya menggambarkan 6DOF,  $r_1, r_2$  dan  $r_3$  didefinisikan oleh persamaan (6) :

$$r_1 = \begin{bmatrix} r_{11} \\ r_{12} \\ r_{13} \end{bmatrix}, r_2 = \begin{bmatrix} r_{21} \\ r_{22} \\ r_{23} \end{bmatrix}, r_3 = \begin{bmatrix} r_{31} \\ r_{32} \\ r_{33} \end{bmatrix} \quad (6)$$

dari persamaan (5) dapat ditulis seperti pada persamaan (7) :

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} r_1^T & T_x \\ r_2^T & T_y \\ r_3^T & T_z \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (7)$$

atau semua elemen matrik proyeksi dibagi dengan  $T_z$ ,

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} r_1^T/T_z & T_x/T_z \\ r_2^T/T_z & T_y/T_z \\ r_3^T/T_z & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (8)$$

Dari persamaan (8), memberikan,

$$w_i = 1 + \frac{r_3}{T_z} (X_i, Y_i, Z_i) \tag{9}$$

dan menerapkan transposnya pada dua persamaan yang tersisa,

$$[u \ v] = [X \ Y \ Z \ 1] \begin{bmatrix} r_1/T_z & r_2/T_z \\ T_x/T_z & T_y/T_z \end{bmatrix} \tag{10}$$

dengan menggunakan sejumlah titik  $n$ , persamaan (10) dapat dikembangkan menjadi,

$$\begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \\ \vdots & \vdots \\ u_{n-1} & v_{n-1} \\ u_n & v_n \end{bmatrix} = \underbrace{\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 \\ X_2 & Y_2 & Z_2 & 1 \\ \vdots & \vdots & \vdots & \vdots \\ X_{n-1} & Y_{n-1} & Z_{n-1} & 1 \\ X_n & Y_n & Z_n & 1 \end{bmatrix}}_M \begin{bmatrix} r_1/T_z & r_2/T_z \\ T_x/T_z & T_y/T_z \end{bmatrix} \tag{11}$$

dimana  $M$  merupakan model matrik yang mendefinisikan struktur dari model 3D yang digunakan dan  $(u_i, v_i)$  adalah titik koordinat proyeksi *image*  $(X_i, Y_i, Z_i)$ . Solusi dari persamaan (11) untuk memberi parameter *pose*,

$$\begin{bmatrix} r_1/T_z & r_2/T_z \\ T_x/T_z & T_y/T_z \end{bmatrix}_{4 \times 2} = M_{4 \times n}^{-1} \begin{bmatrix} u_1 & v_1 \\ u_2 & v_2 \\ \vdots & \vdots \\ u_{n-1} & v_{n-1} \\ u_n & v_n \end{bmatrix}_{n \times 2} \tag{12}$$

Hasilnya lebih mudah untuk mengambil  $T_z, T_x, T_y, r_1$  dan  $r_2$  semenjak baris rotasi matrik ortogonal adalah  $r_3 = r_1 \times r_2$  dan *pose* sepenuhnya bisa didefinisikan. Dapat diketahui setidaknya ada empat titik korespondensi *non-coplanar* yang diperlukan, jika tidak maka matrik  $M$  adalah tunggal. Pendekatan inilah yang disebut *Pose from Orthography and Scaling* (POS). Misalkan untuk mendapatkan nilai tetap *pose* dari  $w_i$ .

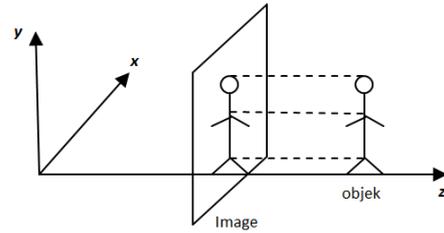
Perspektif *image*  $(u_i, v_i)$  dari titik *world* 3D yang berkaitan dengan *image*  $(u_i^w, v_i^w)$  dihasilkan oleh penskalaan kamera ortografik menurut:

$$\begin{aligned} u_i^w &= w_i u_i \\ v_i^w &= w_i v_i \end{aligned} \tag{13}$$

terminasi tersebut dapat ditentukan jika *pose* kamera sudah diketahui dengan menggunakan persamaan (9). Algoritma POSIT dimulai dengan asumsi bahwa titik proyeksi *image* identik dengan titik penskalaan ortografik *image*, sehingga  $w_i = 1, i = 1, \dots, n$ . Berdasarkan asumsi tersebut maka *pose* kamera dapat ditentukan melalui solusi sebuah sistem persamaan linier (12).

Solusi tersebut hanya perkiraan pada saat pendekatan  $w_i = 1$ , bagaimanapun juga akan membuat lebih akurat estimasi dari *pose* objek. Akurasi dari parameter  $w_i$  dapat ditingkatkan dengan melakukan estimasi ulang menggunakan persamaan (9). Gbr. 4 menampilkan model perspektif skala ortografik digunakan secara iteratif dalam proses komputasi perspektif secara penuh dari *pose*. Proses ini diulang sampai terjadi *pose*

yang konvergen. Metode POSIT tidak memerlukan proses inialisasi estimasi *pose* terlebih dulu, sangat cepat karena proses konvergensi hanya memerlukan sekitar empat sampai lima iterasi, robust terkait dengan pengukuran *image* dan permasalahan kalibrasi kamera.



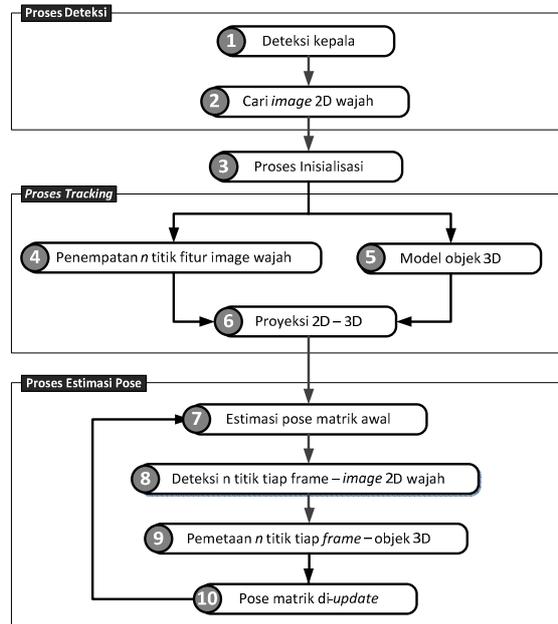
Gbr. 4 Proyeksi perspektif ortografik [10]

### III. METODOLOGI

Metode penelitian yang dilakukan pada penelitian ini adalah sebagai berikut:

#### A. Perancangan Sistem

Dibahas mengenai perencanaan dan pembuatan sistem untuk estimasi *pose* model 3D dalam lingkungan *Augmented Reality* berbasis titik fitur wajah, tampak pada Gbr. 5. Proses *tracking* dari awal *image* terdeteksi untuk tiap *frame* sampai dengan terbentuknya *image* 2D kepala dari aktor. Setelah *image* 2D kepala terdeteksi dengan garis kotak deteksi *tracking*, maka data tersebut siap diolah untuk diambil titik-titik fitur penting pada *image* wajah yang akan digunakan sebagai *landmark* peletakkan (*overlay*) model objek 3D dalam lingkungan *Augmented Reality*.



Gbr. 5 Bagan sistem

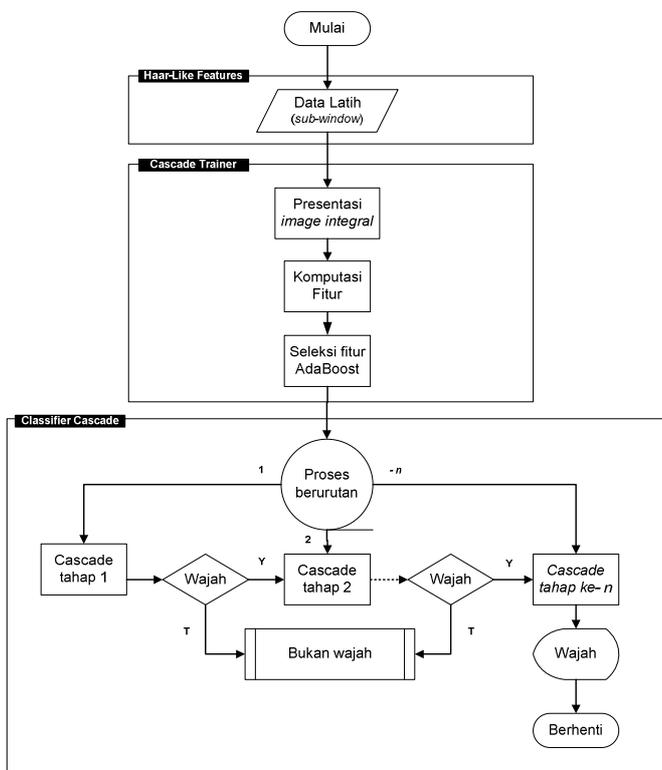
#### B. Proses Deteksi

Pada tahapan ini bertujuan untuk pengambilan gambar (*grabbing*) tiap *frame* dalam waktu nyata (*real time capturing*)

dari sebuah *scene* menggunakan kamera tunggal. Pada aplikasi tersebut, resolusi *image* yang akan ditampilkan pada *form* desain *window* adalah *fix*, artinya tidak mengikuti ukuran *box preview*, yaitu: 640 x 480 piksel. Yang perlu diperhatikan adalah posisi aktor atau *image* kepala juga harus diatur penempatan posisinya agar bisa ditangkap dengan sempurna oleh kamera.

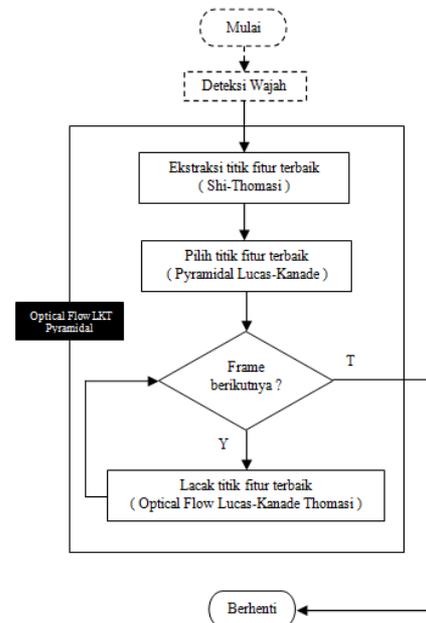
C. Proses Tracking wajah

Setelah proses deteksi pada *frame* pertama selesai, langkah selanjutnya adalah diperlukan informasi ekstraksi dari ekspresi wajah yang direpresentasikan pada beberapa urutan dari *image* sebagaimana pada Gbr. 6. Pada saat otot wajah berkontraksi maka akan menyebabkan perubahan pada fitur wajah dan hasilnya mempengaruhi efek secara visual. Pergerakan dari titik-titik wajah, misalkan alis, mata dan mulut mempunyai informasi relasi yang kuat dalam menunjukkan ekspresi dari wajah.



Gbr. 6 Diagram alur proses deteksi metode Viola-Jones

Untuk melacak titik fitur wajah dari sebuah *image*, pada penelitian ini menggunakan algoritma Kanade-Lucas Thomasi (KLT). Gbr. 7 menampilkan prinsip dasar dari ketiga tahapan metode KLT dalam melacak titik fitur wajah, yaitu: mengekstraksi titik fitur wajah (*feature extraction*), memilih titik fitur wajah (*feature selection*) dan melacak titik fitur (*feature tracking*).

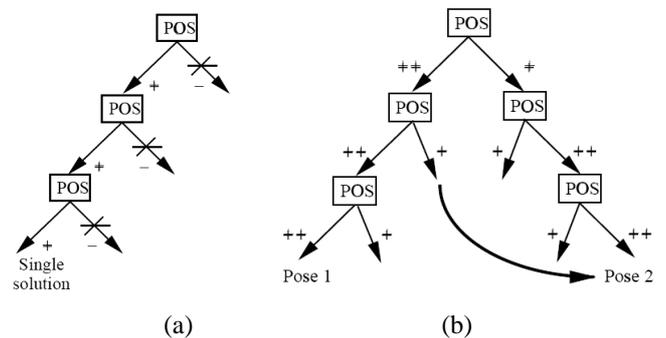


Gbr. 7 Proses Lucas-Kanade Thomasi

D. Proses Estimasi Pose

Pada dasarnya estimasi *pose* atau disebut juga *extrinsic camera calibration* adalah proses untuk mengekstraksi informasi yang terkait dengan informasi posisi dan orientasi titik yang korespondensi dari sebuah model objek dan *image* yang diperoleh dari kamera. *Pose* dari sebuah objek merupakan posisi dan orientasi pada objek yang mempunyai enam arah kebebasan dalam bergerak (*Six Degree Of Freedom - 6DOF*).

Dengan informasi dari 6DOF dapat menampilkan beberapa objek yang relatif pada *pose* yang diketahui atau *pose* terukur dari objek yang diintegrasikan pada objek virtual ke gambar nyata atau video, langkah inilah yang merupakan kunci dari *Augmented Reality*. Untuk setiap *pose* pada semua titik-titik objek yang berada di depan kamera (semua  $Z_i > 0$ ), apabila  $Z_i$  kurang atau sama dengan nol maka *pose* tersebut akan diabaikan.



Gbr. 8 Proses POS, a. Satu POSE dan b. Dua Pose [11]

Pada Gbr. 8 (a) pada proses POS mempunyai satu *pose* yang layak pada setiap prosesnya. dimana tanda + (plus) adalah *pose* yang layak atau terdeteksi dan tanda - (minus) merupakan *pose* yang tidak terdeteksi atau dibuang. Sedangkan pada Gbr. 8 (b) menunjukkan proses POS yang mempunyai dua keluaran *pose* yang layak pada setiap proses iterasinya.

#### IV. HASIL DAN PEMBAHASAN

Sampai pada tahapan ini hasil yang telah dicapai adalah terselesaikannya proses deteksi, proses pelacakan, proses pemetaan proyeksi model objek 2D ke 3D dan korespondensi 3D ke 2D. Semua proses tersebut telah diimplementasikan dalam tiga bentuk tampilan GUI *window* yang berfungsi untuk memudahkan kontrol proses input dan output pada masing-masing tahapan.

##### A. Tampilan GUI Sistem

1) *Tampilan GUI Berbasis Teks*: Pada GUI berbasis teks ini akan menampilkan proses output dari hasil program yang terkait dengan keluaran: lamanya waktu proses deteksi, koordinat posisi dari estimasi *pose*, matrik GL matrix, rotasi matrix dan translasi *vector*.

2) *Tampilan GUI Tracking*: Tampilan GUI *tracking* berfungsi untuk menampilkan *window* yang berisi *image* 2D wajah aktor yang di-*capture* tiap *frame* oleh kamera untuk menggerakkan model objek 3D. Hasil *tracking* merupakan hasil penerapan dari algoritma Viola-Jones berupa garis kotak pada *image* kepala dalam gambar dengan ukuran *sub-window* 24 x 24 piksel yang berfungsi untuk menampilkan hasil proses inialisasi titik - titik fitur wajah oleh Lucas-Kanade Thomasi.

Semakin banyak jumlah titik-titik fitur pada *image* wajah 2D, maka akan semakin *robust* dalam proses pergerakan model objek 3D. titik-titik fitur tersebut berfungsi sebagai *landmark* proses pembangkitan model objek 3D dalam bentuk seperti kepala manusia, sebagaimana tampak pada Gbr. 9.

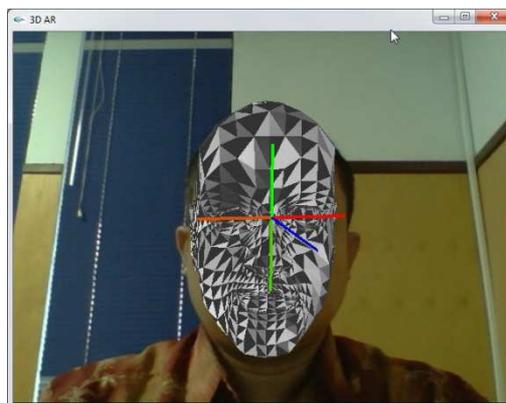


Gbr. 9 GUI Tracking

3) *Tampilan GUI Utama Augmented*: Pada tampilan pada Gbr. 10, merupakan bentuk tampilan GUI utama *Augmented Reality* yang menampilkan model objek 3D bentuk kepala manusia dengan permukaan *polygon* segitiga sebanyak 2.700

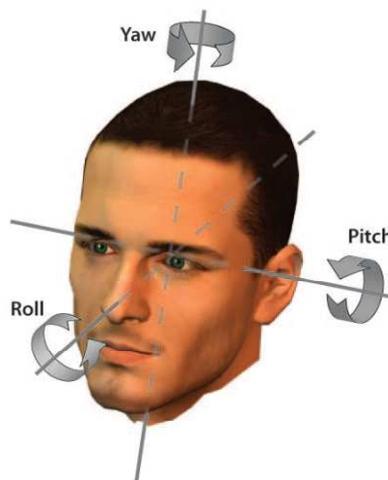
buah. Berdasarkan teori Parke bahwa *surface* wajah lebih dari 250 *polygon* yang terbagi lebih dari 400 *vertex* sudah cukup mencapai tingkat realitis dari wajah sesungguhnya.

Format *file* model objek kepala 3D yang digunakan pada penelitian ini mempunyai ekstensi *.raw*, dimana *file* gambar dengan ekstensi ini biasanya dihasilkan secara langsung dari kamera lama dengan merk, seperti : Fuji, Xerox dan sebagainya. Menurut Steve Anger (*February*, 1996) untuk merubah atau mengkonversi *file* jenis 3D dapat menggunakan utility RAW2POV, dimana pada utilitas ini akan merubah titik - titik geometri *facet* dalam bentuk *triangular*.



Gbr. 10 GUI tampilan utama AR

Model objek 3D dalam bentuk seperti kepala manusia tersebut bisa digerakkan kearah sumbu koordinat XYZ dan kearah sudut pergerakan RPY (*Roll, Pitch, Yaw*) seperti terlihat pada Gbr. 11.



Gbr. 11 Arah rotasi pada sudut RPY

##### B. Hasil Pengujian Sistem

Pengujian sistem yang dilakukan pada penelitian ini, dibahas dari sudut pandang akurasi, ketangguhan pelacakan, serta waktu nyata (*real time*). Nilai akurasi terkait dengan

ketepatan dalam pelacakan, ketangguhan melacak dalam perbedaan intensitas cahaya dan kecepatan gerakan. Sedangkan sisi *real time* menunjukkan, apakah proses pergerakan model objek 3D berjalan dalam waktu nyata ?.

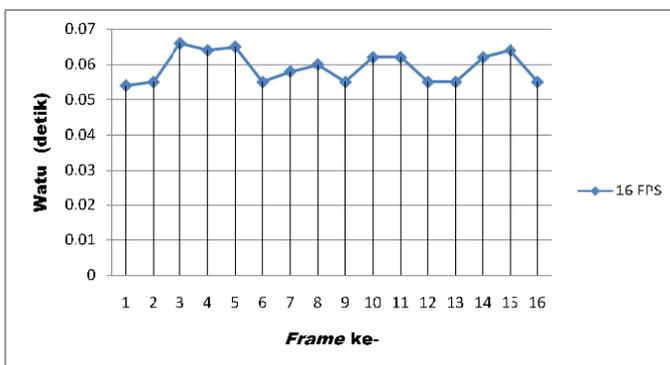
1) *Akurasi Pelacakan*: Implementasi pada sistem, akurasi pelacakan cukup baik karena bisa menggerakkan model objek kepala 3D pada tampilan layar GUI utama hampir sama dengan pergerakan si aktor. Proses pelacakan (*tracking*) mampu mendeteksi *image* wajah dan menggerakkan model objek kepala 3D rotasi pada sudut *Pitch* sebesar  $129.1^{\circ}$  ke arah atas dan bawah, sedangkan pada arah rotasi pada sudut *Yaw* dengan berpaling ke kanan dan kiri besar simpangan sudutnya sebesar  $179.8^{\circ}$ .

2) *Ketangguhan Pelacakan*: Ketangguhan dalam pelacakan *image* wajah 2D serta pergerakan model objek kepala 3D dipengaruhi oleh pencahayaan ruangan, metode Lucas-Kanade Tomasi berdasarkan pada tingkat kecerahan objek (*brightness*) pada cahaya ruangan yang normal. Pencahayaan pada ruangan tidak boleh terlalu gelap atau terlalu terang karena hal itu bisa mengakibatkan beberapa pelacakan titik fitur wajah bergeser dari tempat seharusnya.

Apabila banyak titik fitur wajah yang hilang atau bergeser maka bisa menyebabkan pada tingkat akurasi pelacakan. Apabila terlalu sedikit titik fitur yang disetting maka akan menyebabkan tampilan animasi model objek kepala 3D di monitor jadi bergetar (*tremor*). Oleh karena itu pada penelitian ini banyaknya titik fitur *image* wajah diberikan sebanyak 200 buah titik fitur.

3) *Waktu Nyata*: Untuk mengetahui berapa banyaknya *frame* tiap detiknya pada proses inialisasi *tracking* dari sebuah kamera ada banyak metode yang bisa dilakukan. Fungsi untuk mendeteksi posisi *image* wajah dari gambar kepala 2D dalam area kotak *tracking* menggunakan ukuran *window* 40 x 40 piksel merupakan implementasi dari algoritma Viola-Jones dengan *Haar-like features* dan *trained cascades*.

Hasil keluaran dari penelitian ini adalah sebanyak 16 *frame* pada tiap detiknya, terlihat pada Gbr. 12 dan Tabel I. Dengan *frame* sebesar itu sudah memenuhi unsur waktu secara *real time*, dimana waktu secara *real time* mensyaratkan minimal ada 4 *frame* dalam satu detiknya.



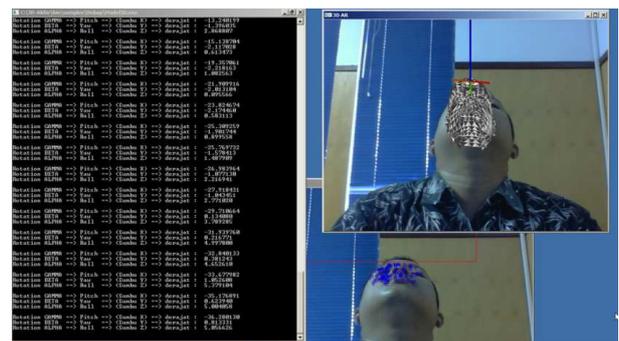
Gbr. 12 Grafik waktu deteksi *frame* tiap detik

TABEL I  
SATU *FRAME PER SECOND*

t Deteksi ke-	t yang diperlukan (detik)	Jumlah (detik)	Σ FPS
1	0.054	0.947 (≈ 1 detik)	1
2	0.055		
3	0.066		
4	0.064		
5	0.065		
6	0.055		
7	0.058		
8	0.060		
9	0.055		
10	0.062		
11	0.062		
12	0.055		
13	0.055		
14	0.062		
15	0.064		
16	0.055		

4) *Posisi dan Orientasi Model Objek pada Sudut RPY*: Posisi dan orientasi pergerakan model 3D dalam AR pada arah enam derajat kebebasan (6DOF) mempunyai enam parameter posisi dan orientasi ( $t_x, t_y, t_z, \alpha, \beta, \gamma$ ) dimana parameter translasi vektor  $[t_x, t_y, t_z]^T$  dan rotasi sudut yang direpresentasikan dengan orientasi RPY ( $\alpha = \text{Roll}, \beta = \text{Pitch}, \gamma = \text{Yaw}$ ). Pergerakan dan pergeseran sudut dari model objek 3D tersebut akan berotasi minimal pada dua orientasi RPY, permasalahan ini timbul dikarenakan menggunakan sudut Euler.

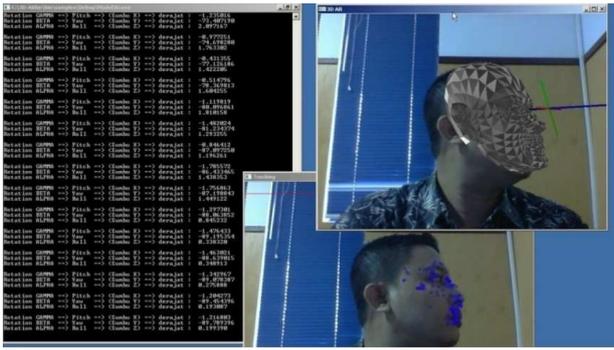
5) *Rotasi pada Sudut Pitch ( $\beta$ )*: Pada Gbr. 13, menunjukkan rotasi model kepala 3D pada sumbu X (*Pitch*,  $\beta$ ) tetap dengan pergeseran besar sudut pada sumbu Y (*Yaw*) dan Z (*Roll*). Dimana rotasi model objek 3D tersebut dari titik awal (0, 0, 0) terhadap pergeseran sudut keatas (0,  $y', z'$ ) dengan pergeseran sudut maksimal sebesar  $65,9^{\circ}$ . Sedangkan pergeseran sudut ke bawah (0,  $y', z'$ ) maksimal sebesar  $63,2^{\circ}$ .



Gbr. 13 Model objek 3D menghadap keatas

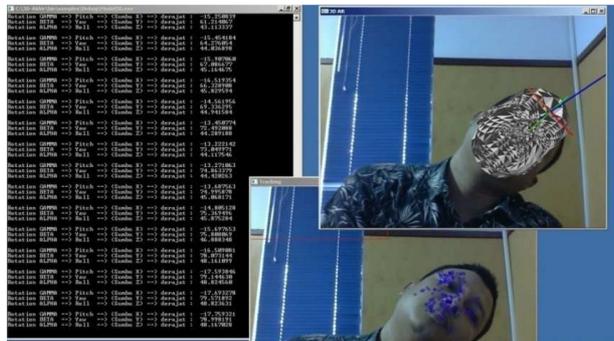
6) *Rotasi pada Sudut Yaw ( $\gamma$ )*: Gbr. 14, menunjukkan rotasi model kepala 3D pada sumbu Y ( $\gamma$ ) tetap dengan pergeseran besar sudut pada sumbu X (*Pitch*) dan Z (*Roll*). Dimana rotasi model objek 3D dari titik awal (0, 0, 0) terhadap pergeseran sudut ke kanan ( $x', 0, z'$ ) dengan pergeseran sudut maksimal sebesar  $89,79^{\circ}$ . Sedangkan rotasi

model 3D berpaling kearah kiri mengalami pergeseran sudut sebesar  $90.01^\circ$ .



Gbr. 14 Model objek 3D berpaling ke kanan

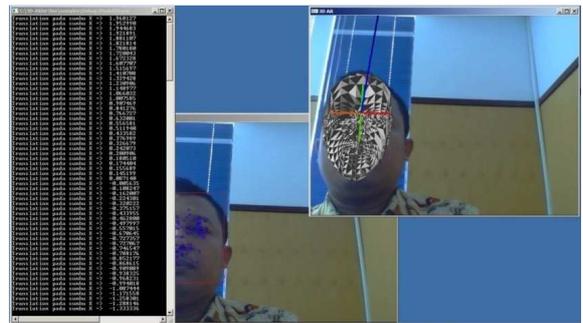
7) *Rotasi pada Sudut Roll ( $\alpha$ )*: Pada Gbr. 15, menunjukkan rotasi model kepala 3D pada sumbu Z ( $\alpha$ ) tetap dengan pergeseran besar sudut pada sumbu X (*Pitch*) dan Y (*Yaw*). Dimana rotasi model objek 3D dari titik awal  $(0, 0, 0)$  terhadap pergeseran sudut rebah ke bahu kanan dan bahu kiri  $(x', y', 0)$  dengan pergeseran sudut maksimal sebesar  $90^\circ$ .



Gbr. 15 Rotasi pada sudut kanan *Roll*

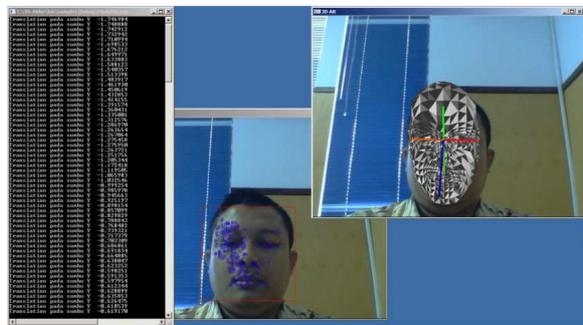
8) *Translasi Model Objek pada Arah XYZ*: Translasi adalah pergeseran suatu benda sepanjang suatu garis lurus, garis yang dimaksud adalah sejajar dengan sumbu X, sejajar sumbu Y, sejajar sumbu Z atau sejajar garis sembarang dalam ruang. Akibat dari translasi ini menyebabkan benda tersebut akan berpindah tempat atau bergeser dari tempat semula dengan bentuk dan orientasi tidak berubah. Gerak adalah perpindahan secara terus-menerus, dengan adanya translasi maka akan diperoleh kesan gerak.

9) *Translasi pada Sumbu X*: Pada Gbr. 16, menampilkan pergerakan model objek 3D kepala manusia pada sumbu X dari arah kanan dan kiri, besarnya pergeseran geometri dari arah kanan ke kiri sebesar 1,66 satuan vektor unit serta pergeseran model 3D dari kiri ke kanan sebesar 0,48 satuan vektor unit.



Gbr. 16 Translasi pada sumbu X dari arah kanan ke kiri

10) *Translasi pada Sumbu Y*: Translasi pada sumbu koordinat Y ditunjukkan seperti pada Gbr. 17 yang menampilkan pergerakan model objek 3D kepala manusia pada sumbu Y dari atas dan bawah. Rata-rata besarnya pergeseran geometri dari bagian atas ke bagian bawah layar sebesar 1,63 satuan vektor unit. Adapun pergeseran model 3D dari bagian bawah layar ke bagian atas layar sebesar 0,33 satuan vektor unit.



Gbr. 17 Translasi pada sumbu Y dari atas ke bawah

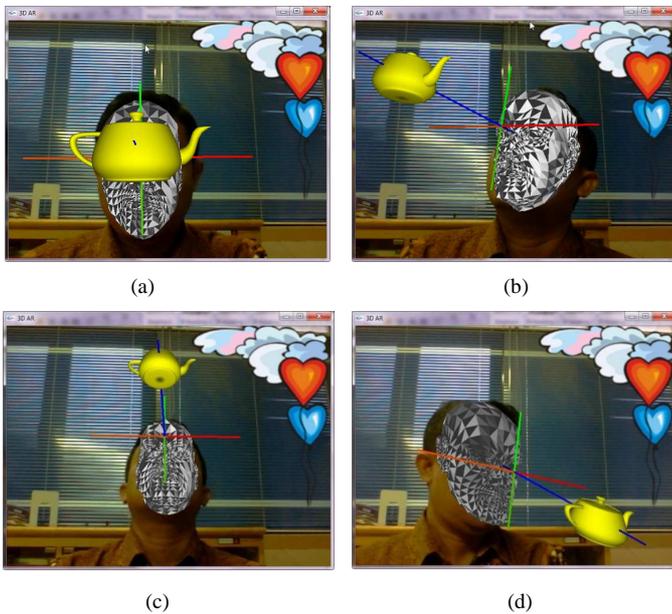
11) *Translasi pada Sumbu Z*: Selanjutnya translasi pada sumbu koordinat Z ditunjukkan seperti pada Gbr. 18 yang menampilkan pergerakan model objek 3D kepala manusia pada sumbu Z dari arah depan dan belakang. Rata-rata besarnya pergeseran geometri dari arah depan (mendekati kamera) ke arah belakang (menjauhi kamera) sebesar 1,73 satuan vektor unit. Adapun pergeseran model 3D dari arah belakang (menjauhi kamera) ke arah depan (mendekati kamera) sebesar 3,62 satuan vektor unit.



Gbr. 18 Translasi pada sumbu Z dari depan ke belakang

### C. Hasil Akhir Kontrol Pergerakan Objek 3D

Model objek 3D yang akan dibuat dan dikontrol pada penelitian ini adalah bentuk *solid* dari sebuah *teapot*, dimana pergerakan dari model objek 3D *teapot* tersebut diproyeksikan secara orthogonal dengan model 3D objek bentuk kepala manusia, tampak seperti pada Gbr. 19. Sehingga pergerakannya bisa dilakukan secara 6DOF, yaitu: translasi pada sumbu koordinat XYZ dan rotasi pada sudut ruang RPY (*Roll*, *Pitch* dan *Yaw*).



Gbr. 19 Pergerakan model objek 3D : a. Posisi awal, b. Pojok kanan atas  
c. Atas-bawah, d. Pojok kiri bawah

## V. KESIMPULAN DAN SARAN

### A. Kesimpulan

Dari hasil perancangan sistem, implementasi sistem dan pengujian sistem, maka didapatkan kesimpulan sebagai berikut.

Hasil keluaran dari sistem yang diterapkan pada penelitian ini sebesar 16 *frame per second* (FPS), sehingga sudah termasuk dalam waktu nyata (*real time*) yang mensyaratkan minimal 4 FPS.

Posisi dan orientasi pergerakan model kepala 3D sudah bisa dirotasikan pada sudut *Pitch* ( $\beta$ ), sudut *Yaw* ( $\gamma$ ) dan sudut *Roll* ( $\alpha$ ).

Model kepala 3D bisa ditranslasikan pada sumbu koordinat XYZ.

Diperlukan minimal 200 titik fitur pada *image* wajah 2D agar model objek 3D tidak bergetar (*tremor*).

Sistem pelacakan pergerakan tidak *robust* pada area pelacakan yang tertutupi (*occluding area*).

### B. Saran

Beberapa saran untuk penelitian lebih lanjut: Sebaiknya menggunakan jumlah titik fitur yang lebih banyak untuk

mendapatkan tingkat ketahanan gerak dari model objek kepala 3D agar tidak bergetar (*tremor*).

Diperlukan pencahayaan ruangan yang merata karena jika pencahayaan terlalu gelap atau terang bisa membuat beberapa titik fitur wajah bergeser atau hilang.

Bisa ditingkatkan lebih lanjut dengan penerapan metode Fuzzy, AAM dan sebagainya sehingga bisa mengekspresikan model objek 3D.

## REFERENSI

- [1] M. Haller, M. Billinghurst, Bruce Thomas, *Emerging Technologies of Augmented Reality: Interface and Design*. IDEA GROUP PUBLISHING, 2007.
- [2] Daniel F. DeMenthon, Larry S. Davis, *Model-Based Object Pose in 25 Lines of Code*. International Journal of Computer Vision. Volume 15. Numbers 1-2. pages 123-141, 1995.
- [3] Yuan. J.S.C., *A General Photogrammetric Method for Determining Object Position and Orientation*. IEEE Trans. on Robotics and Automation. vol. 5. pp. 129-142. 1989.
- [4] Lowe, D.G, *Perceptual Organization and Visual recognition*, Kluwer Academic Publishers. 1985.
- [5] Tsai, R.Y., *An Efficient and Accurate Camera Calibration Technique for 3D Machine Vision*. Proceeding of IEEE Conference on Computer Vision and Pattern Recognition. Miami Beach. Florida. pp. 364-374. 1986.
- [6] Araujo H., Carceroni Rodrigo L. Brown Christopher M., *A Fully Projective Formulation to Improve the Accuracy of Lowe's Pose Estimation Algorithm*, Computer Vision and Images Understanding. Volume 70. Pages 227-238. 1998.
- [7] Resmana Lim. Davina. Sivia R., *Pelacakan dan estimasi Pose Video wajah 3 Dimensi*. Jurnal Teknik Elektro. volume 2. nomer 2. UK Petra. 2002.
- [8] Sunu Wibarama, *Fundamental Techniques For 3D Computer Vision, A Tutorial for Entry Level Researcher, Tutorial 3D Reconstruction. v1.0*, Gadjah Mada University. Indonesia. 2011.
- [9] Gibson, J.J., *The Perception of the Visual World*, Houghton Mifflin, pages 235. Cambridge. Massachusetts. USA. 1950.
- [10] Martins Pedro. Batista Jorge, *Accurate Single View Model-Based head Pose Estimation*, 8<sup>th</sup> IEEE International Conference on Automatic Face and Gesture Recognition. pages 1-6. 2008.
- [11] D. Oberkampf, D. F. DeMenthon, Larry S. Davis. *Iterative Pose Estimation using Coplanar Feature Points*. Computer Vision and Image Understanding. vol. 63 no. 3. pages 495-511. May 1996.