

# Prototype Model 3D Robot Lengan dan Simulasi Kendali Pergerakan pada 4-DOF Berbasis Virtual Reality

M. Fajar Wirayudha<sup>1</sup>, Giri Wahyu Wiriasto<sup>1</sup>, Paniran<sup>1</sup>

<sup>1</sup>Jurusan Teknik Elektro – Universitas Mataram, 83127 – Lombok, Indonesia

## ARTICLE INFO

### Article history :

Received August 29, 2024

Revised August 31, 2024

Accepted August 31, 2024

### Keywords :

Virtual Reality;

Prototype Robot arm;

Simulation;

Degree of freedom;

## ABSTRACT

Penggunaan robotika dalam industri modern telah menjadi krusial, terutama dalam otomatisasi dan manufaktur. Robot lengan, sebagai elemen kunci, memerlukan pengembangan dan pengujian prototipe yang teliti untuk mencapai performa optimal. Teknologi Virtual Reality (VR) muncul sebagai solusi inovatif untuk mensimulasikan gerakan robot lengan di lingkungan virtual, memungkinkan visualisasi dan interaksi mendalam dalam berbagai derajat kebebasan (DOF). Penelitian ini bertujuan untuk membangun prototype model robot lengan 3D dan mengerjakan eksplorasi virtualisasi gerakan dari pengembangan prototipe robot lengan berbasis VR. Hasil penelitian ini diantaranya telah dilakukan integrasi teknologi VR dengan sistem kontrol robot lengan dalam lingkungan Unity, memungkinkan pengguna mengendalikan gerakan robot lengan dalam ruang 3D secara *real-time*. Studi ini menunjukkan bahwa robot lengan dengan 4 DOF dapat mencapai fleksibilitas dan presisi yang diinginkan pada skenario uji yang telah dirancang. Dari hasil pengujian antara parameter simulasi dari lingkungan unity dan perhitungan secara teoritis gerakan robot lengan yang dikendalikan menggunakan controller, diperoleh rentang koordinat yang dijangkau adalah  $x = -19,85$  cm hingga  $28,53$  cm,  $y = 21,69$  cm hingga  $52,04$  cm,  $z = -15,35$  cm hingga  $-24,87$  cm. Setelah dilakukan perhitungan perbandingan antara hasil pengujian di Unity dan hasil perhitungan berdasarkan rumus, didapatkan selisih koordinat sebesar  $x = 0,080$ ,  $y = 0,038$ , dan  $z = 0,034$ .

### Corresponding Author:

Giri Wahyu Wiriasto, Jurusan Teknik Elektro Universitas Mataram, Jalan Majapahit 63 Kota Mataram, 83127 - Lombok, Indonesia

Email: [giriwahyuwiriasto@unram.ac.id](mailto:giriwahyuwiriasto@unram.ac.id)

## PENDAHULUAN

Di era industri modern yang didorong oleh kemajuan teknologi, robotika telah menjadi komponen integral di berbagai sektor, khususnya dalam otomatisasi, manufaktur, dan pemeliharaan fasilitas industri [1]. Salah satu aspek vital dalam robotika adalah robot lengan, yang memerlukan pengembangan dan pengujian prototipe secara menyeluruh guna memastikan kinerja yang optimal dan efisien [2]. Dalam konteks ini, teknologi Virtual Reality (VR) menjadi solusi inovatif untuk mensimulasikan gerakan robot lengan dalam lingkungan virtual[3]. VR memberikan kemampuan untuk memvisualisasikan dan berinteraksi secara imersif dengan prototipe robot lengan, memungkinkan eksplorasi gerakan dalam berbagai derajat kebebasan (Degrees of Freedom/DOF) yang sulit diwujudkan melalui pengujian fisik tradisional [4]. Selain itu, virtualisasi objek 3D melalui platform Unity menjadi elemen kunci dalam menciptakan representasi visual yang akurat dari robot lengan, memungkinkan pengguna untuk memanipulasi dan menguji gerakan dengan presisi tinggi dalam simulasi yang realistis[5]. Dengan mengintegrasikan VR dan Unity, peneliti serta

insinyur dapat lebih memahami dinamika gerakan, mengidentifikasi potensi masalah, dan mengoptimalkan desain robot lengan sebelum diimplementasikan di dunia nyata. Penelitian ini akan mengeksplorasi pengembangan prototipe gerakan objek robot lengan dalam area DOF berbasis VR dan virtualisasi objek 3D di Unity, dengan tujuan untuk mengevaluasi efektivitas pendekatan ini dalam mendukung inovasi serta peningkatan performa robot lengan di industri [6].

Pada penelitian terkait sebelumnya [1], telah dikerjakan pemodelan dan simulasi robot lengan 3-DOF atau derajat kebebasan menggunakan V-Rep. Pergerakan benda dari satu posisi ke posisi lain biasanya dilakukan dengan cara konvensional metode kekuatan manusia. Perkembangan teknologi robotika membuat pergerakan benda masuk industry dengan sistem robot lengan. Dengan menggunakan sistem robot lengan, pekerjaan menjadi lebih efektif. Tujuan dari penelitian ini adalah memodelkan robot lengan 3-DOF dengan menggunakan metode kinematika perangkat lunak V-REP.

Penelitian [7] juga menggunakan metode inverse kinematics dengan pendekatan geometri pada robot denso manipulator. Pada penelitian ini pengambilan data dilakukan menggunakan perangkat lunak toolbox versi.8 pada [www.petercork.com](http://www.petercork.com) dan untuk memverifikasi hasil pengambilan data menggunakan analisis inverse kinematics dengan pendekatan geometri. Dari hasil penelitian, solusi inverse kinematics dengan pendekatan geometri akan menghasilkan delapan kondisi pergerakan robot denso manipulator. Delapan kondisi itu menunjukkan pergerakan robot untuk mencapai titik tujuan robot yang diinginkan.

### Virtual Reality (VR)

Merupakan teknologi yang memungkinkan pengguna untuk berinteraksi dengan lingkungan yang sepenuhnya atau sebagian besar disimulasikan secara komputer. Ini menciptakan pengalaman yang imersif dan sering kali menyerupai realitas yang nyata [8]. Dalam VR, pengguna dilengkapi dengan headset atau kacamata khusus yang memungkinkan mereka melihat dan merasakan lingkungan virtual yang diciptakan oleh komputer. Konsep dasar di balik VR adalah menciptakan simulasi yang menyerupai kondisi pada dunia nyata [9].

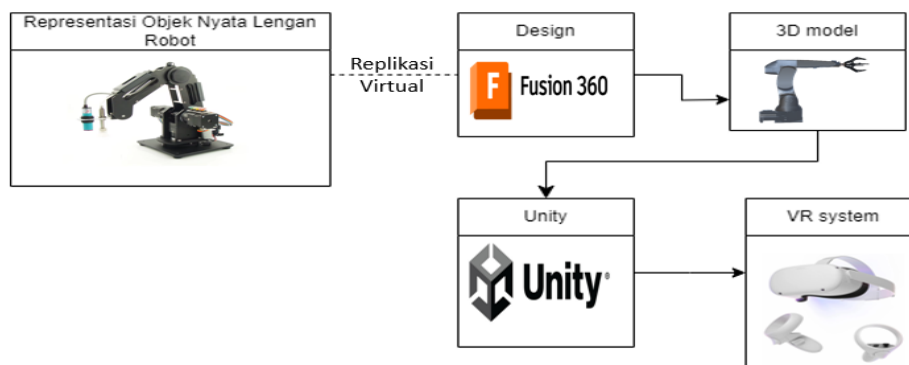
### Definisi Robot lengan

Menurut RIA (Robot Institute of America), robot lengan atau robot manipulator adalah robot multifungsi yang dirancang untuk memindahkan material, peralatan, atau perangkat tertentu. Robot ini memiliki derajat kebebasan atau Degree of Freedom (DOF) gerakan linier pada satu sumbu (axis) [10]. Robot ini hanya memiliki satu "tangan," mirip dengan tangan manusia, yang berfungsi untuk memegang atau memindahkan objek. Robot manipulator merupakan rangkaian benda kaku (rigid bodies) terbuka yang terdiri dari beberapa sendi dan terhubung oleh link, di mana setiap posisi sendi ditentukan oleh satu variabel, sehingga jumlah sendi sama dengan nilai derajat kebebasan [11].

## METODE

*Area Degree of Freedom (DOF)* atau dalam bahasa Indonesia sering disebut dengan Derajat Kebebasan adalah konsep yang digunakan dalam berbagai bidang ilmu seperti mekanika, robotika, dan rekayasa sistem. DOF mengacu pada jumlah pergerakan yang independen atau variabel yang dimiliki oleh suatu objek atau sistem [12]. Dalam konteks mekanika atau rekayasa mesin. DOF mengacu pada jumlah sumbu atau arah di mana suatu objek dapat bergerak secara independent [13]. Sebagai contoh, sebuah objek dalam ruang tiga dimensi (3D) memiliki tiga DOF translasi (bergerak maju-mundur, kanan-kiri, dan atas-bawah) dan tiga DOF rotasi (membelok, mengguling, dan memiringkan)[14].

### Perancangan Sistem



Gambar 1 Blok diagram pengembangan sistem

Pada Gambar 1, metode pengembangan dimulai dengan mencari atau menentukan objek fisik yang akan dibuat versi virtualnya. Dalam penelitian ini, objek yang dipilih adalah robot lengan sebagai model yang akan dikembangkan. Langkah selanjutnya melibatkan desain model 3D menggunakan software Autodesk, di mana objek fisik tersebut akan dibuat dalam bentuk virtual dengan detail yang mendekati aslinya [15]. Setelah model 3D selesai, model tersebut akan diimpor ke Unity Engine untuk pembuatan animasi atau gerakan pengendalian yang diperlukan, serta untuk mengimplementasikan logika yang memungkinkan model menerima perintah dari pengguna. Sebagai bagian dari pengembangan, bola akan disediakan sebagai objek yang diangkat, dan informasi mengenai koordinat, panjang, serta sudut akan ditampilkan pada papan informasi.

### Pengembangan Prototype Model 3D Robot Lengan

Spesifikasi robot lengan :

L1 = 23,5780 cm

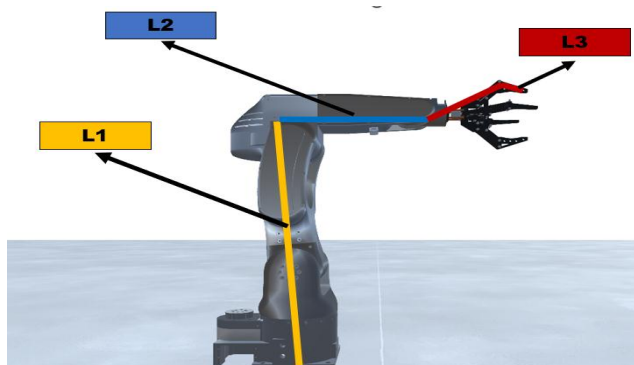
L2 = 26,4605 cm

L3 = 22,0733 cm

L1 = panjang objek 1

L2 = panjang objek 2

L3 = panjang objek 3



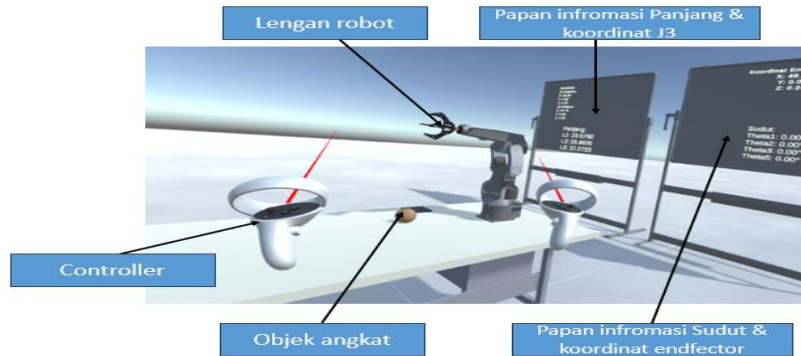
Gambar 2 Spesifikasi design prototype robot lengan

Pada gambar 2 adalah spesifikasi robot lengan dimana 3D model robot lengan dibuat di aplikasi Autodesk Fusion 360. Dimana memiliki spesifikasi L1 = 23,5780 cm yaitu panjang link 1, L2 = 26,4605 cm yaitu panjang link 2, L3 = 22,0733 cm yaitu panjang link 3. Struktur robot lengan terdiri dari 6 joint atau 6 bagian yang digabungkan menjadi 1 dan setelah 3D model selesai, model akan dipindahkan ke Unity untuk dilakukan pembuatan user interface dan user experience.

### HASIL DAN DISKUSI

Simulasi gerakan robot lengan yang dikembangkan menggunakan software Unity. Dihubungkan dengan controller VR, robot lengan dibagi menjadi 6 bagian, yaitu J1 hingga J6, di mana setiap bagian dapat digerakkan berdasarkan input yang diinginkan melalui controller VR Oculus Quest 2. Unity memungkinkan pengaturan interaksi ini dengan mudah, di mana user diminta menekan dua tombol pada controller secara bersamaan, yang disebut sebagai combo—misalnya, menekan tombol X dan thumbstick—untuk menggerakkan komponen J1 hingga J6 sesuai dengan input yang diterima. Unity juga memfasilitasi animasi realistis sehingga caput pada robot lengan dapat mengambil dan memindahkan objek ke koordinat yang ditentukan secara akurat. Pada Gambar 3 tampilan utama simulasi pada software Unity menunjukkan bagaimana Unity digunakan untuk mengontrol setiap bagian lengan atau joint pada robot lengan menggunakan joystick controller, yang dibagi menjadi dua bagian controller kanan dan kiri. Joint pada robot lengan dapat dikendalikan melalui kombinasi tombol yang telah diatur dan dikonfigurasi dalam Unity, dengan 6 kombinasi khusus yang bertugas untuk menggerakkan masing-masing joint atau komponen. Unity juga menyediakan interface yang menampilkan spesifikasi robot lengan, termasuk panjangnya, serta informasi mengenai koordinat end effector, yaitu hasil pengukuran pada ujung gripper atau caput, dan koordinat pada joint 2 dan joint 3. Selain itu, Unity memungkinkan adanya objek bola yang dapat

dipindahkan sesuai keinginan user ke berbagai lokasi di atas meja yang telah disediakan dalam simulasi, memberikan fleksibilitas dan kontrol penuh dalam lingkungan virtual ini.



Gambar 3 Simulasi pengujian prototype dalam lingkungan unity dan perangkat VR

Untuk menghubungkan unity dengan perangkat VR user dapat melakukan konfigurasi dengan menginstall XR plugin management yang telah disediakan oleh unity. Selanjutnya adalah menginputkan model controller yang akan dipakai contohnya memakai controller oculus quest 2 dan mengatur kamera khusus yaitu OVRCameraRig. Untuk menangkap input dari controller dari oculus quest 2 dapat menggunakan OVRInput dengan logika dalam bentuk pseudocode yang menghubungkan controller VR ke unity sebagai berikut:

#### Pseudocode yang menghubungkan VR ke unity

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class VRControllerInput : MonoBehaviour
{
    // Update is called once per frame
    void Update()
    {
        // Detect if the A button on the right controller is pressed
        if (OVRInput.GetDown(OVRInput.Button.One, OVRInput.Controller.RTouch))
        {
            Debug.Log("A button pressed on the right controller!");
            // Add your action here (e.g., grabbing an object)
        }

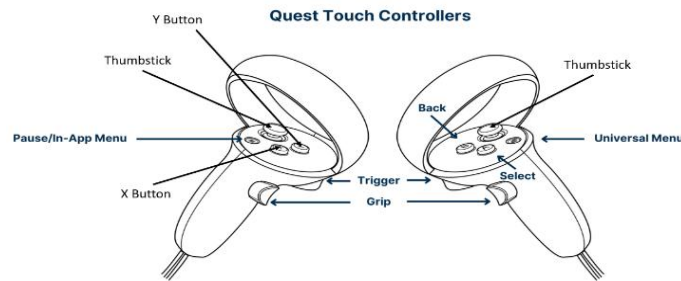
        // Detect if the Trigger on the right controller is pressed
        if (OVRInput.Get(OVRInput.Axis1D.PrimaryIndexTrigger, OVRInput.Controller.RTouch)
        > 0.5f)
        {
            Debug.Log("Trigger pressed on the right controller!");
            // Add your action here (e.g., shooting a raycast)
        }

        // Detect the position of the left controller
        Vector3 leftControllerPosition =
OVRInput.GetLocalControllerPosition(OVRInput.Controller.LTouch);
        Debug.Log("Left Controller Position: " + leftControllerPosition);

        // Detect the rotation of the left controller
        Quaternion leftControllerRotation =
OVRInput.GetLocalControllerRotation(OVRInput.Controller.LTouch);
        Debug.Log("Left Controller Rotation: " + leftControllerRotation);
    }
}
```

Baris kode diatas dirancang untuk menangkap input dari kontroler Oculus Quest 2 dalam aplikasi Unity. Setiap frame, skrip ini memeriksa apakah tombol A pada kontroler kanan ditekan (`OVRInput.GetDown`), yang dapat digunakan untuk memicu aksi tertentu, seperti mengambil objek. Selain itu, skrip ini memeriksa apakah trigger pada kontroler kanan ditekan lebih dari setengah (`OVRInput.Get`), memungkinkan pengembangan interaksi tambahan seperti menembakkan raycast atau memicu tindakan lainnya. Script ini juga terus-menerus memantau dan mencatat posisi (`GetLocalControllerPosition`) dan rotasi (`GetLocalControllerRotation`) kontroler kiri, yang berguna untuk mengontrol objek atau

menyesuaikan perspektif berdasarkan gerakan fisik pengguna. Secara keseluruhan, skrip ini menyediakan fondasi dasar untuk deteksi input dan manipulasi gerakan dalam lingkungan VR menggunakan Oculus Quest 2. Tugas dari baris kode di atas adalah menghubungkan VR yang khususnya controller ke unity, untuk menghubungkannya dapat menginstall tools XR interactions toolkit.



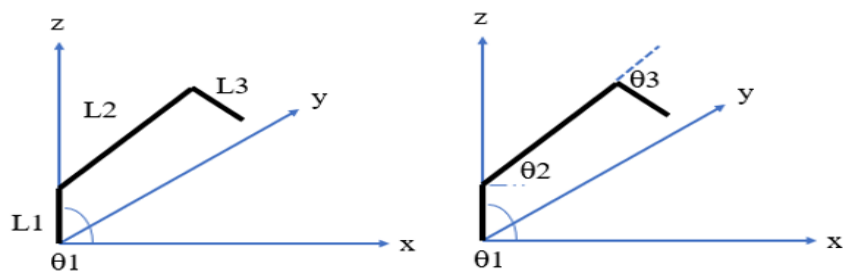
Gambar 4 layout controller oculus quest 2

Pada mode controller robot lengan digerakkan menggunakan controller oculus quest 2, dikenal sebagai Oculus Touch, adalah perangkat ergonomis yang memungkinkan interaksi intuitif dalam lingkungan VR. Controller ini dilengkapi dengan berbagai tombol, termasuk trigger button dan grip button, serta joystick analog untuk navigasi yang lebih lengkap dapat dilihat pada gambar 4 yaitu layout controller oculus quest 2. Dalam aplikasi dan game, trigger button sering digunakan untuk menembak atau memilih objek, sementara grip button digunakan untuk menggenggam objek atau melakukan interaksi fisik lainnya. Pengguna juga dapat memanfaatkan library seperti Oculus Integration SDK di Unity untuk mengakses dan mengatur kontrol controller, termasuk deteksi gerakan tangan dan interaksi objek dalam dunia virtual. Oculus SDK menyediakan berbagai API dan alat untuk pengembang agar bisa memanfaatkan fitur-fitur controller secara maksimal dalam pengembangan aplikasi VR. Pada implementasinya saya menggunakan controller untuk menggerakkan bagian robot lengan dengan memanfaatkan kombinasi antara 2 tombol misalnya untuk menggerakkan joint 1 dapat menekan tombol thumbstick kanan dan trigger kanan secara bersamaan. Desain dan layout tombol pada controller oculus quest 2 dapat dilihat pada gambar 4.

### Hasil pengujian Robot lengan

#### A. Transform dan implementasi rumus

Dalam metode geometri jumlah joint (sendi) pada robot akan sangat berpengaruh terhadap persamaan yang akan digunakan untuk mencari koordinat posisi robot atau benda. Semakin banyak jumlah joint pada robot maka akan semakin kompleks juga persamaannya. Penjelasan joint dan link pada robot lengan 3 derajat kebebasan ditunjukkan pada Gambar 5.



Gambar 5 ilustrasi Joint dan Link Robot Lengan 3 Derajat Kebebasan

Gambar 5 menggambarkan representasi robot lengan 3 derajat kebebasan dalam keadaan ideal, dimana tidak terdapat sambungan-sambungan khusus antar link dan juga tidak adanya sambungan pada ujung end-effector dapat dilihat pada persamaan dibawah [1].

$$x = \{(L_2 \cos \theta_2) + (L_3 \cos(\theta_3 - \theta_2))\} \times \cos \theta_1$$

$$y = \{(L_2 \sin \theta_2) + (L_3 \sin(\theta_3 - \theta_2))\} \times \sin \theta_1$$

$$z = L_1 + (L_2 \sin \theta_2) - (L_3 \sin(\theta_3 - \theta_2))$$

Pada unity untuk menghitung transformasi geometri tidak perlu menghitung manual seperti pada rumus yang telah tercantum pada transformasi geometri karena di unity sudah terdapat fungsi transform. Transform di Unity adalah komponen dasar yang menentukan posisi (position), rotasi (rotation), dan skala (scale) dari sebuah GameObject. Posisi mengatur lokasi objek dalam world space atau local space jika merupakan child dari GameObject lain. Rotasi mengatur orientasi objek menggunakan quaternion atau Euler angles. Skala mengatur ukuran objek di sepanjang sumbu x, y, dan z. Selanjutnya pada simulasi pengujian gerakan robot lengan digunakan metode geometri untuk menentukan koordinat robot lengan dengan rumus yang telah disesuaikan dengan spesifikasi robot lengan yang dibuat [1].

$$x = (L_2 \cos\theta_2) + (L_3 \cos\theta_3) \times \sin\theta_1 \times \cos\theta_5$$

$$y = (L_2 \cos\theta_2) + (L_3 \cos\theta_3) \times \sin\theta_1 \times \cos\theta_5 + L1$$

$$z = (L_2 \sin\theta_2) - (L_3 \sin\theta_3) \times \sin\theta_1$$

Algoritma pseudocode perhitungan koordinat endeffector

```
// Perhitungan posisi end effector
float xEnd = (L2 * Mathf.Cos(theta2Rad) + L3 * Mathf.Cos(theta3Rad)) * Mathf.Sin(theta1Rad) *
Mathf.Cos(theta5Rad)
float yEnd = (L2 * Mathf.Cos(theta2Rad) + L3 * Mathf.Cos(theta3Rad)) * Mathf.Sin(theta1Rad) *
Mathf.Cos(theta5Rad) + L1
float zEnd = L2 * Mathf.Sin(theta2Rad) - L3 * Mathf.Sin(theta3Rad) * Mathf.Sin(theta1rad)
```

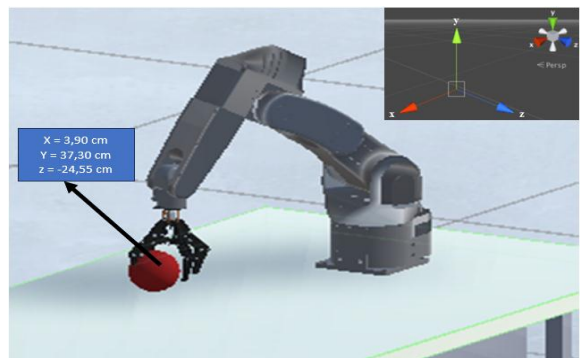
Script diatas adalah algoritma pseudocode implementasi dalam pemrograman c# unity yang mengatur dan menghitung posisi *end effector* dari robot lengan berdasarkan sudut rotasi lokal dari beberapa joint (J1, J2, J3, J5). Dalam metode *Update()*, sudut-sudut (theta1, theta2, theta3, theta5) diperoleh dari rotasi lokal masing-masing joint. Sudut-sudut ini kemudian dikonversi dari rentang [0°, 360°] ke [-180°, 180°] menggunakan fungsi *ConvertToNegativeRange*, dan selanjutnya dikonversi dari derajat ke radian. Setelah konversi, sudut-sudut dalam radian digunakan untuk menghitung posisi *end effector* (xEnd, yEnd, zEnd) menggunakan rumus trigonometri yang melibatkan panjang link (L1, L2, L3). Hasilnya adalah posisi 3D dari *end effector* yang diperoleh dengan memperhitungkan sudut rotasi dan panjang link dari robot lengan.

B. Hasil Pengujian Gerak Robot lengan

Pada pengujian simulasi gerak robot lengan mode controller posisi awal benda diletakkan pada 3 koordinat yang berbeda yaitu pada pengujian 1 sampai 3 posisi awal benda diatur pada koordinat x = 3,90 cm, y = 37,30 cm, z = -24,55 cm, dapat dilihat pada pengujian pertama dimasukan sudut putar sebesar  $\theta_1 = 34,71^\circ$ ,  $\theta_2 = -51,32^\circ$ ,  $\theta_3 = -29,48^\circ$ ,  $\theta_5 = -16,74^\circ$  didapatkan koordinat akhir benda x = 26,12 cm, y = 52,04 cm, z = -19,65 cm . Pada pengujian ke 4 sampai 7 posisi awal benda diletakkan pada koordinat x = -5,33 cm, y = 33,85 cm, z = -24,15 cm dapat dilihat pada pengujian ke 4 dimasukan sudut putar sebesar  $\theta_1 = -49,37^\circ$ ,  $\theta_2 = -60,62^\circ$ ,  $\theta_3 = -5,01^\circ$ ,  $\theta_5 = -35,01^\circ$  didapatkan koordinat akhir benda x = -19,85 cm, y = 24,78 cm, z = -24,32 cm . , pada pengujian ke 8 sampai 10 posisi awal benda diletakkan pada koordinat x = 2,25 cm, y = 41,48 cm, z = -19,80 cm. Kemudian benda dipindahkan dengan masukan sudut motor sebesar yang tertera pada tabel 1.

Tabel 1 Hasil pengujian gerak robot lengan

Uji	Koordinat Awal Benda (cm)			Sudut Putar °				Koordinat Akhir Benda (cm)		
	X	y	z	$\theta_1^\circ$	$\theta_2^\circ$	$\theta_3^\circ$	$\theta_5^\circ$	x	y	z
1	3,90	37,30	-24,55	34,71	-51,32	-29,48	-16,74	26,12	52,04	-19,65
2	3,90	37,30	-24,55	42,87	-60,80	-7,13	-24,65	19,66	48,16	-21,49
3	3,90	37,30	-24,55	-31,41	-60,90	-10,15	-27,30	-14,63	27,77	-24,87
4	-5,33	33,85	-23,15	-49,37	-60,62	-5,01	-35,01	-19,85	24,78	-24,32
5	-5,33	33,85	-23,15	-25,61	-66,60	7,73	-29,02	-11,10	26,96	-23,18
6	-5,33	33,85	-23,15	-1,84	-64,26	4,56	-34,05	-0,81	34,57	-23,79
7	-5,33	33,85	-23,15	4,13	-77,12	37,76	-49,29	0,98	30,18	-26,63
8	2,25	41,48	-19,80	-51,12	-78,84	37,76	-53,25	-9,39	21,69	-16,89
9	2,25	41,48	-19,80	-10,89	-50,88	-29,48	-19,59	-5,92	37,33	-22,30
10	2,25	41,48	-19,80	90	-60,36	-10,25	-34,82	17,96	49,10	-15,35

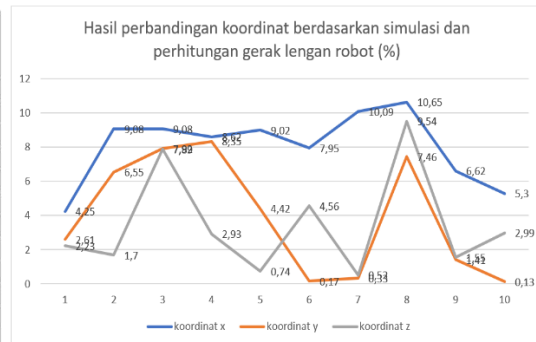


Gambar 6 koordinat robot lengan pada pengujian 1

Pada gambar 6 adalah posisi robot lengan pada koordinat  $x = 3,90$  cm,  $y = 37,30$  cm dan  $z = 24,55$  cm dimana posisi robot lengan dan bola berada pada sebelah kanan meja. Pada pengujian pertama didapatkan hasil  $x = 18,76$ ,  $y = 47,85$  dan  $z = -15,70$  dimana hasil ini memiliki persentase error dibawah 5%, error yang terdapat pada hasil perbandingan karena pada unity pada saat perhitungan mengambil hanya 2 angka dibelakang koma. Hasil perbandingan pengujian simulasi robot lengan dengan hasil koordinat berdasarkan perhitungan dapat dilihat pada tabel 2.

Tabel 2 Hasil perhitungan gerak robot lengan

Uji	Koordinat Akhir Benda Berdasarkan simulasi (cm)			Koordinat Akhir Benda Berdasarkan Perhitungan (cm)			Persentase Kesalahan %		
	x	y	z	x	y	z	x	y	z
1	17,96	49,10	-15,35	18,76	47,85	-15,70	4,25	2,61	2,23
2	19,66	48,16	-21,49	21,62	45,20	-21,13	9,08	6,55	1,70
3	-14,63	27,77	-24,87	-16,09	25,73	-23,05	9,08	7,92	7,89
4	-19,85	24,78	-24,32	-21,72	22,87	-23,63	8,62	8,35	2,93
5	-11,10	26,96	-23,18	-12,20	25,82	-23,01	9,02	4,42	0,74
6	-0,81	34,57	-23,79	-0,88	34,51	-22,75	7,95	0,17	4,56
7	0,98	30,18	-26,63	1,09	30,28	-26,77	10,09	0,33	0,52
8	-9,39	21,69	-16,89	-10,51	20,17	-15,42	10,65	7,46	9,54
9	-5,92	37,33	-22,30	-6,34	36,81	-22,65	6,62	1,41	1,55
10	28,53	52,04	-19,65	30,73	52,11	-19,08	5,30	0,13	2,99
Rata-rata persentase kesalahan							8,08	3,80	3,46



Gambar 7 hasil perbandingan gerak robot lengan

Pada gambar 7 adalah grafik perbandingan dari simulasi dan perhitungan gerak robot lengan dimana didapatkan hasil yang fluktuatif contohnya pada pengujian pertama didapatkan hasil persentase kesalahan  $x = 4,25\%$  dimana hasil ini didapat dari selisih 17,96 dan 18,76 cm, selanjutnya  $y = 2,61\%$  dimana hasil ini didapat dari selisih 49,10 dan 47,85 cm dan  $z = 2,23\%$  didapat hasil selisih -15,35 dan -15,70 cm. 10 percobaan pengujian dilakukan dan didapatkan hasil rata-rata  $x = 8,08\%$ ,  $y = 3,80\%$  dan  $z = 3,46\%$ .

### 1. Kesimpulan

Penelitian ini berhasil mengembangkan prototype model 3D robot lengan dengan spesifikasinya. Telah dapat diintegrasikan perangkat VR dengan sistem kontrol robot lengan dalam lingkungan Unity menggunakan controller oculus quest 2. Integrasi ini memungkinkan pengguna untuk mengendalikan gerakan robot lengan dalam ruang 3D secara real-time. Penggunaan unity dan *XR Interaction Toolkit* memungkinkan pembuatan lingkungan interaktif yang realistis. Penelitian ini menunjukkan bahwa dengan mempertimbangkan jumlah DOF yang tepat yaitu dalam penelitian ini memodelkan robot lengan dengan 4 DOF dengan joint 1 bergerak secara horizontal dan 3 joint lainnya bergerak secara vertikal, robot lengan dapat mencapai fleksibilitas dan presisi yang diinginkan.

hasil pengujian simulasi dan perhitungan gerakan robot lengan yang digerakan menggunakan controller didapatkan rentang koordinat yang dijangkau adalah  $x = -19,85$  cm sampai 28,53 cm,  $y = 21,69$  cm sampai 52,04 cm,  $z = -15,35$  cm sampai -24,87 cm dan setelah dilakukan perhitungan perbandingan antara hasil pengujian pada unity dan hasil perhitungan berdasarkan rumus didapatkan persentase kesalahan dengan hasil rata-rata  $x = 8,08\%$ ,  $y = 3,80\%$  dan  $z = 3,46\%$ .

### DAFTAR PUSTAKA

- [1] Az Uchrowi, Lasmadi, Sutjipto (2019). *Pemodelan dan Simulasi Robot Lengan 3 DOF Menggunakan V-REP*. AVITEC, Vol. 1, No. 1.
- [2] Uhlenkamp, J. F., Hribernik, K., Wellsandt, S., & Thoben, K. D. (2019). Digital Twin Applications: A first systemization of their dimensions. In 2019 IEEE International Conference on Engineering, Technology and Innovation (ICE/ ITMC).
- [3] Naseri & Gil. (2023). *Digital Twin of electric vehicle battery systems: Comprehensive review of the use cases, requirements, and platforms*. *Renewable and Sustainable Energy Reviews* 179.
- [4] Matulisa & Harvey. (2021). A robot arm *Digital Twin* utilising reinforcement learning. *Computers & Graphics* 95 106–114.
- [5] Zable, A., Hollenberg, L., Velloso, E., & Goncalves, J. (2020, November). Investigating immersive Virtual Reality as an educational tool for quantum computing. In *Proceedings of the 26th ACM Symposium on Virtual Reality Software and Technology* (pp. 1-11).

- 
- [6] Lin, M., & San, L. (2019). Construction of Robotic Virtual Laboratory System Based on Unity3D. *IOP Conf. Series: Materials Science and Engineering* 768.
- [7] Prasetya, I. E., & Agustinah, T. (2015). Invers Kinematics dengan Solusi Closed Form pada Robot Denso Manipulator. *Jurnal Teknik ITS*, 4(1), 2337-3539.
- [8] Sulistianingsih, A. S., & Kustono, D. (2022). Potensi Penggunaan Teknologi Augmented Reality (AR) dan *Virtual Reality* (VR) dalam Pembelajaran Sejarah Arsitektur di Era Pandemi Covid-19. *Jupiter (Jurnal Pendidikan Teknik Elektro)*, 7(1), 10-18.
- [9] Pramesti & Sopiya. (2022). SYSTEMATIC LITERATURE REVIEW: PEMANFAATAN *VIRTUAL REALITY* (VR) SEBAGAI ALTERNATIF MEDIA PEMBELAJARAN. *Jurnal Pendidikan Teknologi dan Kejuruan* Vol. 19, No. 2.
- [10] Manolescu, Denis & Secco, Emanuele. (2022). Design of a 3-DOF Robotic Arm and implementation of D-H Forward Kinematics
- [11] Ripaldo & Hasan. (2023). Implementasi DOF (Degree Of Freedom) Pada Pergerakan Motor Stepper Smart Inventory 3 Axis. Jurusan Teknik Elektro Politeknik Negeri Sriwijaya.
- [12] Umam, K., Haryanto, H., & Alfita, R. (2019). Rancang Bangun Robot Pembersih Kaca Otomatis Berbasis Mikrokontroler ARM STM32 Dengan Sensor Proximity. *Indonesian Journal of Engineering and Technology (INAJET)*, 2(1), 24-29.
- [13] Sulistyowati, R., & Kurniawan, M. Y. (2012). Kontrol Posisi pada Robot lengan Pemindah Barang Berbasis Mikrokontroler ATMega16. In *Prosiding Seminar Nasional Sains dan Teknologi Terapan Institut Teknologi Adhi Tama Surabaya*
- [14] Utomo, B., & Munadi. (2013). Analisis Forward dan Invers Kinematics pada Simulator Arm Robot 5 Derajat Kebebasan. *Jurnal Teknik Mesin (S-1)*, 1(3), 11-20.
- [15] Dof Chen, Chaoda & Lv, Zhisheng & Wu, Siyang & Li, Kangxing & Chen, Ziyang & Huang, Zhixing. (2024). Design of control system for six-degree-of-freedom robotic arm. *Journal of Physics: Conference Series*. 2724. 012006. 10.1088/1742-6596/2724/1/012006.