

OPTIMASI PERGERAKAN ROBOT PLANAR 3 SENDI PADA ROBOT PENGGEGAM MENGGUNAKAN METODE PEMROGRAMAN GENETIKA

Endah S. Ningrum, Bambang Sumantri, Ardik Wijayanto, Choirul Yanuar, Moch.
Iskandar Riansyah
Politeknik Elektronika Negeri Surabaya
endah,@eepis-its.edu

Abstrak

Pengembangan robot yang mengadopsi sistem pergerakan bagian tubuh manusia semakin pesat saat ini. Pada paper ini dijelaskan tentang pengembangan robot penggenggam (gripper) yang meniru prinsip jari telunjuk dan ibu jari manusia. Ibu jari memiliki dua sendi sedangkan jari telunjuk memiliki tiga sendi. Jari-jari ini digerakkan dengan menggunakan motor Direct Current (DC) yang terhubung langsung pada masing-masing sendi. Dengan menggunakan persamaan kinematika terbalik, maka sudut pergerakan masing-masing sendi dapat ditentukan untuk mengarahkan end-effectors pada posisi tertentu. Permasalahan yang muncul adalah akan dihasilkan beberapa nilai sudut pada sendi-sendi jari telunjuk yang memiliki tiga sendi. Pada paper ini akan digunakan metode pemrograman genetika untuk menentukan nilai sudut yang paling optimal dari masing-masing sendi pada jari telunjuk. Kriteria optimal adalah total sudut terkecil (ϕ_e) untuk posisi koordinat end-effector yang diinginkan. Dengan pemrograman genetika diperoleh persamaan untuk mendapatkan ϕ_e paling optimal pada posisi koordinat end-effector yang diinginkan.

Kata kunci: *gripper*, pemrograman genetika, kinematika terbalik.

PENDAHULUAN

Teknologi robotika saat ini telah berkembang dengan pesat seiring dengan kemajuan ilmu pengetahuan dan teknologi. Hal ini juga dikarenakan keinginan manusia untuk melakukan sebuah pekerjaan atau kegiatan dengan mudah sehingga diperlukan otomasi dalam pekerjaan. Dalam penggunaannya, robot banyak dirancang dan digunakan khusus pada dunia industri, hiburan, bidang kesehatan dan lain sebagainya. Pembuatan robot dapat diinspirasi dari fenomena makhluk hidup diantaranya mengacu pada prinsip dasar pergerakan tubuh manusia. Misalkan cara manusia berjalan, berbicara, memegang benda dan lain-lain.

Jari adalah salah satu bagian dari tubuh manusia yang mempunyai peranan penting dalam pergerakan manusia untuk melakukan berbagai aktivitas. Manusia mempunyai total sepuluh jari yang berfungsi untuk memegang benda. Dari prinsip kerja jari tangan manusia itulah kemudian digunakan sebagai dasar pengembangan robot jari yang dirancang untuk memegang benda dengan berbagai jenis desain dan metode.

Pada penelitian Shirai dkk, Takaki dkk dan Yamano dkk telah dikembangkan robot jari yang mengadaptasi prinsip kerja dari jari tangan manusia. Mekanisme yang digunakan pada robot ini adalah motor-motor penggerak sendi-sendi jari yang diletakkan terpusat. Untuk menggerakkan masing-masing sendi, motor dihubungkan dengan sendi melalui benang-benang penghubung. Elastisitas benang penghubung akan mempengaruhi kekuatan pencengkeraman robot terhadap objek yang digenggam.

Jika sendi-sendi yang dikendalikan semakin banyak (DoF semakin banyak) maka konfigurasi benang-benang penghubung ini akan semakin kompleks. Untuk meningkatkan kecepatan pergerakan masing-masing sendi, desain robot dapat menggunakan sistem penggerak yang terhubung langsung pada

masing-masing sendi (Namiki, 2003; Kawasaki, 1999; Butterfass, 2001).

Masing-masing sendi digerakkan dengan sebuah motor DC berukuran kecil. Mekanisme ini dimungkinkan setiap sendi untuk bergerak dengan lebih bebas sehingga setiap jari dapat bergerak dengan lebih lentur. Pada paper ini akan dijelaskan tentang perancangan robot penggenggam dengan meniru mekanisme jari telunjuk dan ibu jari tangan manusia. Jari telunjuk dirancang dengan jumlah sendi sebanyak tiga buah sehingga memiliki tiga DoF, sedangkan ibu jari dirancang dengan jumlah sendi sebanyak dua buah. Masing-masing sendi pada robot ini digerakkan dengan sebuah motor DC yang berukuran kecil dan terhubung secara langsung. Kedua jari ini akan bekerja bersama untuk menggenggam sebuah benda dan akan bergerak secara bersama-sama. Di dalam paper ini hanya akan dibahas tentang aspek kinematik untuk pergerakan kedua jari tersebut.

DASAR TEORI

Sistem Mekanik dan Kinematika Robot

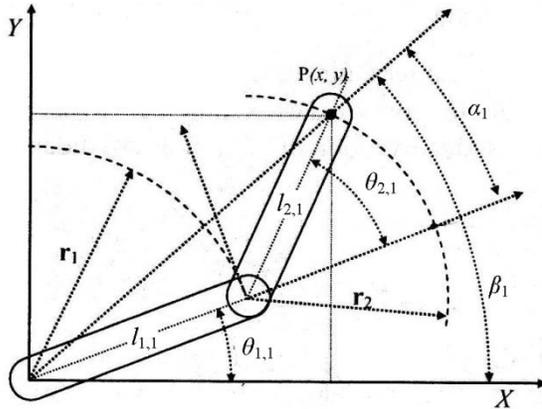
Ide awal dari pembuatan robot *gripper* ini adalah meniru mekanisme ibu jari dan jari telunjuk ketika menggenggam sebuah benda serta memiliki kemampuan untuk menggerakkannya. Sistem mekanik robot *gripper* ini ditunjukkan pada Gambar 1.



Gambar 1. Sistem mekanik robot Gripper

Dari Gambar 1 dapat diketahui bahwa jari ke-1(sebelah kiri) memiliki dua sendi (2 DoF) yang menirukan prinsip ibu jari, sedangkan jari ke-2 memiliki tiga sendi (3 DoF) menirukan prinsip jari telunjuk. Sistem kinematika kedua jari merupakan sistem kinematika robot planar untuk dua sendi dan tiga sendi yang akan dijelaskan sebagai berikut:

Kinematika Robot 2 DoF



Gambar 2. Konfigurasi robot planar 2 sendi

Jika $l_{1,1}$ dan $l_{2,1}$ adalah panjang dari lengan satu dan lengan dua, $\theta_{1,1}$ dan $\theta_{2,1}$ adalah sudut dari lengan satu dan lengan dua, maka *invers* kinematik dari robot planar dua DoF adalah (Pituwarno, 2006).

$$\theta_{2,1} = \arccos\left(\frac{x^2 + y^2 - l_{1,1}^2 - l_{2,1}^2}{2l_{1,1}l_{2,1}}\right) \quad (1)$$

$$\theta_{1,1} = \beta_1 - \alpha_1 \quad (2)$$

$$\beta_1 = \arctan\left(\frac{y}{x}\right) \quad (3)$$

$$\alpha_1 = \arctan\left(\frac{l_{2,1} \sin \theta_{2,1}}{l_{2,1} \cos \theta_{2,1} + l_{1,1}}\right) \quad (4)$$

Dimana x dan y adalah posisi dari ujung lengan dua (*end effector*). Jika nilai x dan y diketahui maka sudut pergerakan masing-masing sendi akan dapat ditentukan dengan tepat.

Kinematika Robot Tiga DoF

Konfigurasi robot planar dengan tiga DoF dapat ditunjukkan pada Gambar 3. Nilai $\theta_{1,2}$, $\theta_{2,2}$, dan $\theta_{3,2}$, yang merupakan sudut sendi satu, dua, dan tiga, dapat dihitung dengan menggunakan kinematika terbalik jika data koordinat *end-effector* x_e , y_e , dan ϕ_e diketahui. Persamaan kinematika terbalik untuk robot planar tiga DoF adalah (Asada, 2005).

$$\theta_{3,2} = \phi_e - \theta_{1,2} - \theta_{2,2} \quad (5)$$

$$\theta_{2,2} = \pi - \beta_2 \quad (6)$$

$$\beta_2 = \arccos\left(\frac{l_{1,2}^2 + l_{2,2}^2 - r^2}{2l_{1,2}l_{2,2}}\right) \quad (7)$$

$$\theta_{1,2} = \alpha - \gamma \quad (8)$$

$$\alpha = \arctan\left(\frac{y_w}{x_w}\right) \quad (9)$$

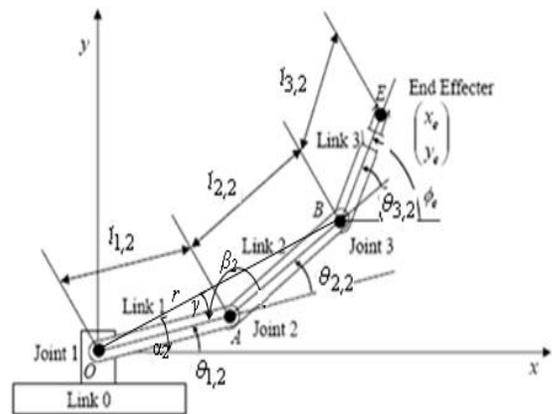
$$x_w = x_e - l_{3,2} \cos \phi_e \quad (10)$$

$$y_w = y_e - l_{3,2} \sin \phi_e \quad (11)$$

$$\gamma = \arccos\left(\frac{r^2 + l_1^2 - l_2^2}{2l_1 r}\right) \quad (12)$$

$$r = \sqrt{x_w^2 + y_w^2} \quad (13)$$

Dimana x_w dan y_w adalah koordinat titik B(wrist), sedangkan $l_{1,2}$, $l_{2,2}$, dan $l_{3,2}$ adalah panjang lengan satu, lengan dua, dan lengan tiga.

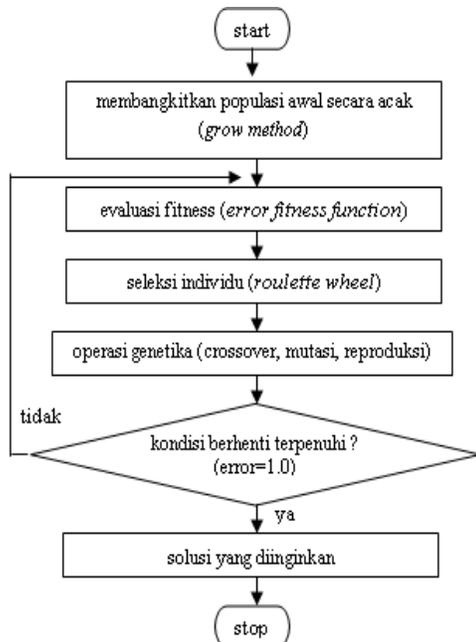


Gambar 3. Konfigurasi robot planar 3 sendi

Pada robot planar tiga DoF, untuk koordinat *end-effector* (x_e, y_e) yang sama bisa dicapai dengan nilai ϕ_e yang berbeda-beda disebut dengan istilah *redundant*. Oleh sebab itu, untuk menempatkan *end-effector* pada koordinat tertentu maka nilai ϕ_e harus ditentukan terlebih dahulu. Proses penentuan nilai ϕ_e akan dilakukan melalui proses optimasi dari nilai-nilai yang mungkin didapat dengan menggunakan metode Pemrograman Genetika (*Genetic Programming*). Nilai ϕ_e yang paling optimal adalah nilai yang terkecil.

Optimasi ϕ_e Pemrograman Genetika

Pemrograman untuk genetika merupakan algoritma pemrograman mengambil konsep dari proses genetika pada makhluk hidup. Pada Algoritma Genetika, tiap-tiap individu calon solusi tidak berupa untaian nilai, tetapi berupa sebuah program komputer yang digambarkan sebagai sebuah pohon berakar dimana setiap simpul adalah fungsi atau parameter fungsi tersebut. Urutan proses yang dilakukan pemrograman genetika pada *paper* ini ditunjukkan pada Gambar 4.



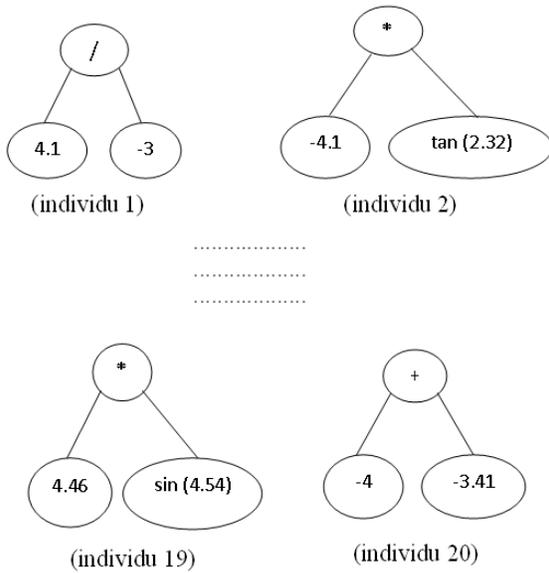
Gambar 4. Alur proses pemrograman genetika

Membangkitkan Populasi Awal

Populasi merupakan kumpulan dari beberapa individu yang merupakan penyusun rangkaian solusi yang diharapkan. Di dalam metode pemrograman genetika, individu terdiri dari terminal dan fungsi. Pada *paper* ini, individu dibangkitkan secara acak dari terminal x, y , dan R yang merupakan koordinat (x, y) dari *end-effector* robot tiga DoF yang diinginkan, sedang R adalah suatu konstanta yang dibangkitkan secara acak pada nilai antara -5 s/d 5. Fungsi yang digunakan untuk membangkitkan populasi awal adalah fungsi awal $\{+, -, /, *, \text{power}, \sin, \cos, \tan\}$. Pada *paper* ini, populasi awal dibangkitkan dengan metode tumbuh (*grow method*) dengan jumlah individu sebanyak 20. Dengan metode ini didapat populasi awal sebagai berikut :

1. $(4.1 / (-3))$;
2. $(-4.1 * \tan (2.32))$;
3. $(-1.2 * \tan (3.25))$;
4. $(-3.39 * \sin (-3.67))$;
5. $(-3.18 * \cos (-0.64))$;
6. $(-4.21 * (-3.07))$;
7. $\text{pow} (-0.29, -2.085)$;
8. $(4.56 / -3.77)$;
9. $(4.31 * \cos (2.9))$;
10. $\text{pow} (0.63, (0.22 * \tan (-4.79))$;
11. $(-1.55 + 3.09)$;
12. $(1.75 * \tan (4.08))$;
13. $(4.98 - 1.07)$;
14. $(0.13 - (-3.82))$;
15. $(3.6 * \cos (-3.46))$;
16. $(1.88 * \cos (0.95))$;
17. $(2.25 * 0.04)$;
18. $(2.61 * \tan (-4.2))$;
19. $(4.46 * \sin (4.545))$;
20. $(-4 + (-3.41))$;

Populasi awal jika direpresentasikan ke dalam *tree structure*, akan terlihat seperti Gambar 5.



Gambar 5. Representasi individu-individu dalam populasi awal dengan *tree structure*

Evaluasi *Fitness*

Evaluasi *fitness* pada pemrograman genetika bertujuan untuk memberikan umpan balik (*feedback*) ke algoritma yang telah dirancang mengenai pemilihan individu yang mempunyai probabilitas tinggi untuk dapat melakukan operasi genetika seperti *crossover*, reproduksi dan mutasi serta pemilihan individu yang mempunyai probabilitas rendah untuk dihapus dari populasi. Jenis *fitness* yang digunakan dalam *paper* ini adalah *error fitness function*. dengan persamaan 14.

$$f(p) = \sum_{i=1}^n |P_i - O_i| \tag{14}$$

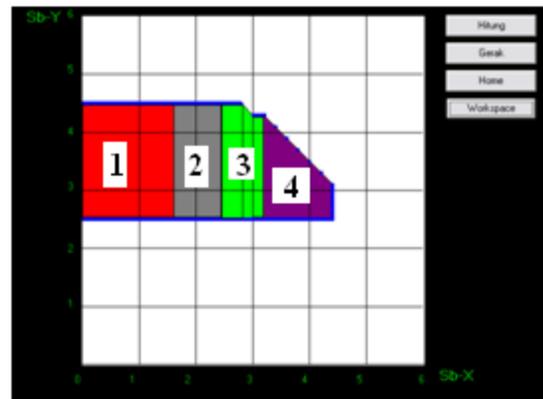
dimana:

P_i = nilai *output* pada program

O_i = nilai *output* yang dihasilkan dari data pembelajaran

Pada *paper* ini, pemrograman genetika digunakan untuk mendapatkan persamaan kinematika terbalik dari pergerakan robot jari tiga DoF. Untuk mempermudah proses pe-

mograman genetika, maka *workspace* dari pergerakan tiga DoF dibagi menjadi empat zona seperti ditunjukkan pada Gambar 6. Sehingga nantinya akan didapatkan empat buah persamaan kinematika terbalik dari pergerakan jari ke-2. O_i diperoleh dari percobaan dan ditunjukkan pada Tabel 1 sampai dengan 4, yang merupakan ϕ_e yang terkecil pada beberapa posisi koordinat (x_e, y_e) .



Gambar 6. Pembagian *workspace* jari ke-2 untuk proses pembelajaran.

Tabel 1. Data *training* untuk $x=0$ s/d 1.6 (zona satu)

data training	sumbu x	sumbu y	ϕ_e optimum (derajat)
1	0	2.5	186
2	0	3.5	170
3	0.2	2.7	183
4	0.2	3.7	167
5	0.4	2.9	181
6	0.6	3.1	178
7	0.6	4.1	162
8	0.8	3.3	175
9	0.8	4.3	159
10	1	2.5	187
11	1.2	3.7	168
12	1.4	2.9	180
13	1.6	3.5	184
14	1.6	4.5	155

Sumber : data olahan

Tabel 2. Data *training* untuk x=1.7 s/d 2.4 (zona dua)

data <i>training</i>	sumbu x	sumbu y	ϕ_e optimum (derajat)
1	1.8	2.5	183
2	1.8	3.5	169
3	1.8	4.5	154
4	2	2.7	178
5	2	3.7	162
6	2	4.5	151
7	2.2	2.5	177
8	2.2	2.9	172
9	2.2	3.9	160
10	2.4	3.1	169
11	2.4	4.1	157

Sumber : data olahan

Tabel 3 Data *training* untuk x=2.5 s/d 3.2 (zona tiga)

data <i>training</i>	sumbu x	sumbu y	ϕ_e optimum (derajat)
1	2.8	2.7	169
2	2.8	3.9	156
3	3	2.9	164
4	3	3.7	156
5	3.2	2.5	201
6	3.2	2.7	163
7	3.2	4.1	149
8	2.6	2.5	175
9	2.6	3.3	165
10	2.6	3.9	162

Sumber : data olahan

Tabel 4. Data *training* untuk x=3.3 s/d 4.4 (zona empat)

data <i>training</i>	sumbu x	sumbu y	ϕ_e optimum (derajat)
1	3.6	2.5	208
2	3.6	3.3	187
3	3.6	3.5	150
4	3.6	3.9	147
5	3.8	2.5	211
6	3.8	3.3	188
7	3.8	3.5	184
8	4	2.5	213
9	4	2.9	202
10	4.2	2.5	215
11	4.2	3.1	198
12	4.4	2.7	210

Sumber : data olahan

Dengan menggunakan data-data pada tabel 1 - 4 serta dengan menentukan *fitness case* = 11, proses pembelajaran yang telah dilakukan menghasilkan model persamaan kinematika terbalik untuk menghitung ϕ_e terkecil pada masing-masing zona, yaitu:

Zona Satu: (15)

$$\phi_e = (0.25 + ((190.3 * \text{Cos}((1.87 * y) - ((x * 8) * \text{Cos}(2.61 * \text{Tan}((y ^ 4)))) * \text{Cos}((((0.2 + y) * \text{Sin}((6.36 * ((y * (-4) + (-2.04)) * \text{Cos}(((4 + (x ^ 1.3)) - 3.6) + (-1.97)) * (((7 * \text{Cos}(\pi)) * x) - ((y * \text{Cos}(4.1 / x)) - x) * (-5.99 * \text{Cos}(x)))) * \text{Cos}(9) / (-0.49)) * y))))))$$

Zona Dua: (16)

$$\phi_e = (((((2.32 + ((179.2 - ((x * y) * ((y - (-0.58)) * \text{Cos}(((y + (((x * \text{Cos}(x)) * \text{Cos}(5.5) * \text{Cos}(-13.6) ^ y)) + (4.95 * (((12.7 * \text{Cos}(-8.35) - (y ^ 8.19)))) * \text{Sin}(-23) / y))) * \pi / 180)))) - (-10.11))) + \pi) * \text{Cos}(1.62 * x)) * \text{Cos}(\pi)))$$

Zona Tiga: (17)

$$\phi_e = ((5 + (6 + (179 - ((3.24 * (((y * \text{Cos}(6 + (2.79 ^ (x * (x * \text{Cos}((2.3 ^ y) * (-7.98)))))) * \text{Cos}(2.03) * 1.04 * \text{Cos}(((y + x) - (-1.46)) * \text{Cos}((y ^ ((0.47 * x) * x)))) + 8.58)) * x) - 3)))) * \text{Cos}(2)$$

Zona Empat: (18)

$$\phi_e = (((((200.42 * \text{Cos}(x + ((-1.4 * \text{Cos}((-6.5) + (((y * \text{Cos}(2) - 0.78) * \text{Tan}((3.96 * ((y * x) * \text{Cos}(-4.26)) * 204.2)))) - ((x * y) * (y - 7.4))) * \text{Cos}(y * 12.47)) * \text{Cos}((-1.32 * ((y - 1.69) ^ y)) + (-4 / (-1.76)))) * \text{Cos}(y * \text{Tan}((2.65 ^ x))))))$$

HASIL PENGUJIAN

Dengan menggunakan model persamaan 16-18, pengujian telah dilakukan untuk mengetahui akurasi dari model yang telah diperoleh. Data hasil pengujian diperlihatkan pada Tabel 5.

Dari data pengujian pada Tabel 5 terlihat bahwa model persamaan yang diperoleh melalui algoritma pemrograman genetika dapat digunakan untuk menghasilkan trajektori pergerakan robot planar tiga DoF pada robot jari dengan error rata-rata 1.62%.

Tabel 5. Data pengujian nilai *error* titik target

Koordinat set poin		Koordinat output		% error	
x	y	x	y	x	y
3	3.5	2.97	3.4	1.12	2.67
3	4.1	2.98	4.08	0.5	0.46
3.6	2.7	3.63	2.61	0.99	3.02
4	3.5	4.04	3.58	1.03	2.5
2.8	2.5	2.81	2.65	0.44	6.2
2.4	3.5	2.38	3.48	0.4	0.3
2	3.3	2	3.2	0	2.77
1.6	3.7	1.58	3.6	1.06	2.47
1	2.5	0.99	2.52	0.59	0.82
1	3.9	0.96	3.8	4.1	2.5
2.2	2.9	2.2	2.8	0.11	2.98
4.2	3.3	4.18	3.22	0.3	2.24
4.4	2.9	4.39	2.78	0.16	4.09
1.6	5	1.64	4.99	2.89	0.2
1.6	5.3	1.56	5.24	2	1.2
2	5.2	2.05	5.2	2.77	0.6
2	4.7	1.99	4.7	1.6	0.25
2.3	4.8	2.32	4.85	1.01	1.06
3	4.9	3.06	4.97	0.6	1.46
3	5.4	2.9	5.3	3	1.07
1.4	4.8	1.46	4.8	4.6	0.6
4	3.9	3.9	3.78	1.2	2.4
3.1	5.2	3.1	5.2	1.8	0.5

Sumber : data olahan

KESIMPULAN

Hasil dari pengujian yang telah dilakukan, pemrograman genetika dapat digunakan untuk menghasilkan model persamaan untuk mencari nilai sudut total masing-masing sendi (ϕ_e) yang paling optimal. Dari percobaan, diketahui bahwa rata-rata *error* yang dihasilkan untuk menempatkan *end-effector* sendi

tiga DoF pada koordinat yang diinginkan adalah 1.62% dengan *error* terbesar adalah 4.6%.

DAFTAR PUSTAKA

- Asada. H.H (2005), *Introduction to Robotics : Chapter 4 Planar Kinematics*. Massachusetts: Massachusetts Institute of Technology.
- Butterfass, M.J., Grebenstein, H. Liu, and G. Hirzinger, (2001) "DLR-Hand II: Next Generation of a Dextrous Robot Hand," presented at IEEE International Conference on Robotics and Automation.
- Kawasaki, T.H., Komatsu, K. Uchiyama, and T. Kurimoto, (1999) "Dexterous Anthropomorphic Robot Hand With Distributed Tactile Sensor: Gifu Hand II," presented at IEEE International Conference on Systems, Man, and Cybernetics.
- Pitowarno.E (2006) *Robotika: Disain, Kontrol, dan Kecerdasan Buatan*: Andi Yogyakarta.
- T. Shirai, M. Kaneko, K. Harada, and T. Tsuji, (1999) "Enveloping Grasp with a New Detaching Strategy," presented at the Ninth International Conference on Advanced Robotics.
- T. Takaki and T. Omata, (2009) "High Performance Anthropomorphic Robot Hand with Grasp Force Magnification Mechanism," presented at IEEE International Conference on Robotics and Automation, Kobe, Japan
- Yamano. I and T. Maeno, (2005) "Five-fingered Robot Hand using Ultrasonic Motors and Elastic Elements," presented at IEEE International Conference on Robotics and Automation, Barcelona, Spain.