

Implementasi *Fuzzy Logic* Identifikasi *Gesture* Tangan Untuk Deteksi Kerusakan

Verdika Septian Aryadi¹, Resty Wulanningrum², Patmi Kasih³

^{1,2,3}Teknik Informatika, Fakultas Teknik, Universitas Nusantara PGRI Kediri

E-mail: ¹jhonseptian99@gmail.com, ²resty0601@gmail.com, ³fatakasih@gmail.com

Abstrak – Maraknya kasus kerusakan pada lembaga masyarakat belakangan ini dipicu beberapa faktor. Antara lain kapasitas lapas, petugas lapas, dan lain sebagainya. Untuk meminimalisir kejadian yang sama (kerusakan) maka diperlukan adanya terobosan untuk menangani atau memberi tahu secara dini tentang kerusakan sehingga kerusakan bisa cepat ditangani. Terobosan yang dimaksud yaitu dengan memanfaatkan sistem pengenalan gerakan tangan (*hand Gesture*) dan kamera sebagai penangkap gambar serta metode *fuzzy* untuk mengenali gerakan tangan yang dapat menampilkan notifikasi adanya suatu tindakan kerusakan, dari notifikasi tersebut pengguna (*penjaga*) dapat mendapatkan informasi berupa peringatan kerusakan di dalam lapas. Sistem tersebut akan dijalankan menggunakan komputer. Untuk memastikan sistem tersebut dapat beroperasi dengan baik maka dilakukan beberapa skenario pengujian yaitu dengan jarak tangan terhadap kamera yang bervariasi serta menggunakan beberapa kamera dengan resolusi gambar yang berbeda. Maka diperoleh hasil yaitu sistem dapat mendeteksi gerakan tangan dengan stabil dan akurat apabila jarak input dengan kamera berada pada jarak lebih dari 50 cm, kondisi cahaya terang, kondisi background polos, dan menggunakan kamera beresolusi tinggi.

Kata Kunci — Deteksi Kerusakan, *Fuzzy*, *Hand Gesture*, Kamera

1. PENDAHULUAN

Di zaman sekarang ini kehidupan manusia tidak lepas dari kemajuan teknologi. Teknologi telah mempengaruhi masyarakat dan lingkungan disekitarnya. Semakin berkembangnya zaman, manusia juga membutuhkan interaksi dengan komputer yang menjadikan aspek yang penting dalam kehidupan sehari-hari. Pada saat ini Interaksi antar muka komputer personal telah berkembang dari *command line* berbasis teks ke bentuk grafis. Interaksi antar muka *command line* ialah “Perintah berupa teks yang diketik dengan tujuan untuk berinteraksi dengan sistem komputer” [1] mekanisme untuk menjalankan sistem operasi dengan mengetikkan code baris perintah yang dilakukan di terminal (*Command Prompt*) ini dirasa kurang efektif. Karena terdapat keterbatasan dalam berinteraksi dengan personal komputer. Oleh sebab itu, agar interaksi lebih mudah dan maksimal perlu dibuat sebuah sistem interaksi dan kontrol berbasis gestur tangan [2].

Gesture tangan sendiri memiliki arti pola gerak tangan yang bertujuan memberikan isyarat. Isyarat yang dimaksud adalah pikiran atau mengkomunikasikan perasaan kepada orang lain. Setiap individu dapat mengkomunikasikan berbagai perasaan dan pikiran menggunakan gerak tangan. Menurut Surya [3] (2009: 4) gerakan tangan adalah salah satu gerakan yang memiliki banyak variasi penggunaan dalam pembelajaran bahasa tubuh. Manusia menggerakkan tangan untuk mendapatkan *Gesture*. Pada setiap gestur yang dilakukan tentu memiliki maksud dan makna tersendiri. Informasi juga dapat dipahami dengan melihat gestur tangan dan kemudian mendefinisikan makna dari gerakan tangan. Karena hal tersebut penelitian ini ingin mencoba peluang metode pendekatan *vision* pada proses pengenalan citra digital gestur tangan manusia yang berisi informasi.

Pendekatan berbasis *vision* teknik yang efektif untuk mengembangkan sistem *interface computer* dengan manusia. Pengenalan gestur tangan bertujuan menciptakan sebuah sistem yang dapat mengidentifikasi secara spesifik gestur tangan manusia untuk menyampaikan informasi atau perangkat pengontrol. Karena tangan manusia adalah objek yang kompleks dengan banyak sendi dan hubungan antar komponen pembentuk tangan, penelitian tentang *Gesture* tangan masih terbuka. [4].

Terdapat penelitian sebelumnya dengan judul [5] “Pengenalan Gestur Jari Menggunakan Pengolahan Citra Untuk Mengendalikan *Joint* pada *Base Robot Lengan*” dimana pada penelitian tersebut mengimplementasikan kamera untuk mengenali gestur jari operator sehingga mampu memberikan perintah kepada robot untuk mengikuti gerakan jari membantu manusia dalam melakukan suatu kegiatan dan menggunakan *computer vision* sebagai pengenalan gestur tangan, algoritma *skin detector* akan diterapkan untuk mengenali tangan. Pada penelitian lain yang berjudul [6] “Pelacakan Gerakan Tangan Untuk Pengenalan Gerak-Isyarat” penelitian Perbedaan dengan penelitian yang dilakukan adalah, jika sebelumnya penelitian ini digunakan untuk melacak pergerakan tangan dalam sebuah rangkaian frame video. menggunakan 2 buah algoritma yakni *mean shift* dan *optical flow*.

Pada hasil penelitian diatas maka peneliti berfokus pengenalan gerakan tangan menggunakan metode *fuzzy* yang mana digunakan untuk melacak posisi gerakan dan konfigurasi jari untuk menentukan perintah yang harus dijalankan. Jari-jari akan dikenali oleh webcam yang mana jari-jari yang mengepal akan menampilkan notifikasi.

Nantintanya dari notifikasi tersebut pengguna dapat mendapatkan informasi berupa peringatan kerusakan di lembaga pemasyarakatan.

2. METODE PENELITIAN

2.1 *Gesture* Tangan

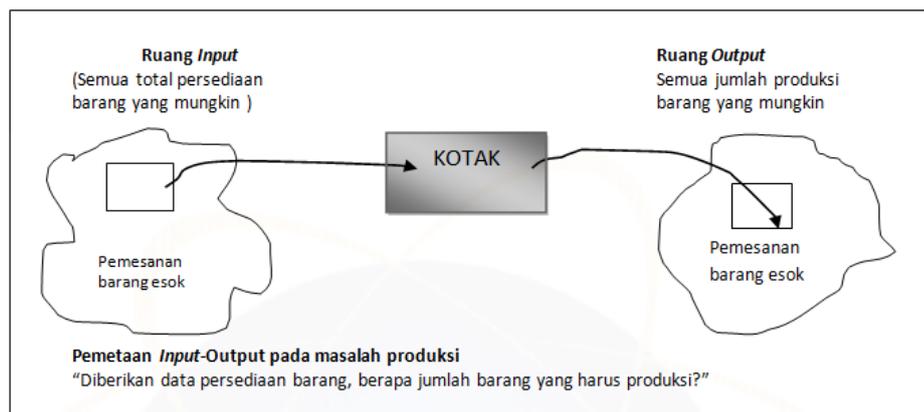
Menyampaikan suatu informasi kepada orang lain, dapat dilakukan dengan beberapa cara, salah satunya adalah dengan pergerakan fisik seperti *Gesture* (gerak isyarat). *Gesture* adalah pergerakan dari bagian tubuh antara lain; tangan, lengan, kaki, kepala dan wajah. Gestur dianggap sebagai salah satu cara interaksi yang lebih natural karena secara alamiah *gesture* biasanya digunakan untuk menyampaikan informasi. Dengan pengenalan *Gesture* manusia hanya perlu melakukan gerakan sederhana untuk mengoperasikan perangkat disekitarnya [7].

2.2 Pengolahan Citra

Pengolahan citra adalah setiap bentuk pengolahan sinyal dimana *input* adalah gambar, seperti foto atau video bingkai, sedangkan *output* dari pengolahan gambar dapat berupa gambar atau sejumlah karakteristik atau parameter yang berkaitan dengan gambar. Pengolahan citra merupakan proses pengolahan dan analisis citra yang banyak melibatkan persepsi visual. Proses ini mempunyai ciri data masukan dan informasi keluaran yang berbentuk citra. Istilah pengolahan citra digital secara umum didefinisikan sebagai pemrosesan citra dua dimensi dengan komputer. Dalam definisi yang lebih luas, pengolahan citra digital juga mencakup semua data dua dimensi. Citra digital adalah barisan bilangan nyata maupun kompleks yang diwakili oleh bit-bit tertentu [8].

2.3 Logika *Fuzzy*

Logika *Fuzzy* adalah suatu cara yang tepat memetakan suatu ruangan *input* ke dalam suatu ruang *output*[9]. Contoh pemetaan suatu *input-output* dalam bentuk grafis terlihat pada gambar 1.



Gambar 1. Bentuk Grafis

Logika *Fuzzy* dikatakan sebagai logika baru yang lama, sebab ilmu tentang logika *Fuzzy* modern dan metode baru ditemukan beberapa tahun lalu, namun sebenarnya konsep tentang logika *Fuzzy* telah ada pada diri manusia sejak lama. Logika *Fuzzy* merupakan bagian dari logika boolean, yang digunakan untuk menangani konsep derajat kebenaran, yaitu nilai kebenaran antara benar dan salah. Teori ini diperkenalkan oleh Dr. Lotfi Zadeh dari Berkeley pada tahun 1960-an, sebagai model ketidakpastian. Ada beberapa hal yang perlu diketahui dalam memahami sistem *Fuzzy*. Yaitu:

1. Variabel *Fuzzy*

Variabel *Fuzzy* merupakan variabel yang hendak dibahas dalam suatu sistem *Fuzzy* contoh umur, temperatur, permintaan dan sebagainya.

2. Fuzzifikasi

Fuzzifikasi adalah proses untuk mengubah variabel-variabel non *Fuzzy* (variabel numerik) menjadi variabel *Fuzzy* (variabel linguistik) nilai masukan-masukan yang masih dalam bentuk variabel numerik. Sebelum diolah oleh pengendali logika *Fuzzy* harus diubah terlebih dahulu kedalam variabel *Fuzzy*. Melalui fungsi keanggotaan yang telah disusun maka nilai-nilai masukan tersebut menjadi reformasi *Fuzzy* yang berguna nantinya untuk proses pengolahan.

2.4 *Python*

Python merupakan bahasa pemrograman tingkat tinggi yang sangat fleksibel. *Python* sangat cocok digunakan untuk pembuatan program yang membutuhkan waktu singkat, karena bahasa pemrograman *Python* dilengkapi dengan modul dan paket yang mempermudah dalam pembuatan *software*. *Python* dikembangkan oleh Guido Van

Rossum pada akhir 1989 dan dipublikasikan pada tahun 1991. *Python* dapat digunakan untuk berbagai keperluan pengembangan *software* dan dapat berjalan pada berbagai macam sistem operasi karena sifatnya yang *multiplatform*. Saat ini kode *Python* dapat dijalankan pada sistem berbasis: *Linux/Unix, Windows, Mac OS X, OS/2, Amiga, Palm, Symbian* (untuk produk- produk *Nokia*). Untuk menulis script program *Python* dapat menggunakan *word processing* seperti *notepad* pada *windows* atau *vim* pada *linux* dengan ketentuan ekstensi file yang dibuat adalah “.py”. Namun pada dasarnya *Python* telah dilengkapi dengan IDE yang digunakan untuk penulisan *script* program yang bernama *IDLE*. [10].

2.5 Open Source Computer (OpenCV)

Pada penelitian ini menggunakan sebuah *library Open Source Computer Vision*, yaitu *library open source multiplatform* berlisensi *BSD (Berkeley Software Distribution)* yang bersifat gratis untuk digunakan baik di kegiatan akademik maupun komersial. *Library* ini berfungsi untuk mentransformasikan data dari citra diam atau kamera video ke salah satu keputusan atau representasi baru. Semua representasi tersebut dilakukan untuk mencapai beberapa tujuan tertentu [11].

2.6 Webcam

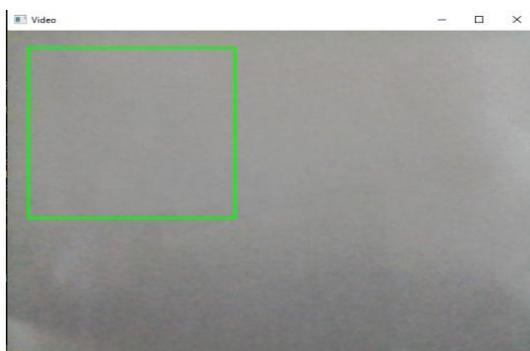
Webcam (singkatan dari *web camera*) adalah sebutan bagi kamera *real-time* (bermakna keadaan pada saat ini juga) yang gambarnya bisa diakses atau dilihat melalui *World Wide Web*, program *instant messaging* atau aplikasi *video call*. *Webcam* atau *web camera* adalah sebuah kamera video digital kecil yang dihubungkan ke komputer melalui (biasanya) *port USB* ataupun *port COM*. Sekarang ini *web camera* yang ada di pasaran pada umumnya terbagi ke dalam dua tipe: *web camera* permanen (*fixed*) dan *revolving web camera*. Pada *web camera* permanen terdapat pengapit untuk mengapit lensa standar di posisi yang diinginkan untuk menangkap gambar pengguna. Sedangkan pada *revolving web camera* terdapat landasan dan lensa standar dipasang di landasan tersebut sehingga dapat disesuaikan ke sudut pandang yang terbaik untuk menangkap gambar pengguna.

3. HASIL DAN PEMBAHASAN

Aplikasi ini dibuat dengan bahasa pemrograman *python* dengan metode *fuzzy* untuk identifikasi *Gesture* tangan. Setelah *Gesture* tangan terdeteksi maka algoritma pengenalan *Gesture* tangan dapat digunakan.



Gambar 2. Menu Utama



Gambar 3. Menu Deteksi

3.1 Tampilan Aplikasi

Aplikasi Yang Telah Dirancang diimplementasikan sehingga memberikan *output* berupa tampilan. Tampilan dibuat sederhana sehingga mudah digunakan dan tidak mengesampingkan fungsionalitasnya.

1. Menu Utama

Gambar 2 merupakan tampilan pertama kali saat aplikasi dijalankan. Terdapat Dua tombol yaitu menu deteksi dan menu keluar sistem.

2. Menu Deteksi

Gambar 3 merupakan tampilan menu deteksi terdapat kotak hijau untuk inputan dari citra kemudian akan menampilkan pesan teks notifikasi peringatan berupa teks di bawah kotak hijau.

3.2 Pengujian

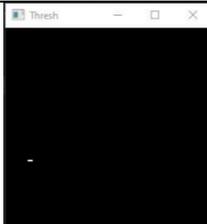
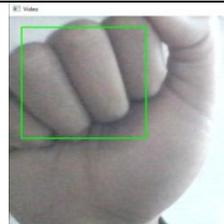
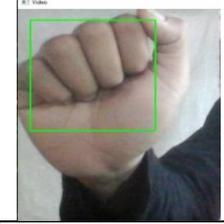
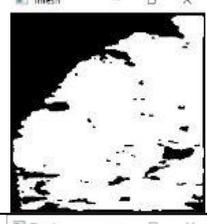
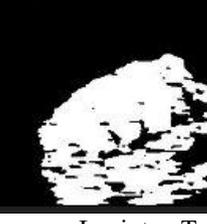
Bagian ini membahas hasil pengujian aplikasi yang telah dibuat dengan menggunakan data yang diambil secara langsung menggunakan kamera laptop. Untuk data *testing* dilakukan langsung terhadap *user (realtime)*. Untuk melakukan pengujian terdapat beberapa skenario yaitu:

1. Skenario Pertama

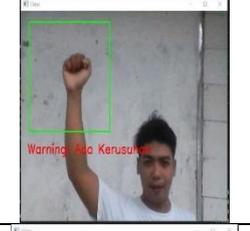
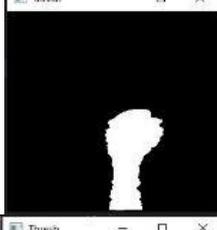
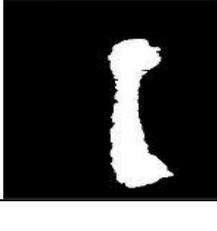
Pengujian dilakukan dengan tangan mengepal menghadap kamera dimana jarak antara tangan dan kamera laptop bervariasi mulai dari yang dekat 10 cm sampai yang paling jauh 100 cm dengan pencahayaan yang cukup serta menggunakan background polos. Maka diperoleh hasil seperti tabel 1.

Pada skenario uji coba ini terdapat beberapa hasil menunjukkan deteksi gagal atau tidak stabil dan tidak akurat sesuai output hal ini disebabkan karena jarak objek tangan pengguna terlalu dekat dengan webcam yaitu jarak deteksi antara input citra dengan kamera kurang dari 50 cm sedangkan hasil menunjukkan pengujian terdeteksi sempurna apabila input citra berada pada jarak lebih dari 50cm.

Tabel 1. Tangan mengepal menghadap kamera

Pengujian ke-	Jarak (Cm)	Threshold	Hasil deteksi	Keterangan
1	10 cm			Gagal
2	20 cm			Gagal
3	30 cm			Gagal
4	40 cm			Gagal
5	50 cm			Gagal

Lanjutan Tabel 1.

Pengujian ke-	Jarak (Cm)	Threshold	Hasil deteksi	Keterangan
6	60 cm			Terdeteksi
7	70 cm			Terdeteksi
8.	80 cm			Terdeteksi
9.	90 cm			Terdeteksi
10.	100 cm			Terdeteksi

2. Skenario kedua

Pada skenario kedua ini menggunakan dua kamera yaitu kamera *on board* laptop Asus tipe X441N dan *webcam logitech* type c52 dengan jarak sama dan pencahayaan yang cukup serta menggunakan background polos. Dimana pengujian kedua ini ditujukan untuk melihat apakah spesifikasi kamera dapat mempengaruhi hasil deteksi. Maka hasil pengujian skenario kedua ini dapat dilihat pada tabel dibawah ini.

Pada skenario pengujian ini, deteksi *Gesture* tangan berjalan stabil dan akurat sesuai output pada semua *Gesture* tangan (Tabel 2) , hal ini disebabkan karena jarak, dan resolusi kamera *webcam* tercukupi sehingga *thresholding* tangan dapat dibentuk dengan sempurna dan terbedakan dengan background.

Pada skenario pengujian ini, deteksi *Gesture* tangan ada beberapa berjalan tidak stabil dan tidak akurat sesuai *output*, hal ini disebabkan karena kurangnya resolusi kamera laptop sehingga *thresholding* tangan tidak dapat dibentuk dengan sempurna.

4. SIMPULAN

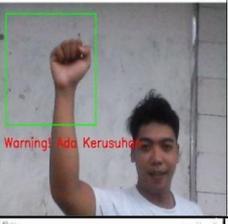
1. Sistem didesain dengan tampilan sederhana untuk metode pengenalan peringatan kerusakan di lapas.
2. Sistem dapat menerjemahkan *Gesture* tangan dari pengguna menjadi notifikasi yang sesuai dengan output.
3. Proses mendeteksi *Gesture* tangan dapat dilakukan oleh webcam yang terintegrasi melalui laptop pengguna sehingga dapat dimaksimalkan fungsinya, namun webcam tetap memiliki kelemahan pada bagian resolusi dan sensor.

Tabel 2. Hasil dengan kamera webcam

Pengujian	Threshold	Hasil deteksi	Keterangan
-----------	-----------	---------------	------------

ke-			
1			Terdeteksi
2			Terdeteksi
3			Terdeteksi
4			Terdeteksi

Tabel 3. Hasil dengan kamera laptop

Pengujian ke-	Threshold	Hasil deteksi	Keterangan
1			Tidak Stabil
2			Tidak Stabil
3			Terdeteksi

Lanjutan Tabel 3.

Pengujian ke-	Threshold	Hasil deteksi	Keterangan
4			Terdeteksi

5. SARAN

1. *Output* dari pengenalan *Gesture* tangan pengguna sebaiknya dapat divariasikan ke bentuk bahasa lainnya
2. Media pendeteksi *Gesture* tangan agar dapat divariasikan ke kamera *input* lainnya yang memiliki resolusi dan sensor yang lebih baik daripada webcam.
3. Pada penggunaan metode atau algoritma, sebaiknya perlu ditambahkan atau diperbarui sehingga dalam mendeteksi isyarat tangan pengguna dapat dilakukan lebih mudah, akurat dan stabil.

DAFTAR PUSTAKA

- [1] IKAPI. 2009. <http://adaadalah.blogspot.com/2015/12/pengertian-command-line-adalah.html>
- [2] Yunita. 2019. "Hand Gesture Recognition Sebagai Pengganti Mouse Komputer Menggunakan Kamera" *Jurnal Eltikom* vol. 3. hal 64-76.
- [3] Surya. 2009. <https://www.semanticscholar.org/paper/Secara-umum%2C-gerakan-tangan-adalah-salah-satu-yang-Tangan-Isyarat/dd4b97cbc34a5af5ec409d4091643c20223aa1c2>.
- [4] Seni K. 2014. *Perceptual Human Computer Interaction/Interface Menggunakan Teknik Computer Vision*.
- [5] Chaidir, A. R., Muldayani, W., & Kalandro, G. D. 2018. "Pengenalan gestur jari menggunakan pengolahan citra untuk mengendalikan joint pada base robot lengan." *Jurnal Rekayasa Elektrika*, vol 14. hal 145-220.
- [6] Jawas, N. 2018. "Pelacakan Gerakan Tangan Untuk Pengenalan Gerak-Isyarat". *It (Informatic Technique) Journal*, vol 5. hal 2252-746X
- [7] Puturuhi, J. P. 2021. "Deteksi hand Gestures pada ground marshall menggunakan metode hmm dalam proses parkir pesawat (Doctoral dissertation, Institut Teknologi Nasional).
- [8] Pramudana, Y., & Wirayuda, T. A. B. 2015. Pengenalan Tulisan Tangan Menggunakan Algoritma Diagonal Feature Extraction Dan K-nn.
- [9] Kusumadewi, Sri. Purnomo, Hari. 2004. Aplikasi Logika Fuzzy untuk pendukung keputusan Edisi Pertama. Yogyakarta: Graha Ilmu.
- [10] Richardson, M dan S. Wallace. 2013. *Getting Started With Raspberry Pi*. O'Reilly Media, Inc., USA
- [11] Sajati, H., Indrianingsih, Y., & Wulan, P. I. D. C. 2016. Deteksi jerawat pada wajah menggunakan metode viola jones. *Compiler*, vol 5.