

## BAB 4

### HASIL DAN PEMBAHASAN

#### 4.1 Pengujian Motor Servo

Untuk dapat menggunakan motor servo sesuai perhitungan yang diinginkan, servo perlu dikalibrasi. kalibrasi untuk setiap servo diperlukan karena servo memiliki perbedaan karakteristik meskipun memiliki tipe yang sama. Pada umumnya motor servo dapat berputar dari sudut  $0^{\circ}$ - $180^{\circ}$ . Metode kalibrasi yang digunakan adalah dengan memberikan masukan sudut minimal hingga sudut maksimal pada setiap *joint* servo hingga mencapai batas. Ketika servo diberikan masukan yang melebihi batas operasinya maka servo akan bergetar. Hal tersebut dapat mengakibatkan kerusakan pada motor servo. Sudut servo kemudian dicek menggunakan busur derajat. Hasil pengukuran kemudian dibandingkan dengan masukan servo agar mendapatkan hasil yang sesuai. Dari hasil pengalibrasian didapat bahwa servo memiliki sudut opsional yang berbeda-beda. Hal tersebut disebabkan oleh karakteristik tiap servo yang berbeda, contohnya untuk servo *base* menggunakan tipe (*Tower Pro MG995*) dan servo pergelangan tangan menggunakan tipe (*Tower Pro MG90*). Setiap servo memiliki repon yang berbeda-beda untuk input yang sama. Pada proses kalibrasi terdapat karakteristik servo yang berbeda-beda, hal tersebut disebabkan oleh keterbatasan torsi dari servo.

Tabel 4.1 Karakteristik Sudut Servo

	Sudut Servo Aktual		Sudut Model Robot	
	Min	Max	Min	Max
<i>base</i>	0	169	0	180
Bahu	0	169	0	180
Sikut	0	169	0	180
<i>Wrist</i>	20	169	0	180

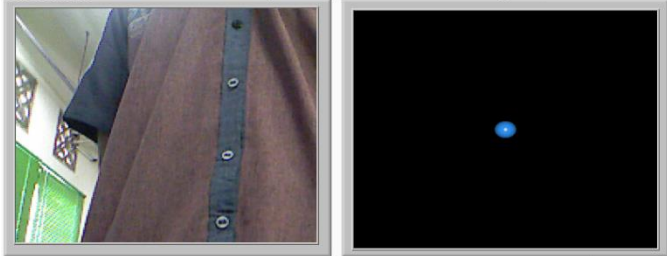
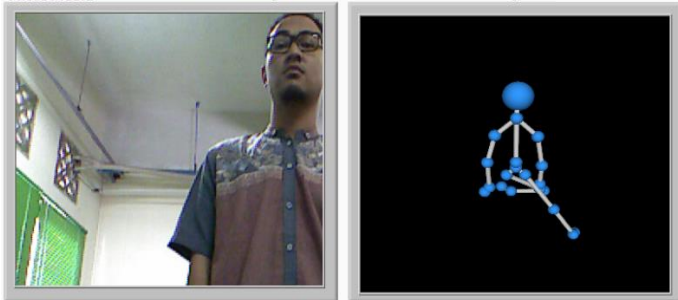
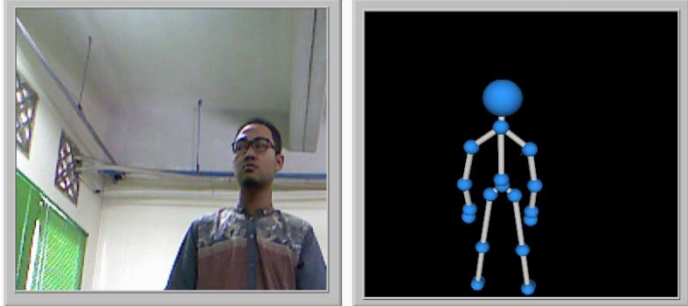
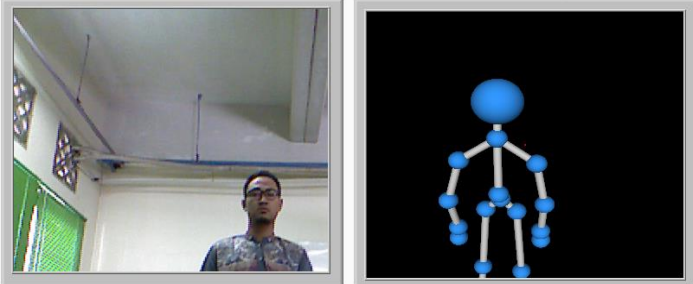
#### 4.2 Pengujian Jarak Kinect

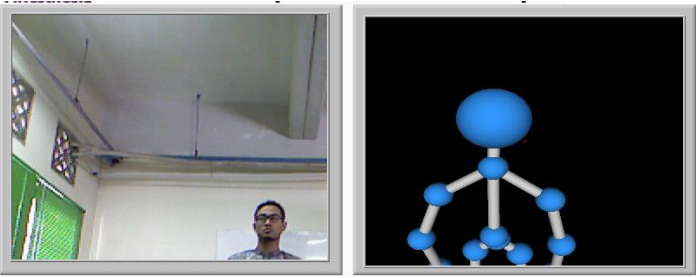
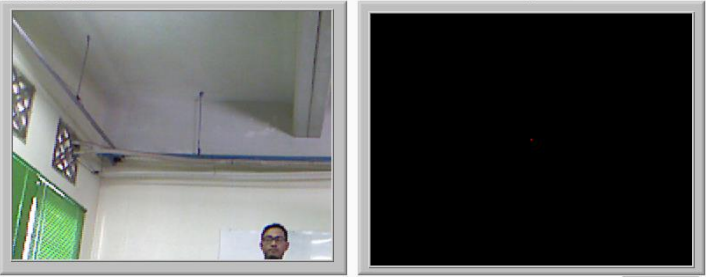
Jarak antara pengguna dan Kinect memiliki pengaruh yang besar terhadap pendeteksian *skeleton tracking*. Pengendalian robot dapat dipengaruhi dengan jarak antara pengguna dan Kinect dan mempengaruhi kineja dari *skeleton tracking*. Kinect tidak akan bisa mendeteksi pengguna jika

pengguna berada terlalu dekat dengan Kinect atau terlalu jauh dari Kinect. Seluruh pengendalian dapat dipengaruhi oleh jarak antara pengguna dan Kinect.

Pengujian dilakukan dengan jarak yang berbeda antara pengguna dan Kinect. Pengukuran digunakan menggunakan meteran dengan ketelitian meter.

Tabel 4.2 Hasil Pengujian Jarak Ideal

No.	Jarak	Satuan	Kamera Kinect dan GUI	Keterangan
1.	0.5	m		<i>Skeleton</i> tidak terdeteksi
2.	1	m		<i>Skeleton</i> terdeteksi tetapi masih banyak <i>error</i> terutama pada bagian kaki
3.	1.5	m		<i>Skeleton</i> terdeteksi secara keseluruhan dan tidak terdapat <i>error</i> .
4.	2	m		<i>Skeleton</i> terdeteksi tetapi bagian bawah lutu tidak terdeteksi

No.	Jarak	Satuan	Kamera Kinect dan GUI	Keterangan
5.	2.5	m		<i>Skeleton</i> terdeteksi hanya setengah badan.
6.	3	m		<i>Skeleton</i> tidak terdeteksi.

Tabel 4.2 menjelaskan hasil pengujian jarak yang ideal untuk digunakan dalam pengendalian robot. Percobaan pengujian dilakukan 3 kali untuk masing-masing jarak. Kinect dapat mendeteksi mulai dari jarak 1 m sampai dengan 2.5 m. Setelah melakukan beberapa percobaan, jarak yang ideal digunakan adalah antara 1.5 m sampai 2 m. Pengujian gestur lengan kanan dapat dilakukan pada 1.5 m – 2.5 m karena jika kurang dari jarak tersebut Kinect tidak bisa mendeteksi kerangka tubuh. gestur lengan kanan dilakukan secara acak dan tidak melakukan gestur tertentu. Gestur dilakukan untuk menguji Kinect dalam mendeteksi kerangka tubuh.

Tabel 4.3 Hasil Pengujian Gestur

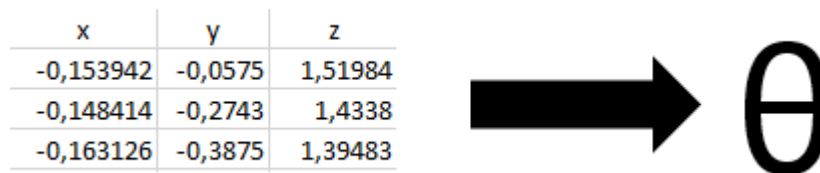
Jarak	Gestur Lengan Kanan
0.5	Gagal
1	Gagal
1.5	Berhasil
2	Berhasil
2.5	Gagal
3	Gagal

Tabel 4.3 merupakan hasil dari pengujian jarak anatar pengguna dan Kinect. Pada jarak 0.5 m gestur lengan kanan tidak dapat terdeteksi sehingga Kinect gagal dalam menangkap gestur

tersebut. Jarak 1 m gestur dapat terdeteksi tetapi masih memiliki *error* yang banyak dan terdapat *noise* yang cukup banyak. Jarak 1.5 m merupakan jarak yang ideal dikarenakan gestur dapat ditangkap dan memiliki *error* dan *noise* yang sedikit. Dari tabel 4.3 dapat diketahui bahwa pengendalian yang paling baik dapat dilakukan pada jarak 1.5 m. Jarak 2 m dan 3 m pembacaan gestur mengalami *error* atau tidak terbaca dari jangkauan *frame* sehingga dapat disimpulkan bahwa pada jarak tersebut tidak berhasil.

### 4.3 Pengelolaan Data Kinect

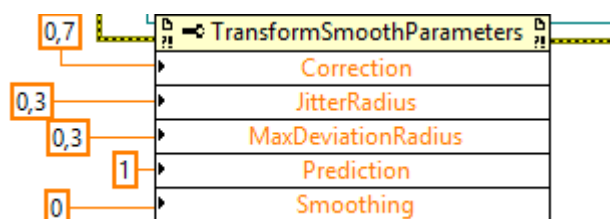
Data mentah dari Kinect dapat dikelola untuk mendapatkan sudut robot. lengan akan digunakan untuk mengendalikan robot. *joint* bahu, siku, dan pergelangan tangan memiliki data yang berbeda-beda. Setiap *joint* memiliki koordinat  $x$ ,  $y$ ,  $z$  dan setiap koordinat memiliki nilai yang bervariasi.



Gambar 4.1. Merubah data mentah ke sudut

Gambar 4.1 menunjukkan data mentah Kinect dirubah menjadi sudut. Data yang digunakan adalah lengan kanan yang memiliki koordinat  $x$ ,  $y$ ,  $z$ . Koordinat tersebut dikelola untuk dijadikan sudut dan mengendalikan robot sesuai pergerakan lengan kanan. Data koordinat  $x$ ,  $y$ ,  $z$  harus dipisahkan terlebih dahulu karena data koordinat setiap *joint* menjadi satu data. Setelah data dipisahkan, persamaan 3.1 dapat diterapkan pada setiap koordiant *joint*.

Kerangka yang dideteksi oleh Kinect terkadang masih kasar. Program Kinect masih harus dimodifikasi untuk dapat mendeteksi kerangka tubuh. Metode yang digunakan untuk dapat memperhalus pendeteksian kerangka tubuh adalah *Transform smoothing*. Metode tersebut memiliki beberapa parameter yang dapat membantu untuk memperhalus pendeteksian kerangka.



Gambar 4.2 *Smoothing Parameter*

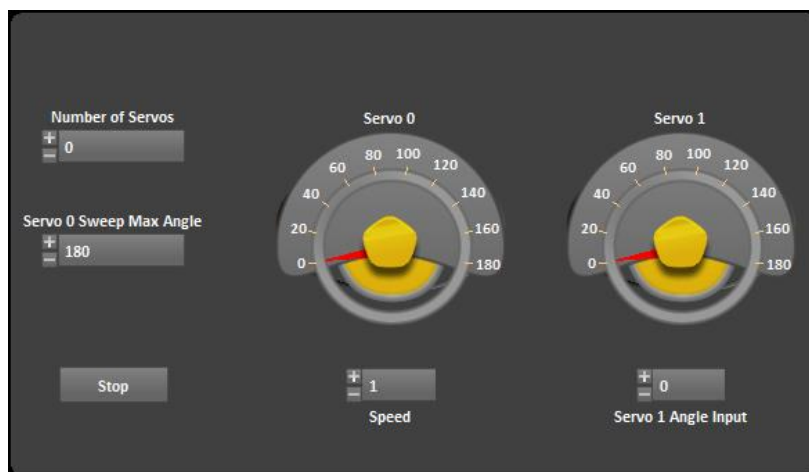
Setiap parameter pada gambar 4.2 memiliki fungsinya masing-masing. Penjelasan terkait setiap parameter dapat dilihat pada tabel 4.4.

Tabel 4.4 Penjelasan Parameter *Transform smoothing*

No.	Parameter	Deskripsi
1.	<i>Smoothing</i>	Menentukan tingkat kehalusan kerangka
2.	<i>Correction</i>	Mengurangi <i>error</i> dalam mendeteksi kerangka
3.	<i>Prediction</i>	Menentukan berapa <i>frame</i> yang diprediksi
4.	<i>Jitter Radius</i>	Menentukan seberapa jauh Kinect dapat mendeteksi kerangka
5.	<i>Max deviation Radius</i>	Menentukan jangkauan Kinect untuk mendeteksi kerangka

#### 4.4 Pengujian Komunikasi Serial

Pengujian dilakukan dengan mengendalikan robot menggunakan Arduino. Pengujian dilakukan dengan memberikan masukan sudut *joint* bahu, siku, pergelangan tangan. Pada pengujian tersebut didapatkan hasil yang sesuai dengan masukan.



Gambar 4.3 Pengujian Komunikasi Serial

Pada pengujian yang kedua digunakan protokol komunikasi LIFA untuk menangani komunikasi serial pada LabVIEW. Sesuai dengan pengujian yang pertama dilakukan dengan mengirim sudut ke setiap *joint*. Software Arduino digunakan untuk mengirim data saja. Dari hasil pengujian robot dapat bergerak sesuai masukan. Tetapi terkadang robot mengalami *error* dengan tidak merespon kepada masukan yang diberikan dari LabVIEW. Kalibrasi antara Arduino dan LabVIEW harus diulang untuk mengatasi hal tersebut. Dari gambar 4.3 jumlah servo motor dapat ditentukan dengan batas sudut sesuai kebutuhan.

#### 4.5 Hasil Respon Manipulator Robot Wayang

Pengujian dilakukan untuk mendapatkan hasil yang akurat dari pengendalian manipulator Robot Wayang. Pengujian respon manipulator Robot Wayang dilakukan dengan menguji akurasi gerakan lengan kanan dengan respon manipulator Robot Wayang. Pengujian dilakukan untuk mengetahui apakah sistem pengendalian sudah sesuai perancangan yang diinginkan dan memberikan respon yang tepat. Tingkat akurasi respon manipulator Robot Wayang dilakukan dengan menghitung selisih data *error* dengan data input. Perhitungan selisih data *error* menggunakan persamaan :

$$x_{e i} = \frac{|x_t - x_i|}{x_t} \quad (4.1)$$

$$\bar{x}_e = \left( \frac{x_{e i} + x_{e i+1} \dots x_{e n}}{n} \right)$$

Keterangan:

$x_{e i}$  = Nilai error sudut pada percobaan ke-*i*

$x_t$  = Masukan sudut

$n$  = Banyak percobaan

$x_i$  = Nilai sudut riil percobaan ke-*i*

$\bar{x}_e$  = Nilai rata-rata error

Pengujian dilakukan sebanyak 5 kali untuk setiap sudut. Pengujian setiap *joint* dilakukan dengan memberikan masukan sudut 30°,45°,90°,120°.

Respon *joint base* dapat dilihat pada tabel 4.5.

Tabel 4.5 Respon *Joint Base*

Masukan Sudut	Percobaan ke-					<i>Error</i>
	1.	2.	3.	4.	5.	
30°	25	32	30	30	32	0.033%
45°	40	47	40	46	47	0.111%
90°	90	90	90	90	89	0.011%
120°	127	122	121	118	121	0.075%
Rata-Rata <i>Error</i>						0.251%

Tabel 4.5 merupakan hasil percobaan respon *joint base*. Pada tabel tersebut dapat dilihat bahwa nilai rata-rata *error* 0.251%. Nilai *error* disebabkan karena pembacaan Kinect yang kurang sempurna meskipun sudah di filter. *Error* yang lain disebabkan ketika Kinect tidak dapat membaca *joint* yang berada pada satu garis lurus. Terkadang respon dari manipulator Robot Wayang keluar

dari sudut yang diberikan. *Error* tersebut menyebabkan respon menjadi tidak sesuai. Karakteristik *joint base* bergerak secara *horizontal*. *Joint base* dapat mengikuti gestur pengguna dengan *error* yang sedikit. Respon *joint* berada pada jangkauan 25-32 masukan sudut 30° dan begitupun pada masukan sudut yang lainnya.

Respon pada *joint* bahu dapat dilihat pada tabel 4.6.

Tabel 4.6 Respon *Joint* Bahu

Masukan Sudut	Percobaan ke-					<i>Error</i>
	1.	2.	3.	4.	5.	
30°	33	31	32	27	31	0.133%
45°	45	46	44	42	45	0.066%
90°	90	87	90	90	90	0.033%
120°	122	120	119	125	127	0.108%
Rata-Rata <i>Error</i>						0.085%

Respon dari *joint* bahu memiliki *error* lebih sedikit dari *joint base*. *Error* terjadi disebabkan pembacaan Kinect yang sensitif. Pengguna memiliki kesulitan untuk tetap diam pada saat pengujian. *Joint* bahu, siku, pergelangan tangan bergerak secara vertikal. Respon *joint* berada pada jangkauan 27-33 pada masukan sudut 30°.

Respon *joint* Siku dapat dilihat pada tabel 4.7

Tabel 4.7 Respon *Joint* Siku

Masukan Sudut	Percobaan ke-					<i>Error</i>
	1.	2.	3.	4.	5.	
45°	48	47	49	46	45	0.22%
90°	90	90	90	90	90	0%
120°	123	122	122	121	120	0.066%
Rata-Rata <i>Error</i>						0.095%

*Joint* siku tidak diberikan masukan sudut 30° disebabkan pengguna memiliki kendala untuk mensimulasikan sudut tersebut. Pergerakan *joint* siku terjadi ketika siku lengan kanan pengguna bergerak dari atas kebawah atau sebaliknya. Perhitungan sudut *joint* siku menggunakan persamaan 3.1 seperti *joint* yang lain. Tetapi ada perbedaan pada *joint* siku. Hasil dari persamaan 3.1 dikurangi 180 dan di *absolute* untuk mendapatkan sudut positif. Jika tidak dilakukan pengurangan tersebut maka respon dari *joint* siku akan terbalik. Hal tersebut dikarenakan motor servo yang memiliki keterbatasan. *Joint* siku dan bahu adalah *joint* yang stabil pada saat pengujian dan tidak memiliki banyak *error*.

Respon *joint* pergelangan tangan dapat dilihat ada tabel 4.8

Tabel 4.8 Respon *Joint* Pergelangan Tangan

Masukan Sudut	Percobaan ke-					<i>Error</i>
	1.	2.	3.	4.	5.	
120°	118	119	121	117	122	0.25%
150°	153	148	152	152	153	0.053%
180°	180	180	180	180	180	0%
Rata-Rata <i>Error</i>						0.151%

Masukan sudut pada *joint* pergelangan tangan berada pada jangkauan 120°, 150°, 180°. Hal tersebut disebabkan pembacaan dari Kinect yang kurang sempurna. Pengguna tidak bisa mensimulasikan gerakan kurang dari 120° karena keterbatasan *joint* pergelangan tangan pengguna. Disisi lain perubahan pergerakan pada Kinect tidak signifikan.

#### 4.6 Akurasi Respon Manipulator Robot Wayang

Dari semua pengujian maka dapat diitung seluruh data *error* pengendalian manipulator lengan Robot Wayang sebagai berikut :

$$\text{Rata - Rata Data Error} = \frac{0.251 + 0.085 + 0.095 + 0.151}{4} = 0.089\% \quad (4.2)$$

Akurasi pergerakan manipulator Robot Wayang ditentukan apakah robot dapat mengikuti pergerakan lengan kanan yang dilakukan oleh pengguna. *Error* terjadi ketika Kinect tidak dapat mendeteksi secara sempurna *joint* lengan kanan. Terutama ketika *joint* berada pada satu garis lurus sehingga menyebabkan respon robot diluar pergerakan pengguna.