

KENDALI *MOBILE ROBOT* BERBASIS POSE TANGAN MENGGUNAKAN *CONVOLUTIONAL NEURAL NETWORK* (CNN)

1st Alfian Miftah Arzaqi

Departemen Teknik Komputer
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
alfan.19072@mhs.its.ac.id

2nd Ahmad Zaini, S.T., M.Sc.

Departemen Teknik Komputer
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
zaini@te.its.ac.id

3rd Dr. Eko Mulyanto Yuniarno, S.T., M.T.

Departemen Teknik Komputer
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
ekomulyanto@ee.its.ac.id

Abstract—Perkembangan teknologi *Human Computer Interaction* atau HCI yang memungkinkan untuk berkomunikasi dengan komputer menggunakan pose tangan. dengan majunya teknologi suatu *remote control* pada robot mulai dikembangkan dengan mengutamakan interaksi pengguna dengan robot. Sistem kendali pada robot sudah mulai meninggalkan *remote control joystick* dan beralih pada kendali menggunakan gestur tangan atau secara *autonomous*. Penggunaan *remote control joystick* mulai ditinggalkan karena terdapat beberapa kekurangan dalam mengendalikan robot. Dalam penelitian ini dibuat kendali robot berbasis pose tangan menggunakan *Convolutional Neural Network*. Pada penelitian ini *remote control joystick* digantikan dengan kamera sebagai input pose tangan dari user. Citra yang ditangkap oleh kamera diklasifikasikan menggunakan *Convolutional Neural Network* untuk menerjemahkan perintah dari pengguna kepada robot. Hasil klasifikasi diubah menjadi kode instruksi yang selanjutnya dikirimkan kepada robot menggunakan internet untuk diterjemahkan menjadi perintah navigasi oleh *mikrokontroler*. Nilai akurasi yang didapatkan dari hasil *training* menggunakan CNN sebesar 0.95 dan memiliki jarak ideal pada 50cm.

Index Terms—*Convolutional Neural Network, Mobile Robot, Pose,*

I. PENDAHULUAN

Pose adalah potongan-potongan dari gerakan yang bertujuan untuk komunikasi non verbal yang digunakan untuk menyampaikan pesan penting, baik secara langsung maupun tidak langsung tanpa menggunakan kata-kata. Setiap pose dapat memiliki arti tersendiri sesuai dengan kesepakatan umum ataupun personal yang melakukan komunikasi. Pose tangan berarti suatu sikap yang diberikan oleh tangan untuk berkomunikasi [1].

Pose tangan ini biasa digunakan oleh penderita tuna rungu untuk berkomunikasi, namun dengan perkembangan teknologi dimana adanya teknologi *human computer interaction* yang memungkinkan untuk berkomunikasi dengan komputer menggunakan pose tangan. *Human Computer Interaction* atau biasa disingkat HCI adalah suatu bidang studi yang merancang teknologi komputer untuk lebih interaktif saat dipakai. Adanya HCI ini dapat memudahkan manusia untuk menyelesaikan tugas-tugasnya dengan bantuan komputer [2].

Pada saat ini perkembangan HCI cukup pesat yang diakibatkan dari perkembangan pada teknologi *machine learning* dan robotika. Teknologi *machine learning* memungkinkan komputer untuk belajar dari data-data yang diberikan oleh manusia dan juga mengambil keputusan dari data-data tersebut. Pada salah satu cabang *machine learning* yaitu *computer vision* memungkinkan untuk komputer mengerti gerakan atau pose yang diberikan manusia melalui gambar atau video [3]. Tidak hanya *machine learning* yang berkembang namun pada robotika juga berkembang terutama pada sistem kendalinya. Saat ini robot dapat dikendalikan tidak hanya menggunakan *joystick*, namun juga dapat dikendalikan dengan gestur tangan dan juga secara *autonomous*, namun saat ini kendali menggunakan *joystick* secara perlahan mulai ditinggalkan karena kurang memberikan pengalaman yang interaktif saat digunakan [4].

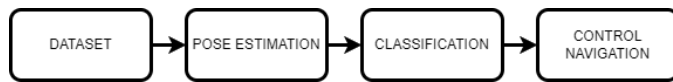
Terdapat penelitian yang dilakukan oleh Fauzia [5] mengenai kendali robot menggunakan gestur tangan dengan bantuan alat yang ditempelkan pada punggung tangan. Kekurangan penelitian tersebut masih menggunakan alat yang ditempelkan pada tangan untuk mendapatkan gestur tangan. Kendali robot menggunakan gestur tangan mulai dikembangkan menggunakan sensor yang diletakkan pada tangan dan juga dapat menggunakan *computer vision* untuk mengetahui gestur tubuh manusia. Terdapat penelitian penggunaan kamera untuk kendali dari *mobile robot* yang dilakukan oleh Kukuh [6] mengenai pengendalian robot menggunakan tanda panah yang terdeteksi. Kekurangan dari penelitian ini untuk kendali menggunakan gambar panah yang diletakkan pada depan robot untuk dapat menggerakkan robot.

Pada penelitian ini diusulkan penelitian kendali *mobile robot* berbasis pose tangan menggunakan *Convolutional Neural Network* (CNN). CNN dipilih dikarenakan dapat mengolah data citra yang merupakan data 2 dimensi [7].

II. METODOLOGI

Tugas akhir ini merupakan penelitian pada bidang visi komputer yang bertujuan untuk mengendalikan robot menggunakan metode *classification* CNN. Secara umum metode yang

digunakan adalah pertama pembuatan dataset, prediksi pose tangan, *classification* menggunakan model CNN, dan terakhir navigasi control pada robot. Alur metodologi dapat dilihat pada



Gambar 1. Blok diagram metodologi

A. Dataset

Tahapan pengambilan dataset dilakukan dengan pengambilan data-data berupa citra menggunakan kamera pada laptop. Kamera pada laptop dipakai untuk mendapatkan citra yang nantinya digunakan untuk mendeteksi suatu pose atau gestur tangan membutuhkan citra tangan untuk menemukan bagian tangan. Proses pengumpulan citra pose tangan yang digunakan sebagai navigasi robot dimulai dengan menentukan jumlah kelas dataset. Kelas dataset berjumlah enam pose yang terdiri dari perintah maju, mundur, diam, belok kiri, belok kanan, dan tembak. Setiap pose membutuhkan dua tangan untuk memberikan perintah dengan telapak tangan dihadapkan ke arah kamera. Pose untuk perintah maju dilambangkan dengan kedua tangan mengepal. Pose mundur dilambangkan dengan meluruskan jari manis dan jari tengah seperti halnya pose angka dua pada SIBI pada kedua tangan dan telapak tangan. Pose diam dilambangkan dengan membuka telapak tangan atau meluruskan semua jari pada kedua tangan. Pose belok kiri dilambangkan dengan membuka telapak tangan pada tangan kiri dan mengepalkan tangan pada tangan kanan. Pose belok kanan merupakan kebalikan dari pose belok kiri yang dilambangkan dengan tangan kiri mengepal dan tangan kanan membuka telapak tangan. Pose tembak dilambangkan dengan meluruskan ibu jari dan jari manis sedangkan jari lain dikepalkan dengan jari manis diarahkan ke atas dan telapak tangan menghadap ke kamera.

B. Pose Estimation

Tangan yang terangkap kamera terdeteksi dengan bantuan framework Mediapipe. Framework Mediapipe ini memiliki 21 titik *keypoint* pada tangan. Proses mendeteksi tangan dimulai dengan cara mencari telapak tangan yang terdapat titik nol *keypoint* yang disebut sebagai palm detection model. Setelah titik nol terdeteksi atau telapak tangan ditemukan kemudian mencari 20 titik lainnya pada tangan. Titik-titik *keypoint* ini dicari menggunakan regresi. Setelah 21 titik *keypoint* terdeteksi selanjutnya menyimpan koordinat x dan y dari yang relatif terhadap layar yang serta menghubungkan 21 titik tersebut sehingga membentuk *hand landmark*. Rangka tiap tangan yang terdeteksi memiliki nilai-nilai koordinat pada tiap *keypoint*, dari koordinat ini dicari nilai terkecil dan nilai terbesar. Nilai-nilai ini nantinya digunakan untuk membuat bounding box pada setiap frame yang terdeteksi tangan didalamnya. Setiap frame memiliki dua bounding box dikarenakan harus terdapat dua tangan untuk bisa menavigasi robot. Bounding box ini

digunakan untuk melakukan hand lokalisasi yang bertujuan untuk menormalisasi variasi posisi dari *hand landmark* yang telah dibuat oleh Mediapipe. Hand lokalisasi ini nantinya membuang bagian citra diluar bounding box dan menyatukan kedua tangan setelahnya. Hasil dari hand lokalisasi nantinya disimpan pada suatu folder pada setiap class. Setelah seluruh data pada setiap class terpenuhi maka dilanjutkan tahapan selanjutnya yaitu *training* dan *classification*.

C. Classification

Sebelum dilakukannya *classification* dilakukan training data terlebih dahulu. Training data merupakan tahapan dari dataset yang telah dibuat yang kemudian dilatih untuk memperoleh suatu pola dari data tersebut. Pola tersebut digunakan untuk membantu mengenali objek dalam citra. Dalam proses training dibutuhkan konfigurasi yang digunakan. Konfigurasi training yang digunakan yaitu CNN dengan *convolution layer*, *max pooling layer*, dan *fully connected Layer*. Setelah dilakukan proses *validation* dan model yang telah dibuat memiliki akurasi yang tinggi (sesuai dengan *threshold*), maka tahapan selanjutnya yaitu *classification*. Tahap *classification* dilakukan testing menggunakan citra baru lalu dideteksi pose dalam citra tersebut. Citra yang posenya telah diketahui selanjutnya diterjemahkan menjadi kode instruksi yang akan menjadi acuan dalam memberikan perintah kepada robot.

D. Control Navigation

Tahapan ini berupa cara mengontrol robot menggunakan kode intruksi dari hasil *classification*. Laptop dengan robot terhubung secara *wireless* menggunakan koneksi internet. Hasil *classification* dari laptop dikirimkan kepada robot menggunakan jaringan web socket, dimana laptop dan robot terhubung dalam suatu port dan ip yang telah ditentukan. Robot dapat dikontrol sesuai dari banyaknya *class* pada model yang telah dibuat dalam hal ini terdapat enam *class*. Aksi dari robot ini sesuai dari data yang diterima dari hasil *classification* yakni maju, mundur, kanan, kiri, diam, dan tembak. Tiap-tiap aksi diwakilkan dengan huruf abjad tertentu. Sesuai dengan aksi-aksi tersebut maka robot Pergerakan robot dapat terjadi dikarenakan adanya sepasang roda yang dipasangkan pada robot. Perputaran roda diatur dari motor driver dengan menyalurkan listrik kepada motor dc yang sudah dipasangkan roda sebagai outputnya. Kemampuan robot untuk tembak menembak menggunakan bantuan *infrared*. Tiap robot nantinya memiliki *infrared transmitter* dan *infrared receiver* yang diletakkan disekeliling robot. Aksi menembak dimulai dari *infrared transmitter* mengirimkan pesan dan nantinya diterima oleh *infrared receiver*.

III. PENGUJIAN DAN PEMBAHASAN

Dalam bab ini dijelaskan mengenai hasil dan pembahasan dari pelaksanaan penelitian, dan juga proses evaluasi pengujian. Melalui pembahasan serta pengujian mengenai hasil penelitian, diharapkan dapat menjadi acuan dalam pengambilan kesimpulan dari pelaksanaan tugas akhir ini.

A. Hasil Dataset

Proses pengumpulan dataset dilakukan dengan pengambilan citra menggunakan kamera. Proses tersebut dilakukan selama beberapa waktu untuk dapat mendapatkan beberapa frame citra untuk suatu class dan diulang kembali dengan class yang berbeda sampai seluruh class terpenuhi. Seluruh citra tangan diambil dari satu orang yang sama dan menggunakan jarak pengambilan yang sama, yaitu jarak antara kamera dan tangan. Dalam proses pengambilan citra dilakukan beberapa variasi dalam pose tangan yaitu dengan sedikit merotasikan tangan dan sedikit melemaskan tangan. Hal ini dilakukan agar tiap class memiliki variasi pose dan dapat mendeteksi dari berbagai kondisi pose. Adapun pose yang digunakan sebagai dataset dapat dilihat pada Tabel I.

Tabel I
CITRA HASIL DATASET

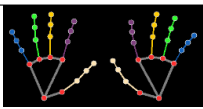
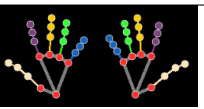
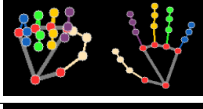
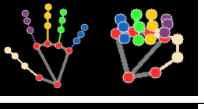
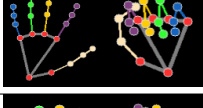
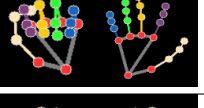
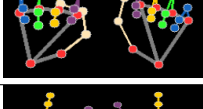
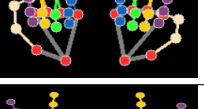
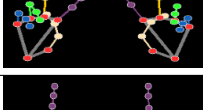
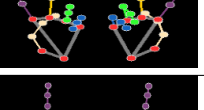
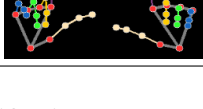
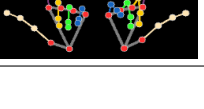
Pose	Citra
Diam	
Kanan	
Kiri	
Maju	
Mundur	
Tembak	

B. Hasil Pose Estimation

Proses pose prediction dilakukan dengan menggambar hand landmark yang telah terdeteksi dengan menggunakan Mediapipe pada tiap-tiap citra yang telah dikumpulkan pada tahap sebelumnya. Setelah hand landmark tergambar maka tiap-tiap hand landmark digambar pada citra berlatar gelap. Hand

landmark yang terdapat pada citra ini diberikan bounding box pada tiap tangan. Setelah bounding box terbuat maka selanjutnya dilakukan hand localization pada hand landmark yang berada pada citra hitam. Hand localization ini untuk mengecilkan ukuran citra menjadi hanya selebar bounding box dengan ditambah 20 pixel pada nilai maksimal dan dikurangi 20 pixel pada nilai minimum pada tiap bounding box. Penambahan pixel ini dimaksudkan untuk mencakup semua hand landmark yang telah dibuat Mediapipe. Dengan mengurangi ukuran citra ini bertujuan mengurangi waktu training dan dapat meningkatkan performa. Citra diluar bounding box dibuang dan tiap bounding box dirubah ukurannya menjadi 128x128 pixel untuk menyamakan kedua bounding box dalam satu frame citra. Setelah setiap ukuran bounding box sama maka setiap bounding box ditumpuk secara horizontal. Citra yang disimpan dan dijadikan dataset adalah citra yang terdapat dua hand landmark yang telah ditumpuk secara horizontal. Citra yang disimpan nantinya memiliki ukuran 256x128 pixel dikarenakan hasil dari dua tangan yang terdeteksi. Pada tahap menumpuk citra terdapat dua macam tumpukan yaitu citra dengan posisi tangan kiri berada pada kotak sebelah kiri dan tangan kanan pada kotak sebelah kanan. Posisi kedua yaitu tangan kiri berada pada posisi kanan dan tangan kanan berada pada kiri yang selanjutnya disebut sebagai posisi invers. Posisi hand landmark dari hasil hand localization dapat dilihat pada Tabel II.

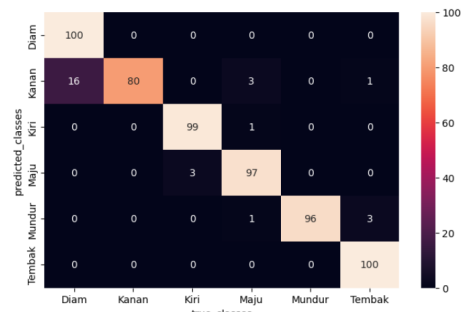
Tabel II
HASIL POSISI HAND LOCALIZATION

Pose	Pose Benar	Pose Invers
Diam		
Kanan		
Kiri		
Maju		
Mundur		
Tembak		

C. Hasil Classification

Proses *classification* dimulai dari *training* dataset. Sebelum melakukan *training* maka dibutuhkannya data citra pada tiap class. Tiap class memiliki *train* data dan *validation* data. Data-

data ini diambil dari proses sebelumnya yaitu citra yang telah melewati tahap *hand localization*. Citra untuk train data pada tiap class berjumlah 400 dengan rincian 200 citra posisi benar dan 200 citra posisi invers, sedangkan citra untuk *validation* data berjumlah 80 citra dengan 40 citra posisi benar dan 40 citra posisi invers. Adapun tabel pesebaran data citra tiap class dapat dilihat pada Tabel III. *Training* ini menggunakan CNN dengan menggunakan satu layer convolution ukuran 3x3 dengan filter 16, sehingga ukuran citra yang awalnya 256x128 turun menjadi 254x126 yang disebabkan dari adanya konvolusi. Selanjutnya dilakukan pooling dengan menggunakan max pooling ukuran 2x2 yang menyebabkan ukuran citra turun menjadi setengah dari hasil konvolusi yaitu 167x63. Layer selanjutnya yaitu flatten dimana citra hasil dari max pooling di ubah menjadi vektor. Output yang didapatkan dari layer flatten yaitu 128016 vektor. Setelah proses konfigurasi layer CNN dan hyperparameter telah dilakukan. Selanjutnya yaitu proses *training* yaitu berupa model dan detail data pada tiap proses pengulangan. Hasil dari *training* ini didapatkan *training accuracy*, *validation accuracy*, *training loss*, dan *validation loss*. Setiap perulangan mendapatkan nilai tersebut, grafik dari nilai tersebut dalam 20 perulangan dapat dilihat pada Gambar 2. Hasil dari *training* didapatkan Evaluasi model menggunakan confusion matrix dengan data testing citra sebanyak 100 pada masing-masing class. Hasil dari confusion matrix didapatkan akurasi 0.95, hasil tersebut secara detail tiap class dapat dilihat pada Gambar 3.



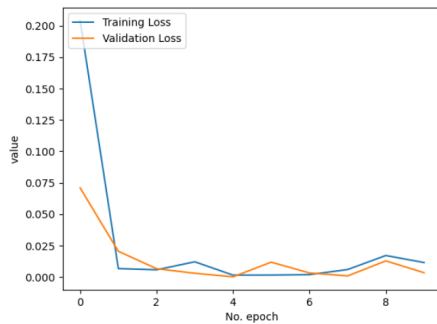
Gambar 3. Hasil *Confusion Matrix*

Tabel III
JUMLAH CITRA TIAP Class

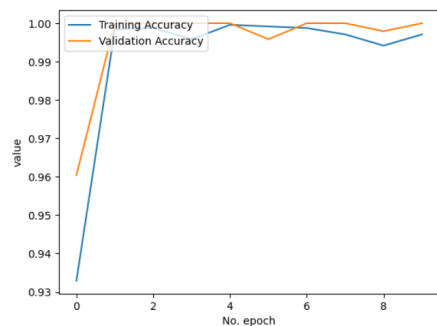
Class	Train Data	Validation Data
Diam	400	80
Kanan	400	80
Kiri	400	80
Maju	400	80
Mundur	400	80
Tembak	400	80

D. Hasil Control Navigation

Proses *control navigation* diawali dengan perancangan pada robot. Robot dirancang dengan memiliki kemampuan untuk bergerak dan menembak. Pergerakan robot dapat terjadi dikarenakan adanya sepasang roda yang dipasangkan pada robot. Perputaran roda diatur dari motor driver dengan menyalurkan listrik kepada motor dc yang sudah dipasangkan roda sebagai outputnya. Kemampuan menembak pada robot diatur dengan adanya sensor *infrared*. Sensor *infrared transmitter* diletakkan didepan robot sedangkan sensor *infrared receiver* membutuhkan empat yang diletakkan disekeliling sisi robot. Sensor infrared receiver akan di sambungkan secara paralel untuk ke empat sensornya dan dihubungkan kepada mikrokontroler dalam hal ini yang digunakan adalah "nodeMCU ESP8266". Robot yang telah dirancang selanjutnya diuji dengan cara mengirimkan kode instruksiyang dihasilkan dari proses *classification* kepada mikrokontroler robot. Pengujian ini dilakukan dengan mengklasifikasi 20 citra pada tiap class. Pengujian ini bertujuan untuk mengetahui apakah ada data yang tidak terkirim atau hilang saat dikirim. Hasil dari klasifikasi akan dirubah menjadi kode instruksi dan dikirim kepada mikrokontroler. Pada mikrokontroler dihitung jumlah kode instruksi yang masuk dan dibandingkan dengan kode yang dikirim. Hasil pengujian ini dapat dilihat pada Tabel IV



a. Grafik *training* dan *validation loss*



b. Grafik *training* dan *validation accuracy*

Gambar 2. Grafik hasil training

Tabel IV
HASIL UJI KOMUNIKASI LAPTOP DENGAN ROBOT

Pose	Hasil Uji
Diam	20
Kanan	20
Kiri	20
Maju	20
Mundur	20
Tembak	20

E. Hasil Pengujian Jarak

Pengujian akurasi model bertujuan untuk membandingkan akurasi model dalam beberapa situasi uji guna mengetahui situasi ideal untuk melakukan *classification*. Pengujian ini menggunakan tangan peneliti dengan variasi jarak tangan terhadap kamera dengan ketinggian tangan sejajar dengan kamera. Pengujian jarak dilakukan dengan cara mengambil 20 citra dari tiap class pada jarak tertentu. Jarak ini dimulai dari 30cm, 50cm, 80cm, dan 100cm antara tangan dengan kamera. Pengujian dilakukan karena adanya perubahan bentuk dari *hand landmark*. Hasil pose yang terdeteksi dari pengujian jarak dapat dilihat pada Tabel V.

Tabel V
POSE YANG TERDETEKSI DARI PENGUJIAN JARAK

Pose	Jarak (cm)			
	30	50	80	100
Diam	20	19	20	20
Kanan	20	20	16	10
Kiri	18	20	20	20
Maju	16	20	20	20
Mundur	20	20	20	20
Tembak	20	20	15	10
Akurasi	0.95	0.99	0.93	0.83

REFERENCES

- [1] Wibowo, Arif Ardy, Gestur Tangan Manusia dalam Karya Fotografi, Jurnal Fotografi Televisi Animasi, vol 17 No 2, Oktober 2021
- [2] Z.Ren, J.Meng, Yuan J. Depth Camera Based Hand Gesture Recognition and its Application in Human-Computer-Interaction. In Processing of the 2011 8th International Conference on Information, Communication and Signal Processing (ICICS). Singapore. 2011.
- [3] Hidayatullah Priyanto, Buku Sakti Deep Learning Computer Vision Menggunakan YOLO untuk Pemula, Stunning Vision AI Academy, 2021
- [4] Leksono, J. W. and Agung Samudra and Nanndo Yannuansa and Ahmad Fauzi, Kendali mobil robot menggunakan isyarat tangan berbasis arduino, Jurnal Electro Luceat, 2020
- [5] Fauzia P. Lestari1, A.B.U. Berlian, M.D.Badrianto, Gesture control car: mengatur gerak mobil hanya dengan memiringkan tangan, 2019
- [6] Kukuh Darmawan Setyanto, Ike Fibriani, S.T, M.T., Sumardi, S.T., M.T., PENGENDALIAN MOBILE ROBOT VISION MENGGUNAKAN WEBCAM PADA OBJEK ARAH PANAH BERBASIS RASPBERRY PI, Jurnal Arus Elektro Indonesia (JAEI)
- [7] Hidayatullah, P., Buku sakti deep learning computer vision menggunakan yolo untuk pemula, Stunning Vision AI Academy, 2021