

Jurnal *Rekayasa Elektrika*

VOLUME 14 NOMOR 3

Desember 2018

**Pengenalan Gestur Jari Menggunakan Pengolahan Citra untuk Mengendalikan
Joint pada Base Robot Lengan** 174-180

Ali Rizal Chaidir, Wahyu Muldayani, dan Guido Dias Kalandro

JRE	Vol. 14	No. 3	Hal 145–220	Banda Aceh, Desember 2018	ISSN. 1412-4785 e-ISSN. 2252-620X
-----	---------	-------	-------------	------------------------------	--------------------------------------

Pengenalan Gestur Jari Menggunakan Pengolahan Citra untuk Mengendalikan Joint pada Base Robot Lengan

Ali Rizal Chaidir, Wahyu Muldayani, dan Guido Dias Kalandro
Fakultas Teknik Universitas Jember
Jl. Kalimantan No. 37, Kampus Tegal Boto, Jember 68121
e-mail: ali.rizal@unej.ac.id

Abstrak—Robot merupakan teknologi yang dapat membantu manusia dalam melakukan suatu kegiatan tertentu. Salah satu jenis robot adalah robot lengan. Robot lengan dapat bekerja secara otomatis maupun manual, beberapa teknik yang dapat digunakan untuk memberikan perintah pada robot adalah menggunakan gestur tangan dan jari, untuk melakukan teknik tersebut dapat memanfaatkan mesin *vision*. Salah satu kelebihan menggunakan mesin *vision* adalah tidak memerlukan banyak komponen elektronika untuk mengenali gestur tangan atau gerakan jari. Fokus penelitian ini adalah mengimplementasikan kamera untuk mengenali gestur jari operator sehingga mampu memberikan perintah kepada robot. Kamera akan menangkap citra dari jari operator, citra tersebut akan diproses di dalam komputer untuk mendapatkan parameter koordinat piksel ujung jari telunjuk dan piksel bagian bawah kanan dari citra, dan kemudian akan ditentukan sudut yang diperoleh dari kedua koordinat tersebut. Hasil yang didapat adalah robot dapat mengikuti gerakan jari pada sudut antara 30° sampai 150°.

Kata kunci: *gestur, pengolahan citra, robot lengan*

Abstract— The Robot is a technology that can help human to do daily activities. One type of robot is an arm robot, arm robot can work automatically, manually, and a combination of both. Some techniques that can be used to give commands to the robot are to use hand and finger gestures, to do the techniques can utilize the vision machine. One of the advantages of using a vision machine is that it does not require many electronic components to recognize hand and finger gestures. The focus of this research is to implement a camera to recognize the operator's finger gestures so that the gesture can give commands to the robot. The camera captures the image of operator's finger, the image is a process inside the computer to obtain the pixel coordinate parameters of the index fingertip and right lower pixel of the image, and then the angle obtained from both coordinates using trigonometric equations. The result is that the robot can follow the movement of the finger when the angle between 30° to 150°.

Keywords: *gesture, image processing, arm robot*

Copyright © 2018 Jurnal Rekayasa Elektrika. All right reserved

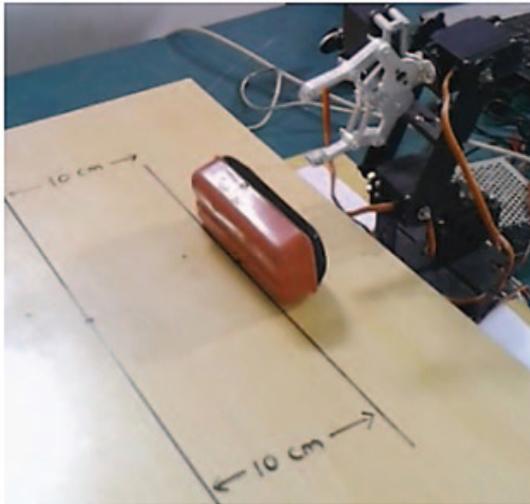
I. PENDAHULUAN

Robot adalah teknologi yang dapat memberikan manfaat untuk kehidupan manusia [1]. Robot dapat membantu manusia dalam berbagai bidang, misalnya di bidang pertanian [2], [3], industri [1], dan medis [4]. Robot dapat bekerja secara otomatis dan manual, gestur tangan atau jari adalah salah satu teknik yang digunakan untuk memberikan perintah pada robot secara manual. Untuk menggunakan teknik tersebut bisa menggunakan sarung tangan yang sudah terpasang sensor [5] dan menggunakan pengenalan gestur berbasis komputer *vision* [6]. Salah satu kelebihan komputer *vision* dibandingkan dengan menggunakan perangkat sensor pada sarung tangan adalah tidak memerlukan banyak komponen elektronika untuk mengenali gerakan tangan atau jari.

Pengendalian berbasis komputer *vision* telah menjadi

topik penelitian yang populer, salah satu alasannya adalah pengguna tidak langsung terhubung dengan komponen elektronika seperti misalnya sensor.

Beberapa penelitian yang berkaitan dengan komputer *vision* sebagai perintah gerakan suatu perangkat atau robot adalah robot beroda yang dikendalikan menggunakan gestur tangan [7]. Tidak hanya robot, perangkat lain seperti pengaturan posisi tempat tidur bagi pasien juga dapat dikendalikan menggunakan gestur tangan, tujuannya adalah agar pasien tidak memerlukan tenaga yang besar untuk mengubah posisi tempat tidur [8]. Selain itu ada beberapa contoh penelitian yang menerapkan komputer *vision* untuk mengendalikan gerakan robot lengan dua DOF, misalnya untuk proses pengambilan barang [9], robot lengan dua DOF tersebut bekerja pada koordinat x dan y (Gambar 1), sehingga robot tersebut dapat memindahkan atau menggeser barang pada sumbu tersebut. Hal tersebut



Gambar 1. Robot pengambil objek dua DOF

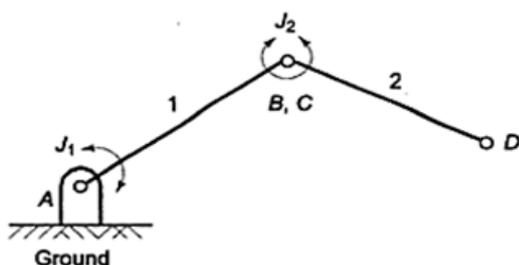
dikarenakan robot lengan tersebut belum memiliki aktuatur pada base robot.

Robot lengan atau robot manipulator memiliki kesamaan dengan lengan manusia. Gambar 2 menunjukkan ilustrasi robot manipulator [10], struktur mekanik robot manipulator terdiri dari links, joint, dan end effector, J1 merupakan posisi joint pada base robot lengan, J2 adalah penghubung antara link 1 dan 2 robot lengan, dan D merupakan posisi *end effector robot*. Joint J1 bergerak ke atas dan ke bawah, sedangkan pada joint J2 bergerak ke kanan dan ke kiri.

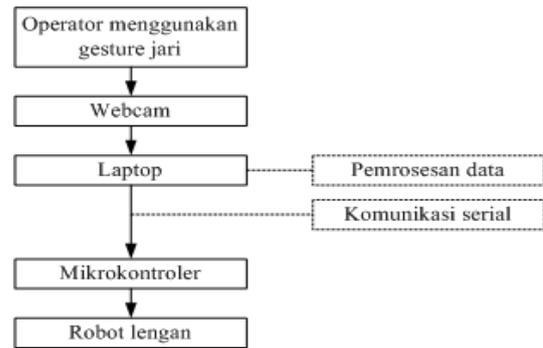
Pada penelitian ini telah dirancang suatu pengenalan gestur tangan untuk mengendalikan robot lengan. Komputer vision digunakan sebagai pengenalan gestur tangan, algoritma *skin detector* akan diterapkan untuk mengenali tangan, berbeda dengan penelitian sebelumnya [11], [12] yang menggunakan algoritma *template matching*, *signature signal algorithm*, dan jaringan syaraf tiruan untuk pengenalan gestur tangan. Pada penelitian ini langkah setelah proses *skin detector* adalah menerapkan algoritma *scanning pixel* untuk menemukan koordinat tangan yang kemudian akan dijadikan sebagai acuan gerakan robot lengan, dengan menggunakan algoritma *scanning pixel*, gerakan tangan pengguna dapat merepresentasikan gerakan robot lengan.

II. METODE

Blok sistem yang telah dibuat dapat dilihat pada Gambar 3, webcam diletakkan di atas jari operator,



Gambar 2. Ilustrasi robot manipulator dua DOF



Gambar 3. Blok sistem

intensitas cahaya disekitarnya konstan, webcam akan meng-capture setiap gerakan dari jari. Setiap gambar yang diambil oleh webcam akan diproses menggunakan laptop, data hasil pemrosesan gambar selanjutnya dikirim ke mikrokontroler menggunakan komunikasi serial, dan tahap terakhir adalah mikrokontroler memberikan sinyal kepada servo. Proses pengolahan gambar terdiri dari dua tahap, yaitu tahap pemisahan *background*, penerapan algoritma deteksi ujung jari dan selanjutnya adalah pengenalan perubahan sudut.

A. Pemisahan Background

Langkah awal untuk melakukan proses pemisahan background adalah mengkonversi gambar yang diperoleh menggunakan webcam A4tech pk-920h HD 1080P menjadi gambar YCbCr menggunakan persamaan (1), tahapan pemisahan background ditunjukkan pada Gambar 4. YCbCr dianggap lebih baik dalam pengenalan karakter daripada RGB [13] dan memiliki tumpang tindih yang kecil antara skin color dan non-skin color dalam berbagai kondisi pencahayaan [14], [15], dan [16].

$$\begin{pmatrix} Y \\ Cb \\ Cr \end{pmatrix} = \begin{pmatrix} 0.2568 & 0.5041 & 0.0979 \\ -0.1482 & -0.2910 & 0.4392 \\ 0.4392 & -0.4392 & -0.00714 \end{pmatrix} \begin{pmatrix} R \\ G \\ B \end{pmatrix} + \begin{pmatrix} 16 \\ 128 \\ 128 \end{pmatrix} \quad (1)$$

Di setiap citra dimungkinkan akan terdapat derau yang berbentuk bintik-bintik. Untuk mengurangi derau tersebut, tahap selanjutnya adalah melakukan filter median menggunakan persamaan (2), filter median berguna untuk menghilangkan bintik-bintik pada sebuah citra [17].

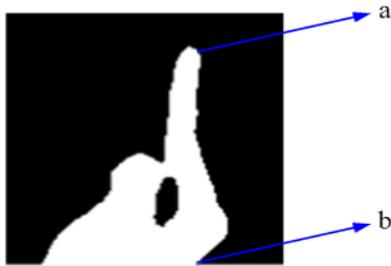
$$Y[x,y] = \text{median}(I_{orig}[i,j], i, j \in \text{nbor}[x,y]) \quad (2)$$

dengan $\text{nbor}[x,y]$ adalah sub image dari citra YCrCb dan $Yorig[i,j]$ adalah sebuah citra YCrCb.

Langkah terakhir adalah melakukan proses *thresholding* menggunakan persamaan (3). Nilai ambang diatur untuk



Gambar 4. Tahapan pemisahan citra



Gambar 5. Posisi Parameter a dan b pada jari operator

memisahkan citra background dan foreground, citra foreground direpresentasikan dengan nilai piksel di setiap parameter YCbCr dengan nilai 255 (berwarna putih), sedangkan warna piksel citra background adalah berwarna hitam.

$$G(x,y) = \begin{cases} 1 & \text{if } T_{y\max} \geq Y(x,y) \geq T_{y\min} \text{ and} \\ & T_{cr\max} \geq Y(x,y) \geq T_{cr\min} \text{ and} \\ & T_{cb\max} \geq Y(x,y) \geq T_{cb\min} \\ 0 & \text{if } T_{y\max} < Y(x,y) < T_{y\min} \text{ and} \\ & T_{cr\max} < Y(x,y) < T_{cr\min} \text{ and} \\ & T_{cb\max} < Y(x,y) < T_{cb\min} \end{cases} \quad (3)$$

dimana $G(x,y)$ adalah image biner, $Y(x,y)$ adalah image YCrCb, $T_{y\max}$ dan $T_{y\min}$ menyatakan nilai *threshold* maksimal dan minimal komponen Y, $T_{cr\max}$, dan $T_{cr\min}$ menyatakan nilai *threshold* maksimal dan minimal komponen Cr, $T_{cb\max}$ dan $T_{cb\min}$ menyatakan nilai *threshold* maksimal dan minimal komponen Cb.

B. Algoritma Deteksi Ujung Jari

Dibutuhkan dua koordinat untuk menentukan besaran sudut servo, koordinat pertama diperoleh dari ujung jari telunjuk operator (huruf 'a' pada Gambar 5), yang kedua adalah diperoleh dari piksel putih dari gambar paling kanan bawah (huruf 'b' pada Gambar 5).

Tahap selanjutnya setelah mendapatkan 2 koordinat tersebut adalah mencari nilai gradien dari 2 koordinat tersebut dan selanjutnya adalah menemukan besaran sudut dengan menggunakan persamaan trigonometri. Tahap terakhir adalah mengirimkan nilai sudut tersebut secara serial ke mikrokontroler untuk kemudian digunakan servo untuk menentukan gerakan dari robot.

Koordinat pertama (posisi 'a' pada Gambar 5) diperoleh menggunakan listing program berikut ini:

```
For i = 0 To 240 - 1
  For j = 0 To 320 - 1
    If gam_bin.Data(i, 319 - j, 0) = 255 Then
      y_jari = 320 - j
      x_jari = 240 - i
      keluar = 1
    Exit For
  End If
Next
If keluar = 1 Then
  Exit For
End If
Next
```

Sedangkan koordinat kedua (posisi 'b' pada Gambar 5) diperoleh menggunakan listing program berikut ini:

```
For i = 0 To 240 - 1
  For j = 0 To 320 - 1
    If gam_bin.Data(239 - i, 319 - j, 0) = 255 Then
      y_bawah = 320 - j
      x_bawah = i
      keluar = 1
    Exit For
  End If
Next
If keluar = 1 Then
  Exit For
End If
Next
```

Citra yang digunakan adalah dengan ukuran lebar 320 piksel dan tinggi 240 piksel, proses pemindaian koordinat piksel titik 'a' dimulai dari piksel terkanan dan teratas citra. Proses pemindaian tersebut dilakukan sampai menemukan piksel dengan warna putih, dan kemudian koordinat titik piksel tersebut disimpan. Hampir sama dengan proses pemindaian koordinat piksel titik 'a', proses pemindaian koordinat piksel titik 'b' dimulai dari piksel terkanan dan terbawah. Koordinat 'a' disimpan di variabel x_jari dan y_jari , sedangkan koordinat 'b' disimpan di variabel x_bawah dan y_bawah . Selanjutnya untuk mendapatkan besaran sudut menggunakan listing program berikut:

```
m = (y_jari - y_bawah) / (x_jari - x_bawah)
sudut = (Math.Atan(m)) / Math.PI * 180
```

Setelah nilai besaran sudut diperoleh selanjutnya nilai tersebut dikirim ke mikrokontroler melalui komputer menggunakan perintah:

```
SerialPort1.Write(TextBox9.Text)
```

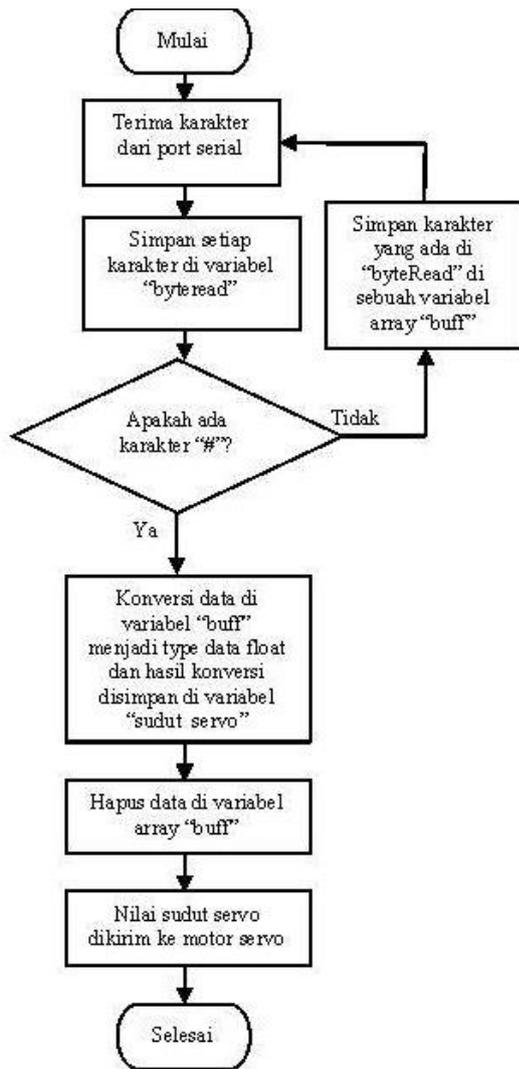
Arti perintah tersebut adalah data yang ditampilkan pada textbox 9 akan dikirim ke mikrokontroler, tapi sebelum itu di setiap akhir nilai sudut yang akan dikirim diberi tanda '#'. Hal ini dilakukan untuk memberikan informasi ke mikrokontroler bahwa data yang dikirim adalah satu paket data serial, perintah yang digunakan adalah

```
str1 = Convert.ToString(sudut)
TextBox9.Text = str1 + "#"
```

Sedangkan pada mikrokontroler program yang digunakan adalah terdiri dari 3 tahap, tahap pertama

```
#include <Servo.h>
Servo myservol;
byte byteRead;
char buff[20];
float sudut_servo[1]={50};
int i=0;
int r=0;
void setup()
{ myservol.attach(9);
  Serial.begin(9600);
```

adalah persiapan variabel, pin output untuk motor servo, komunikasi serial, dan library yang digunakan, yaitu:

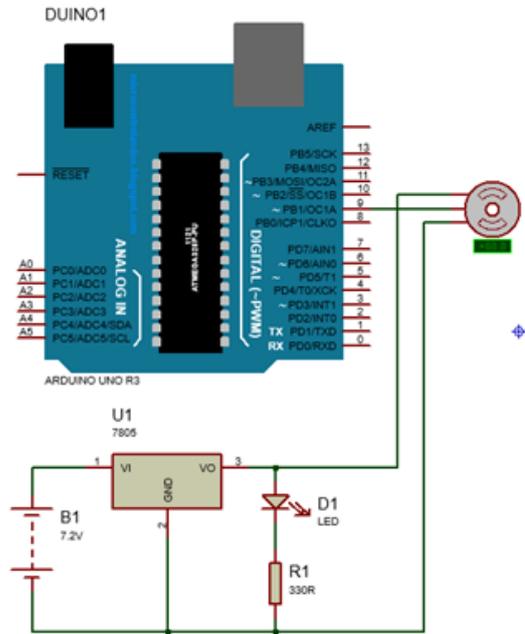


Gambar 6. Diagram alir tahap kedua dari kode program pada mikrokontroler

Tahap kedua adalah algoritma untuk membaca data serial yang dikirim oleh komputer ke mikrokontroler. Mikrokontroler harus mampu membaca data yang dikirim oleh komputer untuk kemudian digunakan sebagai nilai sudut servo pada robot. Program yang digunakan yaitu:

```
void baca_serial()
{
  if (Serial.available())
  {
    byteRead=Serial.read();
    if (byteRead=='#')
    {
      sudut_servo[r]=atof(buff);
      sudut_servo[r]=sudut_servo[r]/1;
      i=0;
      for (int del=0;del<20;del++)
      {buff[del]=0;}
      r++;
      if (r==0)
      {r=0;}
    }
    else
    { buff[i]=byteRead;
      i++; }
  }
}
```

Diagram alir dari kode program tahap kedua pada mikrokontroler ditunjukkan pada Gambar 6. Setiap karakter yang dikirim oleh komputer akan diterima



Gambar 7. Rangkaian elektronika

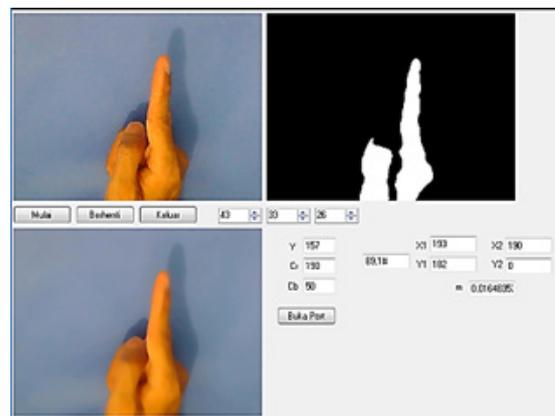
mikrokontroler melalui jalur serial. Setiap karakter yang diterima mikrokontroler disimpan di sebuah variabel, selama karakter yang diterima bukan karakter '#'. Karakter akan disimpan terlebih dahulu di variabel tersebut. Setelah mikrokontroler menerima karakter '#' proses konversi dilakukan. Karakter yang tersimpan sebelumnya dikonversi menjadi tipe data float dan selanjutnya data tersebut digunakan untuk menentukan sudut putaran motor servo.

Tahap ketiga adalah melakukan pengiriman data ke motor servo robot lengan. Program yang digunakan adalah:

```
void putar_servo()
{ myservol.write(sudut_servo[0]); }
```

Program pada tahap ketiga mengirimkan data yang telah dikonversi menjadi tipe data float. Data yang telah dikonversi disimpan di variabel sudut_servo array ke 0.

C. Rangkaian Elektronika



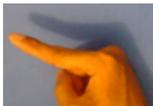
Gambar 8. Tampilan antarmuka algoritma pengenalan gestur jari

Diperlukan rangkaian elektronika untuk mengendalikan motor servo yang digunakan robot lengan. Motor servo tersebut digerakkan oleh sinyal yang diberikan oleh pin “analog out” dari mikrokontroler, mikrokontroler yang digunakan adalah ATmega328P yang telah terpasang pada board Arduino, pin yang digunakan tersebut adalah pin ke 9. Pin 9 dihubungkan pada kaki “sig” dari motor servo (Gambar 7), sedangkan pin 5 V dan GND motor servo dihubungkan ke keluaran dari IC regulator 7805.

III. HASIL DAN PEMBAHASAN

Tampilan antarmuka dari algoritma gestur jari ditunjukkan pada Gambar 8, beberapa informasi yang dapat

Tabel 1. Hasil pengujian gerakan robot terhadap gestur jari (jarak antara webcam dengan jari operator adalah 30 cm)

No	Gestur Jari	Sudut yang dikirim	Respon robot
1		50	
2		30	
3		45	
4		60	
5		90	
6		120	
7		135	
8		150	
9		117	

Tabel 2. Hasil pengujian gerakan robot terhadap gestur jari (jarak antara webcam dengan jari operator adalah 45 cm)

No	Hand gesture	Sudut yang dikirim	Respon robot
1		45	
2		30	
3		45	
4		60	
5		90	
6		120	
7		135	
8		150	
9		136	

ditampilkan di tampilan antarmuka tersebut diantaranya adalah gambar jari dalam ruang warna RGB dan YCbCr, nilai kemiringan antara ujung telunjuk dan pangkal, dan informasi yang dapat dilihat juga adalah informasi sudut yang harus dikirim ke motor.

Algoritma pemrograman tersebut telah berhasil mengenali gestur jari. Kesimpulan tersebut dilihat dari hasil pengujian algoritma untuk menggerakkan motor servo yang terletak pada base dari robot lengan. Beberapa pengujian dapat dilihat pada Tabel 1, 2, dan 3. Pengujian dilakukan sebanyak 9 kali pada setiap jarak yang berbeda antara webcam dengan jari telunjuk operator, dan pengujian

Tabel 3. Hasil pengujian gerakan robot terhadap gestur jari (jarak antara webcam dengan jari operator adalah 60 cm)

No	Hand gesture	Sudut yang dikirim	Respon robot
1		36	
2		30	
3		45	
4		60	
5		90	
6		120	
7		135	
8		150	
9		13	

dilakukan dengan sudut yang berbeda yaitu 20°, 30°, 45°, 60°, 90°, 120°, 135°, 150°, dan 170°. Dari tabel yang ditunjukkan tersebut dapat dilihat bahwa pengujian ke 2 sampai 8 arah gerakan robot lengan di setiap jarak yang berbeda mengikuti gerakan jari, sedangkan pengujian ke 1 dan ke 9 di setiap jarak yang berbeda, yaitu pada sudut 20° dan 170° gerakan robot tidak sesuai dengan arah gestur jari.

Ketidaksesuaian gerakan robot dengan gestur jari tersebut disebabkan karena algoritma yang digunakan

tidak berhasil membaca koordinat piksel ujung jari dan koordinat piksel bagian bawah kanan dari tangan.

Algoritma yang telah dibuat pada penelitian ini dapat digunakan untuk tujuan lain, seperti misalnya untuk mengendalikan posisi tempat tidur pasien disabilitas. Pengendalian posisi tempat tidur pasien disabilitas menggunakan gestur tangan telah dilakukan [8]. Pada penelitian tersebut terdapat empat gestur yang perlu dikenali, gestur pertama digunakan untuk menaikkan posisi tempat tidur, gestur kedua untuk menurunkan, ketiga dan keempat digunakan untuk memiringkan tempat tidur ke kanan dan ke kiri. Berbeda dengan penelitian tersebut yang dapat dikatakan bahwa setiap gestur tersebut menggantikan sebuah tombol untuk mengubah posisi tempat tidur, pada penelitian ini menawarkan algoritma lain yaitu gestur tidak digunakan sebagai pengganti tombol, akan tetapi dapat dikatakan sebagai pengganti tuas untuk merubah posisi tempat tidur pasien disabilitas.

IV. KESIMPULAN

Algoritma pengenalan gestur jari telunjuk untuk mengendalikan robot berhasil dibuat. Langkah pertama yang dilakukan adalah melakukan proses pemisahan citra jari operator dengan latar belakangnya. Citra yang dihasilkan adalah citra biner, warna piksel putih menunjukkan jari operator dan piksel warna hitam adalah latar belakang citra. Langkah berikutnya adalah melakukan pengenalan perubahan gerakan jari operator, posisi ujung jari telunjuk dan posisi warna piksel yang berada di posisi kanan bawah dari citra digunakan sebagai parameter untuk menentukan nilai sudut. Langkah terakhir adalah mengirim nilai tersebut ke mikrokontroler dan kemudian mikrokontroler mengirimkannya ke sebuah motor servo robot. Algoritma pemrograman mampu mengenali gestur jari dan data yang diterima mikrokontroler secara serial sesuai dengan yang dikirimkan dari komputer, sehingga gerakan robot mampu mengikuti gestur jari operator. Hasil yang didapat adalah robot dapat mengikuti gerakan jari antara sudut antara 30° sampai 150° di setiap jarak antara kamera dan jari operator yang berbeda.

REFERENSI

- [1] B. Siciliano, L. Sciavicco, G. Oriolo, "Robotics: modelling planning and control," Springer, 2009.
- [2] U. Masood Ul, U. Mukhtar, I. Jamshed, "Towards autonomy in agriculture: design and prototyping of a robotic vehicle with seed selector," International Conference on Robotics and Artificial Intelligence (ICRAI), pp 37 - 44, 2016.
- [3] U. Saurabh, K. Anil, "Automated seed sowing agribot using arduino," International Conference on Communication and Signal Processing (ICCCSP), pp 1379 - 1383, 2016.
- [4] B. Siciliano, O. Khatib, "Springer Handbook of Robotics," Springer, 2008.
- [5] U. Rajkanna, M. Mathankumar, K. Gunasekaran, "Hand gesture based mobile robot control using PIC microcontroller," International Conference on Circuits, Power and Computing Technologies [ICCPCT], pp 1687 - 1691, 2014.

- [6] K. Kishore, S. Hannes, K. Achim, S. Dirk, "Real time interaction with mobile robots using hand gestures," ACM/IEEE International Conference on Human-Robot Interaction (HRI), pp 177 - 178, 2012.
- [7] M. Manigandan, I. Manju Jackin, "Wireless vision based mobile robot control using hand gesture recognition through perceptual color space," International Conference on Advances in Computer Engineering, pp. 95 - 99, 2010.
- [8] L. Kalpana, M. Swati, "Automatic bed position control based on hand gesture recognition for disabled patients," International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT), pp.148 - 153, 2016.
- [9] C. Ali Rizal, S. Alfredo Bayu, K. Guido Dias, "Design of a gripping imitator robotic arm for taking an object," International Conference on Information and Communication Technology (ICoICT), pp. 1 - 5, 2016
- [10] R. K. Mittal, I. J. Nagrath, "Robotics and Control," McGraw-Hill.
- [11] C. İbrahim Baran, K. Mehmet, "Development of a robotic-arm controller by using hand gesture recognition," International Symposium on Innovations in Intelligent Systems and Applications, pp 1 - 5, 2012.
- [12] S. Hongyong, F. Weijiang, G. Naiyang, H. Xuhu, L. Zhigang, "Toward robust ego-centric hand gesture analysis for robot control", IEEE International Conference on Signal and Image Processing (ICSIP), pp. 661 - 666, 2016.
- [13] W. Wei, P. Jing, "Hand segmentation using skin color and background information," International Conference on Machine Learning and Cybernetics, Vol. 4., pp. 1487 - 1492, 2012.
- [14] D. Ahmad Yahya, A. Junaidi, Md. Jahangir Alam, "Adaptive skin color model for hand segmentation," International Conference on Computer Applications and Industrial Electronics, pp. 486 - 489, 2010.
- [15] D. Ahmad Yahya, A. Junaidi, Md. Jahangir Alam, "A new method for hand segmentation using free-form skin color model," International Conference on Advanced Computer Theory and Engineering(ICACTE), Vol. 2, pp. V2-562 - V2-566, 2010.
- [16] B. Ganesh Choudhary, Chethan Ram B V, "Real time robotic arm control using hand gestures," International Conference on High Performance Computing and Applications (ICHPCA), pp. 1 - 3, 2014.
- [17] E. Ramaraj, A. Senthil Rajan, "Median filter using open multiprocessing in agriculture," IEEE 10th International conference on signal processing proceedings, pp. 42-45, 2010.

Penerbit:

Jurusan Teknik Elektro, Fakultas Teknik, Universitas Syiah Kuala

Jl. Tgk. Syech Abdurrauf No. 7, Banda Aceh 23111

website: <http://jurnal.unsyiah.ac.id/JRE>

email: rekayasa.elektrika@unsyiah.net

Telp/Fax: (0651) 7554336

