

METODE ITERASI UNTUK LENGAN 3 DOF PADA ROBOT HYBRID ABUROBOCON 2016

FirmandaGies Algiffari¹, Indrazno Siradjuddin², Denda Dewatama³, TundungSubali Patma⁴

TeknikElektronika, TeknikElektro, PoliteknikNegeri Malang

firmadagiesa@gmail.com¹, indrazno@polinema.ac.id², dewatama@polinema.ac.id³

Abstract

Robot manipulator merupakan sebuah robot yang memiliki mekanisme pergerakan mirip dengan tangan manusia. Perkembangan robot lengan di bidang pendidikan sangat laju. Termasuk penggunaan robot lengan untuk mengatasi masalah dalam lomba robot aburobocon 2016. Robot lengan diharapkan dapat mengambil *propeller* dan meletakkannya pada *turbine engine*. Metode iterasi merupakan metode analisis untuk mengatasi suatu sistem persamaan.

Penggunaan metode iterasi jacobian untuk mengatasi pergerakan invers kinematik pada robot untuk memudahkan menghitung posisi *end-effector*. Peralatan yang digunakan adalah robot lengan 3 DOF dengan penggerak motor servo turnigy 5513MD, Arduino Mega, Regulator DC to DC, Baterai LiPo 3 cell. Pada pengujian metode iterasi dapat dilihat rata-rata error posisi pada sumbu x yaitu sebesar 0.01 cm dan pada sumbu y yaitu sebesar 0.02 cm. Serta rata-rata error posisi sudut motor servo 1 yaitu sebesar 0.06, sudut motor servo 2 yaitu sebesar 0.09, dan sudut motor servo 3 yaitu sebesar 0.09.

Keywords: *Manipulator Robot, Iteration Metode, Inverse Kinematic, Arduino, Motor Servo*

1. Pendahuluan

Kemajuan dalam bidang ilmu pengetahuan dan teknologi merupakan cara untuk mendapatkan sesuatu dengan kualitas yang lebih baik yaitu lebih murah, lebih mudah, lebih cepat dan menyenangkan. Salah satunya teknologi yang selalu berkembang pesat pada kehidupan manusia, khususnya pada bidang elektronika. Hal ini ditandai dengan adanya berbagai peralatan elektronik yang telah diciptakan dan dapat dioperasikan secara otomatis. Disebabkan oleh kemajuan inilah maka dikembangkan suatu ilmu yang merupakan salah satu dari ilmu elektronika yaitu bidang ilmu Robotika. Hal ini dapat dilihat dari muncul beragam jenis robot yang diciptakan sehingga, muncul beragam kontes robotika. Salah satu kontes robot yang bergengsi di Asia adalah Abu Asia-Pacific Robot Contes 2016 yang bertemakan *Clean Energy Recharging the World* dan di Indonesia disebut KRAI 2016 (Kontes Robot Abu Robocon Indonesia).

Jenis robot yang akan dibuat adalah jenis robot lengan yang disebut juga robot *manipulator*. Definisi tentang robot *manipulator* adalah sebuah mekanisme pergerakan yang tersusun dari beberapa bagian secara seri yang terhubung dengan sebuah poros yang bergerak melingkar maupun bergeser yang memiliki derajat kebebasan *Degree of Freedom*. Pambudi, W et al (2014).

Sedangkan menurut Nugraha (2011), derajat kebebasan (*Degree of Freedom*) suatu robot dapat diartikan sebagai jumlah gerakan independen yang dapat dibuat suatu objek terhadap sistem koordinat yang dapat menyebabkan perubahan posisi atau orientasi.

Dalam pembuatan alat ini penulis menggunakan metode iterasi yang digunakan pada lengan pengambil dan penaruh *propeller*. Lengan difungsikan khusus untuk mengambil dan meletakkan suatu objek yang berupa *propeller*. Lengan robot ini diimplementasikan pada robot *hybrid*. Sendi-sendi pada lengan akan bergerak secara sekuensial untuk dapat mengambil dan menaruh *propeller*. Gerak lengan ini memerlukan perhitungan sudut. Sehingga dengan melakukan perhitungan sudut, akan dapat diketahui posisi dan arah ujung lengan (*end effector*). Dengan menggunakan metode iterasi diharapkan pergerakan lengan menjuposisi yang diinginkan bergerak dengan lembut.

Berdasarkan latar belakang ini maka pengendalian robot dilakukan dengan metode iterasi pada lengan 3 DOF robot hybrid Aburobocon 2016 diharapkan dengan adanya pembuatan alat ini dapat memberikan manfaat bagi mahasiswa atau peneliti yang berkecimpung di dunia robotika serta memberikan

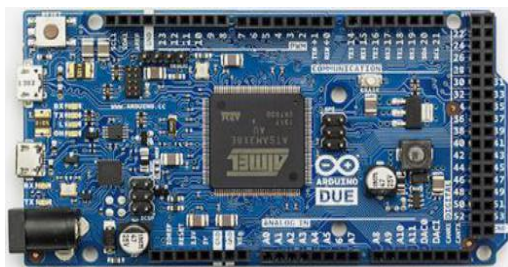
motifasi pada mahasiswa Program Studi Teknik Elektronika Politeknik Negeri Malang.

2. Tinjauan Pustaka

Lengan pengambil dan penaruh *propeller* merupakan sebuah lengan secara sekuen bergerak untuk mengambil dan menaruh propeller. Dalam perancangan alat tersebut diperlukan teori – teori yang menunjang alat yang dirancang adalah arsitektur perangkat keras yang meliputi perangkat elektronik, mekanik, dan organisasi dalam pengolahan perangkat lunak untuk basis pengetahuan dan pengendalian secara nyata. Pada bab ini akan membahas teori – teori dasar yang digunakan untuk merealisasikan lengan pengambil dan penaruh *propeller* pada *robot hybrid* Abu Robocon 2016. Dengan mikrokontroler Arduino Mega sebagai pengendali utamanya. Adapun penjelasan dari teori - teori dan hal pendukung lainnya yang digunakan dalam perancangan alat ini adalah sebagai berikut:

2.1 AduinoMega

Arduino merupakan salah satu jenis mikrokontroler single-board yang bersifat open-source, diturunkan dari Wiring platform, dirancang untuk memudahkan penggunaan elektronik alam berbagai bidang. Hardware nya memiliki prosesor Atmel AVR dan software nya memiliki bahasa pemrograman sendiri. Hardware yang diprogram menggunakan bahasa berbasis Wiring (sintaks + perpustakaan), mirip dengan C++ dengan beberapa penyederhanaan dan modifikasi, dan pengolahan berbasis IDE. Karseno, (2011).

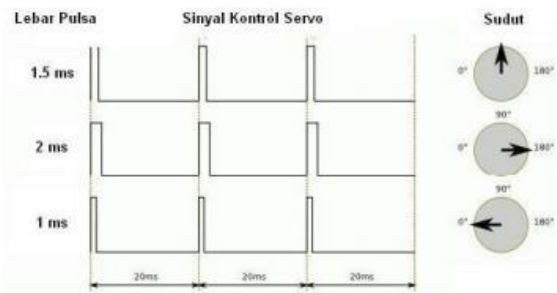


Gambar 2.1 Board Arduino Mega

2.2 Motor Servo Turnigy 5513MD

Motor servo turnigy 5513MD merupakan motor servo digital dengan gear berjenis metal dan memiliki dimensi tinggi 44,2 mm, lebar 20,2 mm, dan panjang 54 mm. Tipe servo ini memiliki torsi 11,89 kgcm sampai 12,25

kgcm dengan *ranget* tegangan operasi 4,8 volt sampai 6 volt. Pada saat supply tegangan 4,8 volt, servo mampu bergerak dengan kecepatan putar 0,2 sec/60°. Sedangkan saat supply tegangan 6 volt, servo mampu bergerak dengan kecepatan putar 0,18 sec/60°. Motor servo bergerak ke kanan atau ke kiri, tergantung dari nilai delay yang kita berikan. Untuk membuat servo pada posisi center, berikan pulsa 1.5ms. Untuk memutar servo ke kanan, berikan pulsa <= 1.3ms, dan pulsa >= 1.7ms untuk berputar ke kiri dengan delay 20ms. Sujarwata, (2013), seperti ilustrasi berikut:

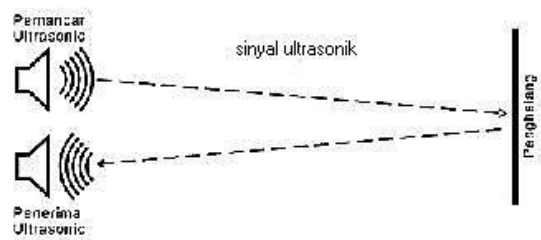


Gambar 2.3 Pensinyalan motor servo

2.3 Sensor Ultrasonic

Sensor ultrasonik adalah sensor jarak yang mengukur jarak dengan cara mengukur gelombang ultrasonik yang dipantulkan ke suatu objek yang jaraknya akan diukur oleh sensor tersebut. Sensor ini memiliki *transmitter* dan *receiver* sebagai pemancar dan penerima gelombang ultrasonik. Gelombang ultrasonik adalah gelombang dengan besar frekuensi di atas frekuensi gelombang suara yaitu lebih dari 20 KHz. Seperti telah disebutkan bahwa sensor ultrasonik terdiri dari rangkaian *transmitter* dan rangkaian penerima ultrasonik yang disebut *receiver*. Sinyal ultrasonik yang dibangkitkan akan dipancarkan dari *transmitter* ultrasonik. Ketika sinyal mengenai benda penghalang, maka sinyal ini dipantulkan, dan diterima oleh *receiver* ultrasonik. Sinyal yang diterima oleh rangkaian *receiver* dikirimkan ke rangkaian mikrokontroler untuk selanjutnya diolah untuk menghitung jarak terhadap benda didepannya (bidang pantul). Zein, (2013).

Prinsip kerja dari sensor ultrasonik dapat ditunjukkan dalam gambar dibawah ini :



Gambar 2.6 Prinsip Kerja Sensor Ultrasonik

2.4 Limit Switch

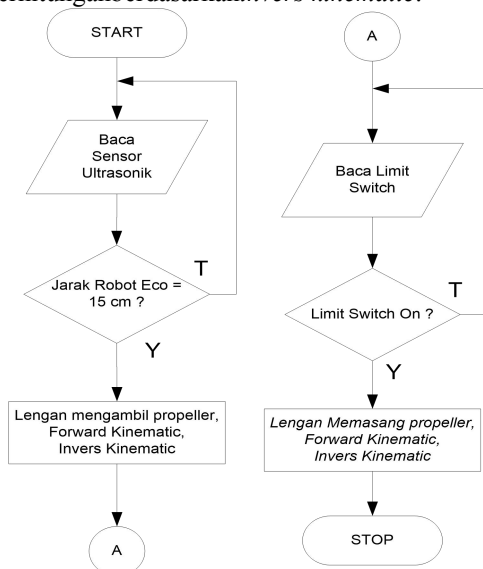
Limit switch adalah salah satu sensor yang akan bekerja jika pada bagian aktuatornya ditekan suatu benda, baik dari samping kiri ataupun kanan, mempunyai makro switch yang berfungsi untuk mengontakkan atau sebagai pengontak, gambar batang yang mempunyai roda pada limit switch dinamakan aktuator lalu diikat dengan sebuah baud, berfungsi untuk menerima tekanan dari luar, roda berfungsi agar pada saat limit Switch menerima tekanan, bisa bergerak bebas, kemudian mempunyai tiga lubang pada bodynya berfungsi untuk dudukan baud pada saat pemasangan. Irmansyah, (2013).



Gambar 2.7 Limit Switch

3. Flowchart Sistem Kerja Alat

Flowchart sistem software menjelaskan tentang urutan algoritma atau urutan kerja program. Sistem kerja software menggunakan kontrol sekuensial. Sensor ultrasonic digunakan untuk mendeteksi keberadaan target (*propeller*) dan memberi input data yang digunakan untuk menggerakkan lengan robot mengambil *propeller*. Limit switch sebagai input data untuk menggerakkan lengan robot memasang *propeller*. Nilai koordinat posisi (x,y,z) digunakan dalam proses perhitungan berdasarkan *invers kinematic*.



Gambar 3.5 Flowchart system software

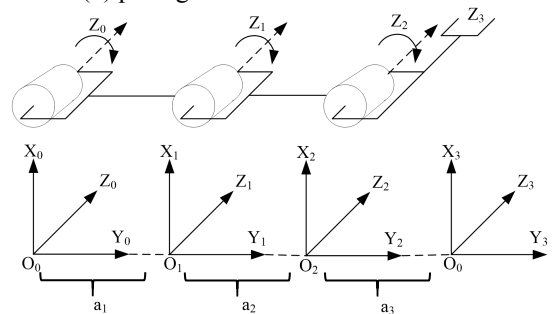
Setelah sistem diaktifkan, sensor ultrasonic akan aktif untuk mendeteksi keberadaan robot eco yang membawa *propeller*. Nilai jarak 20 cm yang telah ditentukan melalui pengukuran menjadi setpoint. Setpoint yang telah tercapai mengakibatkan lengan robot bergerak mengambil *propeller* menggunakan metode *invers kinematic*. *Invers kinematic* mendapatkan data input berupa posisi koordinat (x,y,z) yang nantinya akan diolah menjadi sebuah gerakan.

Dalam melakukan gerakan memasang *propeller*, sistem mendapatkan data input dari *limit switch*. Limit switch akan memberikan data saat robot mencapai posisi pemasangan *propeller*. Lengan robot bergerak memasang *propeller* menggunakan metode *invers kinematic* menggunakan data berupa koordinat (x,y,z) dari proses perhitungan.

3.1 Perancangan Forward Kinematic

Forward

kinematic adalah analisis kinematik untuk mendapatkan koordinat posisi (x,y,z) jika diketahui sudut dari tiap sendi. Misalnya, jika mempunyai robot n-DOF dan diketahui sudut dari tiap sendi maka dapat digunakan analisis *forward kinematic* untuk mendapatkan koordinat posisi robot. Pada gambar 3.6 berikut ini dapat dilihat diagramatik joint dengan sumbu koordinat (y) pada garis vertikal dan sumbu (x) pada garis horizontal.



Gambar 3.6 Diagramatik Joint

Dari gambar diagramatik joint di atas terdapat Denevit-Hartenberg parameter, sehingga alat ini memiliki notasi parameter seperti pada tabel berikut :

Tabel 3.1 DH Parameter

Link	a_i	a_i	d_i	θ_i
1	0	a_1	0	θ_1
2	0	a_2	0	θ_2
3	0	a_3	0	θ_3

Prinsip dasar representasi D-H adalah suatu matriks (4x4) yang berupa sistem koordinat dari suatu *link* dengan *link* yang terhubung pada pangkalnya (*link* sebelumnya). Sehingga dari

parameter pada tabel diatas direpresentasikan kedalam matrix 4x4 adalah sebagai berikut :

Matrix A pada sendi ke-n adalah :

$$A = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cos\alpha_i & \sin\theta_i \sin\alpha_i & a_i \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \sin\alpha_i & a_i \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Dari D-H parameter pada table 3.1 yang telah dibuat, kemudian dipresentasikan kedalam matrix 4x4 Denevit-Hartenberg menjadi:

$$A_1 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & a_1 \cos\theta_1 \\ \sin\theta_1 & \cos\theta_1 & 0 & a_1 \sin\theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & a_2 \cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & a_2 \sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 & 0 & a_3 \cos\theta_3 \\ \sin\theta_3 & \cos\theta_3 & 0 & a_3 \sin\theta_3 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_0^3 = A_1 \cdot A_2 \cdot A_3$$

$$T_0^3 = \begin{bmatrix} \cos_{123} & -\sin_{123} & 0 & a_1 \cos_{123} + a_2 \cos_{12} + a_3 \cos_{123} \\ \sin_{123} & \cos_{123} & 0 & a_1 \sin_{123} + a_2 \sin_{12} + a_3 \sin_{123} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Dimana :

A₁ = Persamaan matrix untuk *joint* pertama

A₂ = Persamaan matrix untuk *joint* kedua

A₃ = Persamaan matrix untuk *joint* ketiga

T₀³ = Hasil perkalian matrix A₁, A₂ dan A₃

Cos₁₂ = Cos (θ₁+θ₂)

Cos₁₂₃ = Cos (θ₁+θ₂+θ₃)

Sin₁₂ = Sin (θ₁+θ₂)

Sin₁₂₃ = Sin (θ₁+θ₂+θ₃)

Catatan : bahwa kolom terakhir dari matrix diatas menyatakan komponen (x, y, dan z). Sehingga persamaan x, y, dan z adalah :

$$x = a_1 \cos\theta_1 + a_2 \cos(\theta_1 + \theta_2) + a_3 \cos(\theta_1 + \theta_2 + \theta_3)$$

$$y = a_1 \sin\theta_1 + a_2 \sin(\theta_1 + \theta_2) + a_3 \sin(\theta_1 + \theta_2 + \theta_3)$$

$$z = 0$$

$$\alpha =$$

$$\beta =$$

$$\gamma =$$

3.2 Perancangan Inverse Kinematic

Inverse kinematic adalah analisis kinematik untuk mendapatkan besar sudut dari masing-masing sendi jika diketahui koordinat posisi (x, y, z). Dalam

permasalahan digunakan metode iterasi untuk memecahkan masalah adalah *Inverse kinematic*.

Inverse kinematic 3 DOF

Plannardapat diselesaikan dengan cara metode iterasi seperti berikut ini :

$$\zeta = f(\bar{q}) \tag{1}$$

Keterangan:

ζ = orientasi posisi (x,y,z,α,β,γ)

q̄ = kecepatan rotasi

Persamaan 1 adalah persamaan forward kinematic.

Dengan pendekatan forward kinematic velocity maka

$$\begin{aligned} \zeta &= f(\bar{q}) \\ \frac{d\zeta}{dt} &= \frac{df(\bar{q})}{dt} \\ \xi &= \frac{df(\bar{q})}{d\bar{q}} \cdot \frac{d\bar{q}}{dt} \\ \xi &= \frac{df(\bar{q})}{d\bar{q}} \cdot \dot{q} \\ \xi &= J \cdot \dot{q} \end{aligned} \tag{2}$$

Desain kontroler dengan fungsi untuk mereduksi antara ξ-ξ̇, maka:

$$E = \xi - \xi \tag{3}$$

Kontroler dapat mereduksi error secara eksponensial dengan fungsi error decrease berikut:

$$\dot{E} = -\lambda \cdot E \tag{4}$$

Maka kecepatan error

eksponensial yaitu sebagai berikut :

$$\begin{aligned} E &= \xi - \xi \\ \frac{dE}{dt} &= \frac{d\xi}{dt} - \frac{d\xi}{dt} \end{aligned}$$

Karena ξ konstan, maka $\frac{d\xi}{dt} = 0$. Sehingga

$$\dot{E} = \xi$$

$$\dot{E} = J \cdot \dot{q}$$

$$-\lambda \cdot E = J \cdot \dot{q}$$

$$\dot{q} = -\lambda \cdot J^{-1} \cdot E$$

Karena $J \in R^{m \times n}$, dimana $m \geq n$ maka J^{-1} adalah pseudo inverse

$$J^{-1} \rightarrow J^\dagger = (J^* \cdot J)^{-1} \cdot J^*$$

Sehingga

$$\dot{q} = -\lambda \cdot J^\dagger \cdot E$$

Keterangan:

q̇ = posisi target

λ = nilai decrease pada fungsi error decrease

J[†] = invers dari matriks jacobian

E = nilai error

Untuk mendapatkan nilai sudut dari tiap joint maka digunakan metode iterasi sebagai berikut:

$$q_k = q_{k-1} + ts \cdot \dot{q}$$

Keterangan

q_k = Posisi sudut baru

q_{k-1} = Posisisudutawal
 ts =Waktu sampling
 \dot{q} = Kecepatansudut

Setelahmendapatkanposisisudut yang barumasukkanpersamaandiataskepersamaan *forward kinematic* sehinggadidapatkanpersamaan

$$\xi = f(q_k)$$

Dari persamaandiatasdidapatkanorientasiposisi target.

Denganmelakukanperulanganterusmenerusdalammenariposisisudutbarudengannilai λ yang telahditentukansampainilai $E < 0.001$ makaposisisudutbaruakanmendekatiposisisudut target.

4. Pengujian

PengujianMetodelterasimenggunakan software Matlabsehinggadidapatkanhasilpengujiansebagaimana berikut

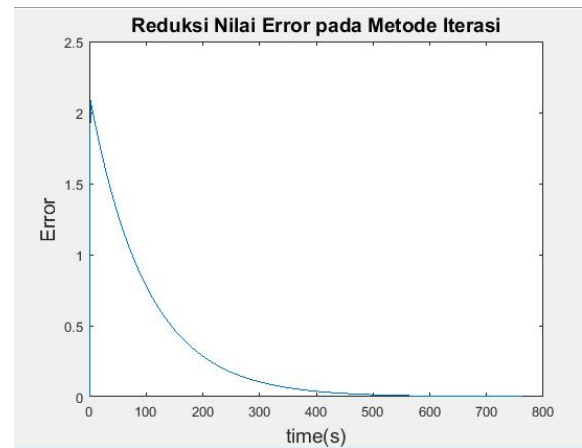
Tabel 5.1 HasilPengujianLengan 3 DOF

Sudut Target(°)			SudutUkur(°)			Error(%)		
q1	q2	q3	q1	q2	q3	q1	q2	q3
30.0	30.0	30.0	29.9	30.1	29.9	0.2	0.3	0.3
40.0	30.0	30.0	39.9	30.1	29.9	0.2	0.4	0.3
50.0	30.0	30.0	49.9	30.1	29.9	0.2	0.5	0.4
60.0	30.0	30.0	59.9	30.2	29.9	0.2	0.6	0.4
70.0	30.0	30.0	69.9	30.2	29.9	0.1	0.6	0.5
80.0	30.0	30.0	79.9	30.2	29.9	0.1	0.6	0.5
90.0	30.0	30.0	89.9	30.2	29.9	0.1	0.6	0.5
30.0	40.0	30.0	29.9	40.1	29.9	0.2	0.2	0.2
30.0	50.0	30.0	30.0	50.1	29.9	0.2	0.1	0.2
30.0	60.0	30.0	30.0	60.0	29.9	0.1	0.1	0.2
30.0	70.0	30.0	30.0	70.0	29.9	0.1	0.1	0.2
30.0	80.0	30.0	30.0	80.0	29.9	0.1	0.0	0.2
30.0	90.0	30.0	30.0	90.0	29.9	0.1	0.0	0.2
30.0	30.0	40.0	29.9	30.1	39.9	0.2	0.3	0.2
30.0	30.0	50.0	29.9	30.1	49.9	0.2	0.2	0.1
30.0	30.0	60.0	30.0	30.1	59.9	0.2	0.2	0.1
30.0	30.0	70.0	30.0	30.1	69.9	0.2	0.2	0.1
30.0	30.0	80.0	30.0	30.1	79.9	0.1	0.2	0.1
30.0	30.0	90.0	30.0	30.1	89.9	0.1	0.2	0.1
Rata-rata						0.2	0.3	0.3

Tabel 5.2 Nilai error posisi *end effector*

Posisi Target(cm)		PosisiUkur(cm)		Error(%)	
x	y	x	y	x	y
35.4	34.8	35.4	34.8	0.03	0.05
28.8	40.4	28.8	40.4	0.03	0.04
21.4	44.8	21.4	44.8	0.05	0.06
13.3	47.8	13.3	47.8	0.08	0.04
4.8	49.4	4.8	49.4	0.00	0.04

-3.9	49.5	-3.9	49.4	0.26	0.02
-12.4	48.0	-12.4	48.0	0.00	0.08
31.5	36.6	31.5	36.5	0.03	0.05
27.4	37.6	27.4	37.6	0.04	0.03
23.1	37.9	23.1	37.9	0.04	0.05
18.9	37.5	18.9	37.5	0.05	0.03
14.8	36.4	14.8	36.4	0.07	0.03
11.0	34.5	11.0	34.5	0.09	0.06
35.3	34.8	35.3	34.8	0.00	0.03
35.2	34.8	35.2	34.7	0.03	0.06
35.1	34.7	35.1	34.7	0.03	0.06
35.1	34.7	35.1	34.7	0.03	0.06
35.0	34.6	35.0	34.6	0.00	0.03
35.0	34.5	35.0	34.5	0.00	0.03
Rata-rata				0.04	0.04



Gambar 3.5 Hasil darimetodeiterasi

5. Kesimpulan dan Saran

Pada sistem lengan robot 3 dof ini dapat diketahui bahwa dengan metode iterasi, pergerakan end effector menjadi lebih halus karena pengurangan nilai error dilakukan secara eksponensial. Hal tersebut dapat dilihat pada gambar

Pada pengujian metode iterasi pada *inverse kinematic* dapat dilihat error posisi pada sumbu x yaitu sebesar 0.01 cm dan pada sumbu y yaitu sebesar 0.02 cm. Serta rata-rata error posisi sudut motor servo 1 yaitu sebesar 0.06, sudut motor servo 2 yaitu sebesar 0.09, dan sudut motor servo 3 yaitu sebesar 0.09.

Penelitian ini dapat di tingkatkan lagi dengan menambahkan pengolahan citra sebagai feedback terhadap lengan. Namun, disarankan untuk menggunakan motor servo yang

tinggi dan pembuatan mekanikalat dengan mempertimbangkan kekuatan bahan. Serta mengembangkandengan menggunakan metode pengendalian yang lebih baik.

Daftar Pustaka:

- Irmansyah, Muhammad & Antonifa, Yul. 2013. Pintu Portal Otomatis Berbasis Teknologi Programmable Logic Device (PLD). Skripsi. Jurusan Teknik Elektro, Politeknik Negeri Padang.
- Pambudi, Wahyu S. & Sumanang, Nona. 2014. *Implementasi Fuzzy-PD untuk Menentukan Posisi Obyek pada Model Simulasi Robot Arm Manipulator 3 Dof (Degree Of Freedom) Dalam Bidang 2 Dimensi*. Jurnal Ilmiah Mikrotek Vol. 1, No. 2. Jurusan Teknik Elektro, Fakultas Teknologi Industri, Universitas Internasional Batam.
- Nugraha, Deny Wiria. 2011. *Pengendali Robot Yang Memiliki Lima Derajat Kebebasan*. Jurnal Ilmiah Foristek Vol. 1, No. 1. Jurusan Teknik Elektro UNTAD Palu.
- Karseno, Doni. 2011. *Sistem Pengamanan Rumah Dengan Security Password Menggunakan Remote Berbasis Mikrokontroler Arduino*. Sekolah Tinggi Manajemen Informatika dan Komputer Amikom Yogyakarta.
- Sujarwata. 2013. *Pengendali Motor Servo Berbasis Mikrokontroler Basic Stamp 2SX Untuk Mengembangkan Sistem Robotika*. Jurnal Angkasa. Fakultas Matematika dan Ilmu Pengetahuan Alam. Universitas Negeri Semarang.
- Zein, Ahmad & Rosady, Nur Fadly. 2013. *Autonomous Boat Single Hull Dengan Sensor Kompas Dan Ultrasonic*. Politeknik Negeri Malang.