

# PENGENALAN WAJAH INDIVIDU BERBASIS 3D BIOMETRIK

Harits Gymnovriza<sup>1</sup>, Ledy Novamizanti<sup>2</sup>, dan Eko Susatio<sup>3</sup>

<sup>1,2,3</sup>Teknik Telekomunikasi, Universitas Telkom, Bandung

Email: harits477@gmail.com <sup>1</sup> ledyaldn@telkomuniversity.ac.id<sup>2</sup>, maharusdi@gmail.com<sup>3</sup>

## Abstrak

*Sistem aplikasi pengenalan individu yang menggunakan media wajah 3D cukup menarik perhatian peneliti. Wajah merupakan identitas yang khas dan unik dari masing individu. Wajah dapat diolah sebagai citra berbasis 2D dan 3D. Pada penelitian ini, citra 3D didapatkan dengan menggunakan kamera Kinect v2. Jumlah pengambilan sebanyak 48 foto setiap individunya. Citra hasil akuisisi diproses dengan beberapa kali iterasi yang terpusat pada wajah individu. Selain itu juga dilakukan partisi terhadap citra wajah 3D menjadi 3 dan 6 bagian untuk mengetahui pengaruh partisi wajah terhadap tingkat akurasi. Metode ekstraksi berdasarkan pada konsep Iterative Closest Point (ICP) dan klasifikasi K-Nearest Neighbor (KNN). Hasil pengujian menunjukkan bahwa sistem pengenalan wajah 3D memberikan akurasi sebesar 88,09% pada percobaan iterasi 25, 6 partisi dan nilai K = 1.*

**Kata Kunci:** Wajah, Iterative Closest Point, KNN, Iterasi, Citra 3D, Kinect.

## Abstract

*Individual recognition application system that uses 3D face media is currently attracting many attention from researchers. The face is a unique and distinct identity of each individual. Faces can be processed as 2D and 3D based images. In this study, 3D images were obtained using the Kinect v2 camera. The number of shots taken is 48 pictures per individual. The acquisition image is processed with several iterations centered on the individual's face. Partitioning was also carried out on the 3D face image into 3 and 6 parts to determine the effect of the face partition on the level of accuracy. The extraction method is based on the concept of Iterative Closest Point (ICP) and K-Nearest Neighbor (KNN) classification. The test results show that the 3D face recognition system gives an accuracy of 88.09% in 25 iterations, 6 partitions, and K = 1.*

**KeyWords :** Face, Iterative Closest Point, KNN, Iteration, 3D image, Kinect

## I. PENDAHULUAN

Kehidupan manusia saat ini tidak bisa lepas dari sebuah teknologi. Hampir setiap saat selalu ditemukan penemuan baru yang dapat memberikan keuntungan untuk mempermudah aktivitas seseorang. Dibalik keuntungan pasti ada kerugian atau kekurangan. Hal ini yang dimanfaatkan oleh pihak – pihak tertentu sebagai sarana tindak kriminal. Misalnya, pencurian identitas pribadi karena mudahnya password untuk ditebak dan lemahnya pengamanan dari sistem tersebut [1]. Di antara banyaknya objek teknologi biometrik yang telah digunakan selama beberapa dekade terakhir, wajah adalah salah satu objek paling sering dipakai dalam proses pengenalan dan identifikasi individu [2]. Hal ini dikarenakan studi kasus tersebut dapat diimplementasikan dalam berbagai bidang [3]. Terdapat sedikit perbedaan diantara 2 basis tersebut yaitu *3D face recognition* dapat mengatasi beberapa keterbatasan yang ada pada pengenalan wajah 2D, contohnya seperti variasi pencahayaan dan pose kepala [4]. Berbagai metode juga telah dipakai untuk mendapatkan nilai akurasi yang lebih baik dalam penelitian tentang studi kasus tersebut.

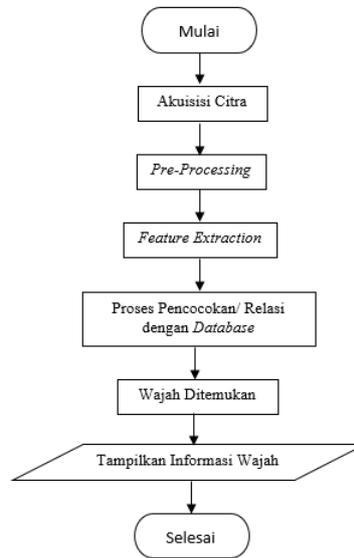
*Eigensurface on Kinect depth-maps* adalah salah satu metode yang sudah digunakan dalam penelitian *face recognition* [5]. Metode lain yang telah digunakan pada studi kasus *3D face recognition* adalah 3DLBP [6]. Pada tahun 2006, Chafik melakukan penelitian tentang pengenalan wajah 3D berdasarkan bentuk kurva wajah [7]. Performansi yang didapat mencapai akurasi sekitar 92% [7]. Penelitian terkait selanjutnya yang membahas *3D face recognition* adalah penelitian Zhang [8]. Penelitian tersebut menggunakan pendekatan *multiple keypoint descriptors* dan *sparse representation classifier* [8].

Di dalam pengolahan citra, KNN cukup banyak digunakan untuk membantu proses klasifikasi. Pada penelitian sebelumnya, mengenai studi kasus pengenalan ekspresi wajah secara 3D, pemanfaatan metode KNN memberikan tingkat akurasi yang tinggi yaitu sebesar 98,02% [9]. Penelitian selanjutnya mengenai penggunaan metode ICP dengan memanfaatkan *Gaussian Mix Model* (GMM) serta Database RMA 3D [10]. Penggunaan metode tersebut telah berhasil dan mendapatkan tingkat ketepatan pengenalan sebesar 97,33% [10]. Pada penelitian ini penulis menggunakan metode *Iterative Closest Point* sebagai ekstraksi ciri dan KNN *euclidean distance* sebagai klasifikasinya.

## II. METODE

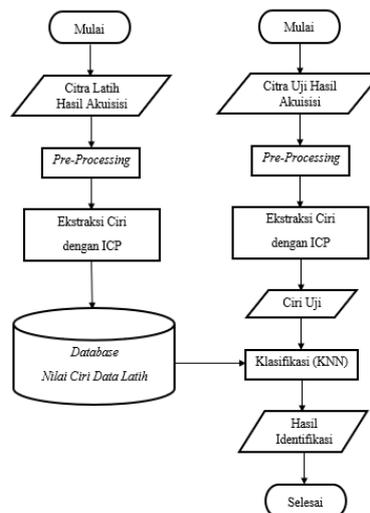
Bagian ini berisi uraian pembuatan atau perancangan sistem face recognition menggunakan Kinect camera v2. Selain itu, pada perancangan perangkat lunak akan dijelaskan pemrosesan citra wajah 3D. Input citra yang diproses oleh sistem adalah hasil akuisisi dari Kinect camera. Setelah itu citra masuk ke dalam tahap *pre-processing* dengan menggunakan *software KScan3D*. Metode ekstraksi ciri menggunakan metode *Iterative Closest Point*. Tahap terakhir adalah mengklasifikasi serta mencocokkan dengan database citra hasil ekstraksi menggunakan metode KNN. Gambar 1 merupakan diagram alir sistem pengenalan wajah

secara umum. Tujuan dibuatnya diagram blok sistem adalah untuk memudahkan pemahaman mengenai langkah demi langkah sistem dalam mengenali wajah seseorang.



Gambar 1: Diagram Alir Sistem Secara Umum

Adapun dilakukannya sebuah proses pelatihan pada sistem menggunakan citra latih. Proses ini bertujuan untuk mengumpulkan data sampel dari masing-masing wajah individu yang tersimpan di dalam database. Setiap citra yang digunakan pada pelatihan akan melalui tahap pre-processing. Hasil output dari sistem pelatihan digunakan sebagai referensi untuk proses uji atau pengenalan wajah.



Gambar 2: Diagram Alir Sistem Citra Latih dan Uji

#### A. Akuisisi Citra

Untuk mendapatkan citra wajah 3D pada tahapan akuisisi atau capture citra menggunakan media kamera Kinect versi 2 dan software KScan3D. Kamera Kinect berisi sensor kedalaman atau depth sensor, kamera warna, *array four-microphone* yang menyediakan pengambilan gerak dan rekonstruksi 3D wajah secara penuh [11]. *Depth map* adalah peta kedalaman hasil tangkapan dari Kinect camera [12]. Depth map merupakan bagian terpenting dari sebuah informasi yang terkandung dalam *3D geometry* [13]. Peta kedalaman direpresentasikan dalam citra keabuan, daerah yang gelap menunjukkan objek yang jauh. Sedangkan yang lebih terang merupakan daerah dengan jarak lebih dekat. Kscan3D merupakan *software* yang dapat membantu untuk menghubungkan kamera Kinect dengan PC. Dengan bantuan aplikasi ini maka akan mempermudah proses akuisisi, *cropping*, serta *build mask*, citra wajah 3D

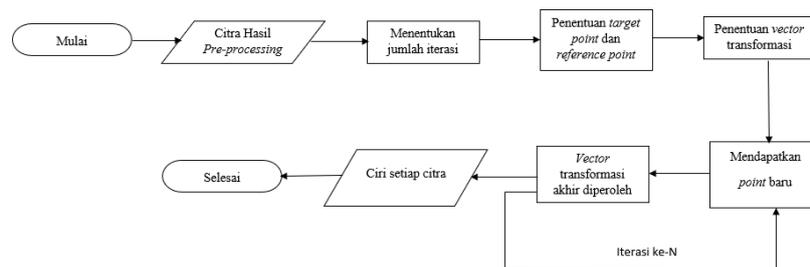
**B. Pre-Processing**

Tahap *pre-processing* dilakukan untuk tujuan mengoptimasi citra yang diperoleh pada tahap akuisisi. Dengan adanya tahapan ini, citra masukan diharapkan memiliki kualitas yang lebih baik sebelum dilakukan ekstraksi ciri. Citra latih dan uji keduanya melewati tahapan *Pre-Processing* guna perbaikan atau pengoptimalan informasi yang terkandung di dalamnya. Tanpa adanya *Pre-Processing* maka nilai ciri yang terkandung di dalam sebuah citra tidak bias didapatkan dengan baik. Alur tahapan *Pre-Processing* adalah sebagai berikut:

- 1) *3D Face Cropping*: Proses *cropping* dilakukan dengan menggunakan bantuan *software* KScan3D. Tahapan ini dilakukan manual dengan membuang bagian - bagian selain wajah.
- 2) *Build Mesh*: *Build mesh* bertujuan untuk mengkonversi *point cloud* menjadi *mesh geometry* sehingga citra 3D bisa untuk di *combine* dan di *finalize*.
- 3) *Combining & smoothing*: Tujuan dilakukannya *combining* dan *smoothing* adalah untuk menghaluskan permukaan yang disebabkan oleh *factor noise* dan menyatukan 16 hasil *take* foto menjadi 1 foto *mesh geometry*. Sehingga nantinya akan didapatkan citra 3D yang lebih *smooth*. Dengan menggunakan bantuan *software* KScan3D proses *smooth* akan dilakukan otomatis seiring proses *combining* yang sedang berjalan.
- 4) *Finalize Mask*: merupakan tahap terakhir dari *pre-processing* dimana output dari tahapan ini yang nantinya akan digunakan untuk proses ekstraksi ciri

**C. Iterative Closest Point (ICP)**

Pada tahapan ini, hasil *output* dari *pre-processing* diproses lebih lanjut. Pemilik citra wajah dikenali berdasarkan ekstraksi ciri menggunakan ICP. Dalam pengimplementasiannya di *software* matlab metode ICP dijalankan dengan menggunakan bantuan *source code toolbox ICP*. Untuk data input yaitu berupa data uji dan data latih yang masing masing terpartisi menjadi 3 bagian, yakni bagian wajah sebelah kanan, tengah, dan kiri. Tujuan dari ekstraksi ciri adalah untuk mendapatkan nilai ciri atau nilai parameter dari citra wajah 3D yang nantinya digunakan untuk proses klasifikasi, seperti terlihat pada Gambar 3.



Gambar 3: Diagram Alir Ekstraksi Ciri

*Iterative Closest Point (ICP)* dapat dinyatakan sebagai berikut. Diberikan 2 *data point clouds* A dan B, yang terdiri dari masing masing M serta N *points* dalam R3 [14]. Adapun sistem kerja dari ICP adalah dengan cara pencocokan terhadap *point cloud* sebagai pola tetap dan pola masukan pada citra 3D [15]. Metode ICP sering dipakai di dalam proses rekonstruksi citra 2D maupun 3D. Algoritma ICP merupakan algoritma pencocokan data 3D. Kunci utama dari algoritma ICP adalah corresponding point set registration. Selain itu algoritma tersebut juga untuk menguraikan matriks transformasi koordinat, serta mencari hubungan atau keterkaitan antar *point set*. Metode quaternion digunakan untuk menguraikan matriks transformasi diantara 2 *point set*. *Point set* tersebut dapat dibagi menjadi dua, untuk point target umumnya disebut P sedangkan *point reference* umumnya disebut dengan Q. Terdapat syarat - syarat untuk point set sesuai (*corresponding set point*), antara lain [16]:

- 1) Hitung jumlah P ( $N_p$ ) dan jumlah Q ( $N_q$ ), dimana  $N_p$  harus sama dengan  $N_q$  gelombang ( $\lambda$ ).
- 2) Untuk setiap *point* ( $p_i$ ) dari P harus sesuai dengan *point* ( $q_i$ ) dari Q.

Vektor rotasi seharusnya merupakan unit quaternion. Vector rotasi umumnya disimbolkan dengan  $q_r$ .

$$q_r = [q_0 \quad q_1 \quad q_2 \quad q_3]^T \tag{1}$$

dimana :  $q_0 \geq 0$  dan  $q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1$

Untuk mendapatkan matriks yang sesuai dengan 2 point set, jarak antara *corresponding point set* haruslah minimum. Solusi terakhir dari *coordinate transformation* adalah untuk menyelesaikan  $q$ , sehingga dapat dicari nilai minimum dari sebuah fungsi objektif.

$$f(q) = \frac{1}{n} \sum_{i=1}^n \|q_i - R(q_R) \cdot p_i - q_r\|^2 \quad (2)$$

Proses dari metode ICP antara lain:

- 1) Penentuan target point set ( $P$ ) dan reference point set ( $Q$ )
- 2) Hitung nilai tengah dari  $P$  dan  $Q$

$$\mu_p = \frac{1}{N_p} \sum_{i=1}^{N_p} P_i \quad (3)$$

$$\mu_q = \frac{1}{N_q} \sum_{i=1}^{N_q} Q_i \quad (4)$$

- 3) Bentuk matriks *covariance* dari  $P$  dan  $Q$

$$\sum_{P,Q} = \frac{1}{N_p} \sum_{i=1}^{N_p} [(p_i - \mu_p)(q_i - \mu_q)^T] = \frac{1}{N_p} \sum_{i=1}^{N_q} [p_i q_i^T] - \mu_p \mu_q^T \quad (5)$$

- 4) Bentuk matriks simetris 4x4 dari matriks *covariance*.

$$H(\sum_{P,Q}) = \begin{bmatrix} tr(\sum_{P,Q}) & \Delta^T \\ \Delta & \sum_{P,Q} + \sum_{P,Q}^T tr(\sum_{P,Q}) I_3 \end{bmatrix} \quad (6)$$

Di dalam formulasi matriks tersebut,  $I_3$  merupakan unit matriks  $3 \times 3$ ,  $tr(\sum_{P,Q})$  adalah *trace* dari  $\sum_{P,Q}$

- 5) Tentukan *eigenvalue* dan *eigenvector* dari  $H(\sum_{P,Q})$ , cari nilai *eigenvalue* dan *eigenvector* yang terbesar, *eigenvector* merupakan *vector* rotasi terbaik.

$$q_r = [q_0 \quad q_1 \quad q_2 \quad q_3]^T \quad (7)$$

- 6) Hitung vector translasi terbaik.

$$q_r = \mu_Q - R(q_R \mu_P) \quad (8)$$

- 7) Diperoleh koordinat vector transformasi final

$$q = [q_R | q_T] = [q_0 \quad q_1 \quad q_2 \quad q_3 \quad q_4 \quad q_5 \quad q_6]^T \quad (9)$$

#### D. K-Nearest Neighbor (KNN)

Pengelompokan data baru berdasarkan jarak data terhadap beberapa data tetangga (*neighbor*) merupakan konsep dasar dari metode klasifikasi KNN [17]. Ada beberapa rumus yang dapat digunakan untuk mencari dan mengukur jarak terdekat dari sebuah data pada metode KNN. Salah satunya adalah *euclidean distance*. Jarak *euclidean* dari 2 buah vektor  $x$  dan  $y$  dapat dihitung menggunakan rumus [18].

$$d(x, y) = \left( \sum_i (x_i - y_i)^2 \right)^{1/2} \quad (10)$$

Selain Euclidian distance, perhitungan jarak pada algoritma KNN juga dapat dihitung dengan rumus *city block*. Rumus *city block* dapat dilihat pada persamaan 11.

$$d(x, y) = L_p = i(x, y) = \sum_i^n |x_i - y_i| \quad (11)$$

Perhitungan *Euclidian distance* digunakan untuk menghitung jarak kemiripan antara citra latih dan citra uji. *Euclidian distance* merupakan selisih antara 2 buah vector yang akan dibandingkan sebagai pengenalan suatu objek yang akan diuji.

#### E. Akurasi Sistem

Akurasi merupakan suatu acuan yang terpenting di dalam penilaian kinerja sebuah sistem. Dengan adanya akurasi ketepatan suatu sistem dalam memproses sebuah masukan akan diukur, sehingga menghasilkan keluaran yang diinginkan.

$$Akurasi = \frac{\sum DataBenar}{\sum DataKeseluruhan} \times 100 \quad (12)$$

Confusion matriks berisi informasi tentang hasil identifikasi atau prediksi serta data aktual yang dilakukan oleh sistem klasifikasi [?], [20]. Tabel Confusion matriks dapat dilihat pada Tabel I.

Tabel I: Hasil Confusion Matriks

		Nilai Sebenarnya	
		TRUE	FALSE
Nilai Prediksi	TRUE	TP (True Positive)	FN (False Negative)
	FALSE	FN (False Negative)	TP (True Positive)

Keterangan:

TP (True Positive) : Data yang dideteksi benar.

FP (False Positive) : Data yang seharusnya dideteksi salah, tetapi oleh sistem dideteksi benar.

FN (False Negative): Data yang seharusnya dideteksi benar, namun oleh sistem dideteksi salah.

TN (True Negative) : Data yang dideteksi salah.

$$Recall(\%) = \frac{TP}{TP + FN} \times 100 \tag{13}$$

$$Precision(\%) = \frac{TP}{TP + FP} \times 100 \tag{14}$$

$$Accuracy(\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100 \tag{15}$$

*Recall* merupakan tingkat keberhasilan dari sebuah sistem untuk menemukan kembali informasi yang sudah didapatkan. Sedangkan *precision* merupakan tingkat keakuratan sistem untuk memperkirakan atau memprediksi informasi yang diminta oleh *user*. *Accuracy* merupakan nilai ketepatan atau keberhasilan dari suatu sistem dalam memprediksi keseluruhan informasi masukan.

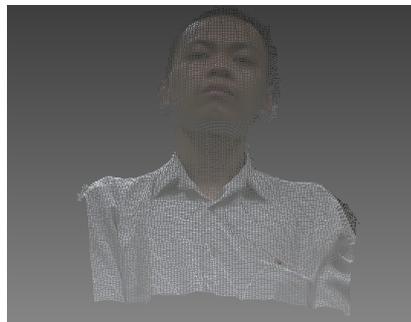
#### F. Waktu Komputasi

Waktu komputasi merupakan waktu yang diperlukan sistem dalam mengolah data masukan hingga didapatkan *output* yang diinginkan.

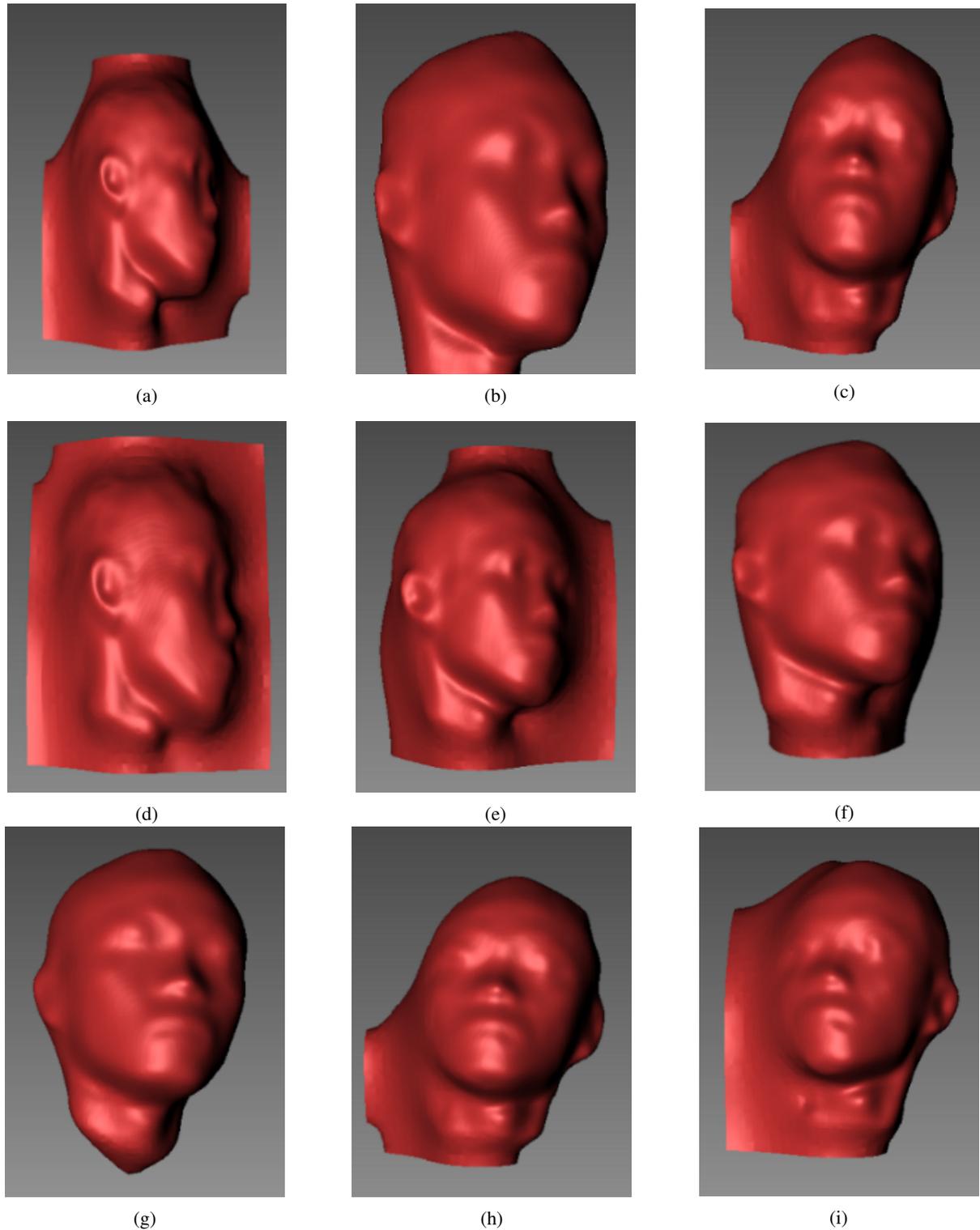
$$waktu\ komputasi = waktu\ selesai - waktu\ mulai \tag{16}$$

### III. HASIL

Dalam proses pengambilan citra posisi wajah tegak dan tanpa ekspresi. Setiap orang diambil sebanyak 5 buah citra 3D. Gambar 4 merupakan contoh hasil akuisisi menggunakan kamera kinect. Proses akuisisi dilakukan sebanyak 48 kali pengambilan untuk setiap citranya. Hasil capture sebanyak 48 buah akan dipartisi menjadi 3 dan 6 bagian sehingga nantinya akan pada tahap akhir terbentuk beberapa dalam format *polygon file* (\*.ply). Gambar 5 merupakan hasil akhir tahapan pre-processing.



Gambar 4: Contoh Hasil Akuisisi Kamera Kinect



Gambar 5: Contoh 3D face hasil Finalize partisi

#### IV. PEMBAHASAN

Setelah dilakukan pengimplementasian sistem pengenalan wajah individu berbasis 3D biometrik menggunakan kamera Kinect versi 2, maka tahapan selanjutnya adalah pengujian terhadap sistem. Tujuannya adalah untuk mengetahui angka keberhasilan atau akurasi dari kinerja sistem dalam mengenali wajah individu yang ada di dalam database.

*A. Pengaruh Jumlah Partisi Terhadap Performansi Sistem*

Berdasarkan bagian ini, dilakukan pengujian sistem menggunakan citra latih 3D. Citra latih berasal dari individu yang wajahnya diakuisisi menggunakan kamera *Kinect*. Adapun jumlah citra latih adalah 60 data, terdiri dari 12 individu yang didaftarkan di dalam *database*. Skenario pertama melakukan pengujian dengan menggunakan jumlah iterasi yang sama yaitu 5 dan nilai  $K = 1$ .

Tabel II: Hasil Pengujian Pengaruh Partisi

Partisi	Akurasi	Waktu Komputasi Rata-rata (detik)
3	70.00%	20.329
6	93.33%	37.80

Berdasarkan Tabel II dapat dilihat bahwa jumlah partisi citra wajah 3D dapat merubah atau mempengaruhi akurasi sistem. Dapat dianalogikan jumlah partisi berbanding lurus dengan akurasi sistem. Jadi, semakin banyak jumlah partisi akurasi pengujian sistem juga meningkat. Akurasi pengujian tertinggi didapat saat jumlah partisi 6 bagian yang mendapatkan angka 93,33%. Dengan demikian partisi yang lebih cocok digunakan adalah 6 bagian. Semakin banyaknya partisi atau point cloud pada dasarnya menambah waktu komputasi yang diperlukan, namun di sisi lain membuat sistem lebih mudah mengenali wajah seseorang dikarenakan pola wajah yang semakin terlihat dibuktikan dari akurasi yang didapatkan pada skenario pertama.

*B. Pengaruh Jumlah Iterasi Terhadap Performansi Sistem*

Pada bagian ini, dilakukan pengujian sistem berdasarkan skenario 2. Setelah didapatkan parameter terbaik dari skenario 1, kemudian dilakukan pengujian menggunakan perubahan jumlah iterasi. Data latih digunakan kembali pada skenario ini, yang bertujuan untuk mengetahui performansi sistem jika dilakukan pengujian terhadap partisi 6 nilai  $K = 1$ , dan jumlah iterasi yang berbeda.

Tabel III: Hasil Pengujian Pengaruh Jumlah Iterasi

Iterasi	Akurasi	Waktu Komputasi Rata-rata (detik)
5	93.33%	37.80
25	96.67%	438.543
49	93.33%	452.780

Berdasarkan Tabel III dapat dilihat bahwa jumlah iterasi ICP yang berbeda - beda dapat mempengaruhi akurasi sistem. Jumlah iterasi 25 merupakan iterasi yang menunjukkan angka akurasi tertinggi dibandingkan jumlah iterasi lainnya. Iterasi yang terlalu sedikit menyebabkan akurasi sistem kurang baik, hal ini dikarenakan antar wajah yang di partisi belum menemukan kesesuaian atau korespondensi satu sama lain. Sedangkan iterasi yang terlalu banyak menyebabkan partisi wajah dari individu yang berbeda semakin mirip. Namun, masih tetap memiliki ciri khasnya dibuktikan dengan akurasi yang tidak berbeda jauh dengan percobaan iterasi 25. Setiap metode klasifikasi, khususnya KNN pada hakikatnya membutuhkan nilai ciri yang unik dari berbagai citra. Hal ini dibutuhkan agar tidak terjadi suatu error atau kesalahan dalam melakukan pengenalan.

*C. Pengaruh Nilai K Terhadap Performansi Sistem*

Pada bagian ini, dilakukan pengujian sistem menggunakan skenario 3. Adapun cara pengujian skenario ini adalah dengan cara mengkombinasikan perbedaan jumlah nilai  $K$  pada euclidian distance dan city block terhadap parameter terbaik yang didapatkan pada pengujian skenario 1 dan 2. Skenario sebelumnya didapatkan parameter terbaik adalah partisi 6 bagian dan jumlah iterasi 25. Pengujian dilakukan terhadap data uji yang terdiri dari 12 individu terdaftar di dalam database dan 6 individu tidak terdaftar di dalam database. Individu yang terdaftar di dalam database memiliki 3 sampel uji citra wajah 3D, sedangkan individu tidak terdaftar di database hanya memiliki 1 sampel uji.

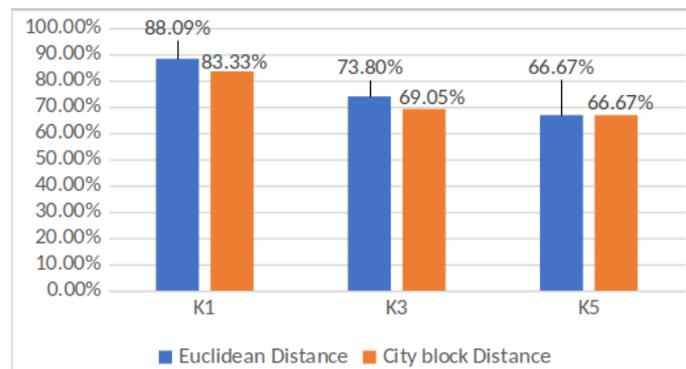
Tabel IV: Hasil Pengujian Pengaruh Jumlah Iterasi

Jenis Distance	Nilai K	Akurasi	Waktu Komputasi Rata-rata (detik)
euclidean distance	1	88.09%	438.5469
	3	73.80%	438.5474
	5	66.67%	438.5484
city block distance	1	83.33%	438.5464
	3	69.05%	438.5488
	5	66.67%	438.5529

Berdasarkan Tabel IV dapat diketahui bahwa variasi nilai K pada KNN *euclidian distance* juga mempengaruhi akurasi sistem. Akurasi tertinggi didapatkan pada percobaan dengan nilai tetangga terdekat 1. Semakin banyak peninjauan tetangga terdekat maka akurasi semakin menurun. Hal ini dikarenakan cukup banyak nilai ciri yang mirip antar individu setelah dilakukan perhitungan terhadap *euclidian distance*. Waktu komputasi rata - rata menunjukkan semakin banyak nilai K maka waktu yang dibutuhkan sedikit lebih lama. Setiap program dijalankan ulang waktu komputasi memang tidak sama persis, namun relatif mirip.

#### D. Pengaruh Jenis Jarak pada KNN Terhadap Performansi Sistem

Berdasarkan Gambar 6 dapat diketahui bahwa penggunaan rumus jarak Cityblock juga dapat mempengaruhi akurasi sistem. Sama halnya dengan Euclidean distance akurasi tertinggi didapatkan ketika nilai K =1. Semakin banyak nilai K maka akurasi semakin kecil.



Gambar 6: Grafik Perbandingan Pengaruh Jenis Jarak

Pada skenario terakhir, digunakan seluruh parameter terbaik yang telah didapatkan sebelumnya yaitu partisi 6 bagian, jumlah iterasi 25, nilai K = 1, dan jenis jarak *Euclidian distance* didapatkan akurasi tertinggi dengan data uji yaitu 88,09%.

## V. SIMPULAN DAN SARAN

### A. Simpulan

Setelah implementasi sistem dan dilakukan beberapa kali percobaan, beberapa kesimpulan pada penelitian ini, antara lain:

- 1) Tingkat akurasi tertinggi sistem dalam mengenali wajah seseorang berbasis citra wajah 3D terdapat pada percobaan dengan melakukan 25 iterasi, 6 partisi wajah dan nilai K =1 pada klasifikasi KNN dengan euclidean distance yang mencapai akurasi 88,09%.
- 2) Partisi wajah sebanyak 6 bagian dinilai lebih baik tingkat akurasinya dibandingkan dengan 3 partisi. Hanya saja membutuhkan waktu komputasi yang lebih lama dibandingkan dengan jumlah partisi wajah 3 bagian. Pada partisi 3 dan 6 dengan iterasi yang sama yaitu 5 menunjukkan tingkat akurasi pelatihan dan waktu komputasi yang berbeda. Pada partisi 3 iterasi 5 hanya mencapai akurasi sebesar 69,80% dan waktu komputasi 20,329 detik, sedangkan partisi 6 pada iterasi 5 dapat mencapai akurasi pelatihan sebesar 93,54% serta waktu komputasi yang lebih lama yaitu 37,80 detik.

### B. Saran

- 1) Metode cropping dan build mask langsung menggunakan software Matlab secara otomatis untuk meminimalkan waktu preprocessing.
- 2) Jumlah point cloud untuk setiap citra dapat dibatasi sehingga setiap citra memiliki keseragaman jumlah point cloud.

## PUSTAKA

- [1] Y.-W. Kao, H.-Z. Gu and S.-M. Yuan, "Personal based authentication by face recognition," in Fourth International Conference on Networked Computing and Advanced Information Management, Gyeongju, 2008.
- [2] B. B. Amor, et al, "3D Face recognition by ICP-based shape matching," in Lyon Research Center for Images and Intelligent Information Systems, Lyon, 2005.
- [3] S.Sumathi and R. Malini, "Face Recognition System to enhance E health," in 2010 International Conference on E-Health Networking, Digital Ecosystems and Technologies , Shenzhen, 2010.
- [4] Z. Guo and Y.-Y. Fan, "Sparse Representation for 3D Face Recognition," in 2013 Fourth World Congress on Software Engineering, Hong Kong, 2013.
- [5] J. B. C. Neto and A. N. Marana, "Face Recognition Using 3DLBP Method Applied to Depth Maps Obtained from Kinect Sensors," X Workshop de Vis'ao Computacional, pp. 168-172, 2014.
- [6] C. Samir, A. Srivastava and M. Daoudi, "3D Face Recognition Using Shape Of Facial Curves," in 2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings, Toulouse, 2006.

- [7] L. Zhang, "3DMKDSRC: A Novel Approach For 3D Face Recognition," in 2014 IEEE International Conference on Multimedia and Expo (ICME), Chengdu, 2014.
- [8] D. Han and Y. Ming, "Facial Expression Recognition With LBP and SLPP Combined Method," in 2014 12th International Conference on Signal Processing (ICSP), Hangzhou, 2014.
- [9] M. Romero, "Face recognition using Eigensurface on Kinect depth-maps," *IPCV'16*, pp. 241-247, 2016.
- [10] A. F. Abate, "2D and 3D face recognition: A survey," *Pattern Recognition Letters*, vol. XXVIII, no. 14, pp. 1885-1906, 2007.
- [11] Z. Zhang, "Microsoft Kinect Sensor and Its Effect," *IEEE MultiMedia*, vol. XIX, no. 2, pp. 4-10, 2012.
- [12] H. Lu, et al, "Depth Map Reconstruction for Underwater Kinect Camera Using Inpainting and Local Image Mode Filtering," *IEEE Access*, vol. V, pp. 7115 - 7122, 2017.
- [13] T.-W. Hui and K. N. Ngan, "Motion-Depth: RGB-D Depth Map Enhancement with Motion and Depth in Complement," in 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, 2014.
- [14] J. Cook, et al, "Face recognition from 3D data using Iterative Closest Point algorithm and Gaussian mixture models," in *Proceedings. 2nd International Symposium on 3D Data Processing, Visualization and Transmission*, Thessaloniki, 2004.
- [15] F. B. t. Haar and R. C. Veltkamp, "A 3D Face Matching Framework," in *IEEE International Conference on Shape Modeling and Applications*, Stony Brook, 2008.
- [16] H. Yang, et al, "An Improved Iterative Closest Points Algorithm," *World Journal of Engineering and Technology*, vol. III, pp. 302-308, 2015.
- [17] A. Rohman, "Model Algoritma K-Nearest Neighbor (K-NN) Untuk Prediksi Kelulusan Mahasiswa," *Neo Teknik*, vol. I, no. I, 2015.
- [18] D. Nugraheny, "Metode Nilai Jarak Guna Kesamaan Atau Kemiripan Ciri Suatu Citra (Kasus Deteksi Awan Cumulonimbus Menggunakan Principal Component Analysis)," *Angkasa*, vol. VII, no. 2, 2015.
- [19] A. Novitasari, E. P. Purwandari and F. F. Coastera, "Identifikasi Citra Daun Tanaman Jeruk Dengan Local Binary Pattern dan Moment Invariant," *Jurnal Informatika dan Komputer (JIKO)*, vol. III, no. 2, pp. 76-83, 2018.
- [20] P.N. Andono, T. Sutojo dan Muljono, "Pengolahan Citra Digital," Yogyakarta: Andi Offset, 2017.