

PENGUJIAN JARAK OPTIMAL ROBOT MANIPULATOR BERBASIS IMAGE-BASED VISUAL SERVOING

Reza Poerwandito¹, Munadi², Ismoyo Haryanto²

¹Mahasiswa Jurusan Teknik Mesin, Fakultas Teknik, Universitas Diponegoro

²Dosen Jurusan Teknik Mesin, Fakultas Teknik, Universitas Diponegoro

Jl. Prof. Sudharto, SH., Tembalang-Semarang 50275, Telp. +62247460059

*E-mail: rezapoer@gmail.com

Abstrak

Perkembangan teknologi robotik merupakan hal yang sangat penting dalam kehidupan manusia untuk mempermudah sebuah pekerjaan manusia. Sistem *visual servoing* menjadi sebuah pilihan dalam pengembangan untuk penggunaan di berbagai sektor seperti industri, manufaktur, transportasi, kesehatan, militer, dan juga sekuriti. Salah satu metode visual servoing adalah *Image-Based Visual Servoing* (IBVS) yang secara langsung menggunakan informasi fitur visual dari kamera untuk mengendalikan kerja sendi robot atau derajat kebebasan robot (*Degree of Freedom*). Dalam Penelitian ini dirancang sebuah robot manipulator 2 DOF yang tujuannya adalah untuk melakukan penjejakan terhadap target bergerak, yang objeknya adalah wajah manusia. Dilakukan perancangan dalam program Visual Studio Code dengan menggunakan bahasa Python. Pengendali yang telah dirancang kemudian diimplementasikan ke dalam perangkat yang telah dibuat. Metode kendali tersebut dirancang untuk mengendalikan dua aktuator untuk memposisikan kamera dengan konfigurasi *pan-tilt* (2 axis) untuk menghasilkan gerak keluaran sistem yang stabil dengan performa yang baik. Dari hasil penelitian, didapat bahwa metode kendali yang dirancang andal dalam mengendalikan sistem. Pada pengujian dengan keadaan suatu wajah objek yang terletak pada jarak 150 cm, sistem dapat bergerak dan mendeteksi objek tersebut pada titik tengah frame (1280 x 720), yaitu koordinat (640, 360). Nilai *threshold* adalah nilai toleransi simpangan dalam satuan piksel. Fungsi dari penetapan nilai *threshold* adalah untuk mengeluarkan *output* sinyal pada masing-masing servo untuk bergerak memposisikan objek ditengah frame. Pada penelitian ini diketahui bahwa nilai *threshold* sebesar 75 piksel adalah nilai yang paling ideal untuk menghasilkan gerakan yang dinamis dan stabil tanpa adanya *noise*.

Kata kunci: penjejakan wajah; robot manipulator; *visual servoing*

Abstract

The development of robotic technology is very important in human life to make human work easier. Visual servoing systems are an option in development for use in various sectors such as industry, manufacturing, transportation, health, military and security. One visual servoing method is Image-Based Visual Servoing (IBVS) which directly uses visual feature information from the camera to control the work of the robot's joints or the robot's degrees of freedom (Degree of Freedom). In this research, a 2 DOF manipulator robot was designed whose aim was to track a moving target, the object of which was a human face. The design was carried out in the Visual Studio Code program using the Python language. The controller that has been designed is then implemented into the device that has been created. This control method is designed to control two actuators to position the camera with a pan-tilt (2 axis) configuration to produce stable system output motion with good performance. From the research results, it was found that the control method designed was reliable in controlling the system. In testing with the face of an object located at a distance of 150 cm, the system can move and detect the object at the center point of the frame (1280 x 720), namely coordinates (640, 360). The threshold value is the deviation tolerance value in pixel units. The function of setting the threshold value is to issue a signal output to each servo to move to position the object in the middle of the frame. In this research, it is known that a threshold value of 75 pixels is the most ideal value to produce dynamic and stable movement without any noise.

Keywords: *face tracking; manipulator robot; visual servoing*

1. Pendahuluan

Seiring dengan perkembangan teknologi, penelitian mengenai robotika sangatlah banyak dan berkembang pesat. Penelitian-penelitian tersebut juga disertai dengan kemajuan teknologi yang terus menerus sehingga saat ini sudah banyak jenis robot yang dimanfaatkan untuk kepentingan produksi ataupun kepentingan lainnya [1].

Metode yang beragam digunakan untuk dapat mengoperasikan model robot yang dapat bekerja untuk meniru cara kerja manusia, contohnya robot dapat memiliki kemampuan untuk dapat melihat, mengamati, dan mendeteksi sebuah objek visual yang digunakan untuk informasi visual dengan memanfaatkan kamera untuk mendapatkan citra statik maupun dinamik (bergerak). Kamera ini berfungsi seperti halnya mata yang digunakan manusia untuk melihat dan berperan sebagai sensor pada robot [2]. Informasi visual telah digunakan sebagai sensor umpan balik untuk menyelesaikan berbagai aplikasi robot, baik dalam dunia industri manufaktur, keamanan, dan industri hiburan. Permasalahan untuk mengendalikan

pergerakan robot menggunakan umpan balik dari sensor penglihatan tersebut disebut sebagai *visual servoing* [3].

Visual servoing telah dipelajari dalam berbagai bentuk selama ini yang perkembangan pemanfaatannya dimulai dari tugas sederhana seperti memilih dan menempatkan hingga dapat menjalankan fungsi manipulasi objek yang canggih dan *real-time* saat ini. Contohnya yaitu kemampuan untuk mengganti tenaga kerja manusia dan melakukan pemeriksaan untuk mengetahui suatu apakah telah memenuhi persyaratan sebuah produk, seperti memeriksa bentuk, ukuran dan ketebalan produk, dengan akurasi deteksi yang tinggi. *Visual servoing* adalah metode pengendalian servo untuk mengendalikan robot atau pergerakan pose kamera ke posisi yang diinginkan dengan cepat menggunakan citra input visual [4]. Sistem *visual servoing* telah banyak digunakan di berbagai sektor seperti industri, manufaktur, transportasi, kesehatan, militer, dan juga sekuriti. Metode *visual servoing* secara garis besar dibagi menjadi dua, yaitu *Image-Based Visual Servoing* (IBVS) yang secara langsung menggunakan informasi visual dari kamera untuk mengendalikan derajat kebebasan robot atau *degree of freedom* (DOF) dan metode lain yang menggunakan interpretasi geometris dari ekstraksi fitur kamera seperti metode *Pose-Based Visual Servoing* (PBVS) dan metode *hybrid* [4]. *Visual servoing* dapat digunakan untuk mengendalikan berbagai sistem dengan banyak DOF. Pada platform dengan banyak DOF, robot dapat digunakan untuk menyelesaikan tugas-tugas yang kompleks, sedangkan pada platform dengan 2 DOF dan konfigurasi *eye-in-hand* seperti pada platform kamera *pan-tilt*, dapat digunakan untuk melakukan tugas-tugas *surveillance* (pengintaian).

Tujuan utama dari sistem *visual servoing* adalah untuk mengontrol *end-effector* lengan robot dengan sumber dari sekumpulan fitur gambar hingga mencapai konfigurasi lengan robot yang diinginkan [5]. Arsitektur dari *visual servoing* dapat dibagi menjadi: *Image Based Visual Servoing* (IBVS), *Position Based Visual Servoing* (PBVS), dan *Hybrid Visual Servoing* (HBS) [1]. Penggunaan metode IBVS sangat unggul untuk mengatasi berbagai kesalahan pemodelan yang berbeda, tetapi memiliki beberapa permasalahan seperti: penanganan saat terjadi kendala pada sistem dan konvergensi ketika sistem mengharuskan untuk melakukan perpindahan dengan skala besar [1]. Untuk mengatasi masalah ini, pendekatan lanjutan dari teori kontrol telah disesuaikan dengan sistem servo visual.

Masalah utama dalam perancangan sistem *visual servoing* adalah untuk memiliki kontrol sistem yang kuat dan stabil, pengontrol harus dirancang dengan mempertimbangkan batasan dinamis dari robot manipulator dan juga konfigurasi ruang kerja. Implementasi secara langsung dari skema input visual sangat sulit untuk diperoleh secara instan karena hambatan pada keterbatasan akses ke komponen penggerak lengan robot (motor servo) yang merupakan bagian dari robot [5]. Karena hambatan ini, diperlukan metode untuk memperluas kemampuan pengontrol robot. Metode pengembangan ini harus dirancang sedemikian rupa sehingga semua periode waktu yang ada (komunikasi, pemrosesan gambar hingga komputasi kontrol) disertakan dan sistem bekerja secara *real-time*.

2. Pendukung Perancangan dan Metode Penelitian

2.1 Software

Software utama yang digunakan dalam penelitian ini adalah software yang terkait dengan pemrograman, pembuatan model object detection hingga menjadi sebuah model sistem *visual servoing*. Pemilihan software digunakan untuk menunjang proses deteksi objek hingga ke proses implementasi kode pada hardware sehingga dapat dijalkannya program penjejakan wajah dengan sistem *visual servoing*.

2.2 Hardware

Hardware merupakan semua perangkat atau komponen yang membentuk dasar dari suatu sistem komputer dan mendukung eksekusi dari *software*. Tabel 1 menunjukkan seluruh komponen yang digunakan pada penelitian ini.

Table 1 Komponen penelitian.

<i>Hardware</i>	<i>Software</i>
<ul style="list-style-type: none"> • Robot Manipulator Custom • Webcam Logitech C270HD • Arduino Uno R3 • Laptop (PC) 	<ul style="list-style-type: none"> • Arduino IDE • Python 3.7.9 • Visual Studio Code • OpenCV 4.8.1

2.3 Objek Detection and Servoing

Deteksi objek adalah proses menemukan suatu objek dari kelas tertentu, seperti wajah, mobil, dan pohon, dalam gambar atau video. Tidak seperti klasifikasi, deteksi objek dapat mendeteksi banyak objek, serta lokasinya di gambar. Detektor objek akan mengembalikan daftar objek yang terdeteksi dengan informasi kelas objek, probabilitas, dan koordinatnya untuk setiap objek [8]. Setelah berhasil mendeteksi dan mengakuisisi lokasi koordinat objek pada bidang gambar [9], kemudian koordinat yang dihasilkan lalu diproses menjadi acuan sistem servo untuk melakukan gerakan. Dalam penelitian ini objek yang akan digunakan adalah wajah manusia. Untuk bisa melakukan proses deteksim perlu dilakukan pembuatan model deteksi terlebih dahulu. Pada bagian ini, akan membahas prosedur penjejakan wajah dimulai dari perancangan sistem detection, perancangan sistem tracking, perancangan sistem draw annotation, perancangan sistem Simulink secara keseluruhan, hingga implementasi langsung pemrograman terhadap hardware.

Output dari prosedur ini adalah berupa pendeteksian kelas objek, pendeteksian koordinat letak objek pada gambar, dan pemrograman implementasi pada hardware berupa robot manipulator yang dilengkapi dengan perangkat kamera serta motor servo sebagai aktuator penggerak. Prosedur ini dilakukan menggunakan beberapa software dan platform yang berbeda.

Pada saat model sudah selesai dibuat, proses pendeteksian objek secara real-time dilakukan pada Arduino Uno untuk proses inferensi program yang dijalankan. Berikut ini akan dijelaskan langkah-langkah yang dilakukan dalam prosedur object detection hingga implementasi pada hardware untuk melakukan proses visual servoing.

2.4 Kode Pemrograman dengan Menggunakan Python

Implementasi Pemrograman dilakukan dengan menggunakan Bahasa pemrograman Python yang prosesnya dilakukan dalam aplikasi Visual Studio Code. Terdapat beberapa ketentuan yang harus dipenuhi untuk melakukan pemrograman yaitu: Python versi 3.7.9, Arduino IDE untuk mendefinisikan modul pyfirmata pada servo yang terhubung pada mikrokontroler Arduino Uno, dan OpenCV sebagai media penyimpanan data training.

Setelah melakukan instalasi aplikasi serta library yang dibutuhkan, lalu dapat dilakukan proses pemrograman dengan menggunakan bahasa Python pada aplikasi Visual Studio Code. Berikut merupakan langkah-langkah pemrograman pada Visual Studio Code.

a. Melakukan import library yang dibutuhkan untuk proses pemrograman

Pada proses pemrograman, diperlukan beberapa library sebagai acuan sistem untuk melakukan proses deteksi objek, berikut adalah beberapa library yang harus di-import. Berikut adalah perintah untuk import library yang dibutuhkan.

b. Membuat perintah untuk memposisikan servo pada posisi awal

Pada awal proses deployment sistem pada perangkat yang telah dirakit, kedua servo diberikan perintah pada posisi 90° dalam derajat posisi servo pada kedua servo baik pada servo yang menggerakkan sumbu x maupun servo yang menggerakkan sumbu y.

c. Mendefinisikan citra gambar serta konfigurasi penangkapan gambar

Proses definisi terhadap kamera yang digunakan serta konfigurasi karakterisasi gambar yang ditangkap oleh setiap frame merupakan hal yang krusial dalam proses visual servoing, maka dilakukan pemilihan modul webcam, serta piksel gambar keseluruhan (1280, 720), serta penentuan axis gambar yaitu sumbu x (matriks baris), dan sumbu y (matriks kolom). Serta notasi peringatan jika modul penangkapan gambar tidak dapat diakses.

d. Mendefinisikan port Arduino yang digunakan

Pada penelitian ini, perangkat Arduino terhubung pada PC. Keterangan port pada setiap computer mungkin dapat berbeda yang pada dasarnya dapat dilihat pada Device Manager di setiap computer untuk mendapatkan port mana yang terhubung dengan Arduino yang pada penelitian ini menggunakan port COM3. Pendefinisian board yang sebelumnya telah dilakukan dengan run library pyfirmata di Arduino IDE lalu dengan melakukan perintah pada baris 19. Proses pendefinisian servo yang digunakan pada masing-masing sumbu gerak ditentukan dengan pemilihan pin yang sesuai.

e. Mendefinisikan modul deteksi yang digunakan serta penentuan *threshold*

Proses pemrograman untuk menentukan modul deteksi objek yang diinginkan dilakukan dengan melakukan definisi pada detector di posisi awal servo. Penentuan nilai *threshold* dilakukan pada tahap ini untuk menentukan toleransi simpangan piksel yang menentukan hasil akurasi pendeteksian objek.

f. Membuat algoritma deteksi objek saat objek terdeteksi oleh sistem

Algoritma deteksi objek berjalan ketika sistem mendapatkan citra wajah yang terdeteksi oleh sistem. Dilakukan definisi matriks baris dan kolom yang kemudian didefinisikan bidang tengah gambarnya center pada masing-masing sumbu bidang gambar.

g. Membuat algoritma logika saat posisi objek melampaui *threshold*

Pembuatan algoritma logika pergerakan servo merupakan proses yang krusial untuk memberi perintah yang spesifik pada kedua servo. Posisi objek didefinisikan sebagai (fx, fy) lalu nilai posisi tersebut menjadi pengurang titik pusat koordinat bidang gambar untuk mendapatkan error pada masing-masing sumbu. Lalu jika error yang didapatkan melewati simpangan toleransi (*threshold*) yang telah ditentukan, servo akan bergerak sejauh 2 (dua) derajat hingga objek berada ditengah bidang gambar.

h. Membuat batasan derajat putar serta menentukan perubahan posisi servo

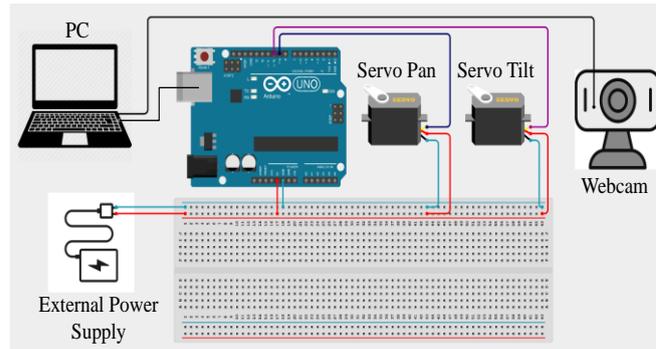
Servo memiliki batasan derajat putar yaitu 180 derajat, jadi pada bagian ini dilakukan pembatasan terhadap besar putaran maksimal yaitu 0-180 derajat. Untuk melakukan input gerak pada servo, dibuat fungsi posisi servo pada masing-masing servo untuk melakukan gerakan yang spesifik sesuai dengan kebutuhan.

i. Pembuatan notasi gambar serta memunculkan *window display* deteksi objek

Pembuatan notasi gambar diperlukan untuk memberi keterangan pada display gambar pada saat pengujian. Notasi keterangan gambar yang ditampilkan adalah posisi servo pada masing-masing servo, lingkaran untuk menunjukan posisi objek yang lengkap dengan posisi koordinat objek pada bidang gambar, notasi apakah pada proses saat itu objek terdeteksi atau tidak. Citra gambar yang didapatkan lalu dapat ditampilkan pada display window.

2.5 Wiring Diagram Assembly Arduino

Pembuatan skema wiring diagram merupakan sebuah langkah penting dalam perancangan sebuah project. Dalam sebuah wiring diagram terdapat informasi dan penjelasan mengenai proses penyusunan komponen elektronik menjadi suatu rangkaian yang dapat bekerja sesuai dengan perintah yang diinginkan atau produk akhir hasil perancangan. Pemasangan komponen dilakukan dengan hati-hati untuk memastikan bahwa komponen yang terpasang sesuai dengan desain dan spesifikasi yang telah ditetapkan. wiring diagram pada rangkaian produk dengan menggunakan mikrokontroler Arduino Uno ditunjukkan pada Gambar 1.



Gambar 1 Wiring diagram rangkaian Arduino Uno

2.6 Metode Pengujian

Pengujian dilakukan dengan meletakkan model robot manipulator pada lokasi tertentu dengan jarak optimal kamera untuk mengetahui program dapat mendeteksi objek wajah dan melakukan gerakan servo yang telah deprogram untuk menjaga objek wajah pada titik pusat bidang gambar yang diakuisisi oleh kamera webcam. Berikut adalah beberapa metode pengujian yang dilakukan.

2.11.1 Pengujian Jarak Kamera

Pengujian jarak kamera dilakukan untuk mengetahui jarak optimal kamera yang dapat digunakan untuk program deteksi objek penelitian berupa wajah. Pengujian dilakukan dalam beberapa variasi jarak kamera. Hasil dari pengujian jarak ini kemudian digunakan untuk dasar pengujian sistem *object detection* yang dapat menghasilkan data input untuk pergerakan motor servo.

2.11.2 Pengujian Gerak Robot Manipulator

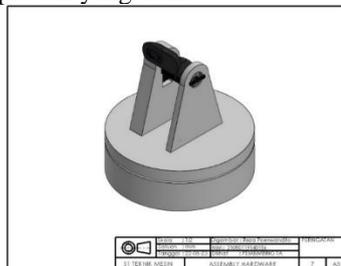
Pengujian gerak robot manipulator dilakukan dengan mengukur kualitas hasil deteksi wajah hingga pergerakan masing masing servo terhadap input dari koordinat objek berada. Dilakukan pengujian secara simulasi dan real-time terhadap suatu objek wajah statis yang terleak pada titik koordinat tertentu untuk melihat respon dinamis keluaran sistem [10]. Prosedur pengujian deteksi objek dan pengujian gerak robot manipulator diuraikan pada keterangan dibawah ini:

- Mempersiapkan semua hardware dan software yang diperlukan pada pengujian ini, yaitu seluruh model robot manipulator yang dibuat meliputi model hardware, Arduino Uno, kamera, dan PC.
- Sambungkan seluruh koneksi pada model yang telah dirancang ke sumber tegangan.
- Letakkan robot manipulator pada lokasi yang diinginkan.
- Setelah kamera fokus pada titik pusat bidang gambar pada koordinat (0,0), objek bergerak pada koordinat tertentu dan amati pergerakan robot manipulator apakah masing-masing servo dapat bekerja dengan baik.
- Amati hasil pengujian yang didapatkan secara real-time.

3. Hasil dan Pembahasan

3.1 Hasil Perancangan dan Pembuatan Robot Manipulator

Perancangan robot manipulator pada penelitian ini dilakukan dengan beberapa proses yang dilakukan beberapa proses manufaktur dengan menggunakan mesin 3D Printing dengan menggunakan filamen berbahan Polylactic Acid (PLA), PLA merupakan material filamen yang digunakan pada proses pencetakan 3D. PLA memiliki sifat mekanik serta kimia yang baik serta memiliki sifat bahan yang murah dan ramah terhadap lingkungan.sebagai bahan penyusun, dan dilanjutkan dengan proses perakitan (assembly). Porses assembly ini bertujuan untuk menyatukan bagian-bagian yang telah difabrikasi menjadi satu kesatuan produk yang utuh.



(a)



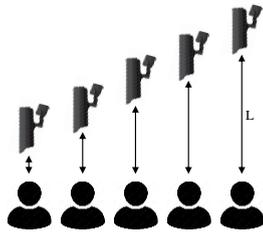
(b)

Gambar 2 Perancangan (a) desain akhir rancangan produk, dan (b) produk hasil manufaktur

Pada dasarnya, terdapat tiga tahapan utama pada proses perancangan untuk menjadi hasil akhir yang ditunjukkan pada Gambar 20. Tahapan pertama dimulai dengan desain pada software Solidworks 2020, kemudian dilanjutkan dengan proses manufaktur fabrikasi dengan menggunakan 3D Printing. Hasil perancangan robot manipulator 2 DOF ditunjukkan pada Gambar 2.

3.2 Hasil Pengujian Jarak Optimal Kamera

Pengujian jarak optimal kamera dilakukan dengan melihat pembacaan objek pada sistem deteksi wajah. Skema pengujian jarak optimal kamera ditunjukkan pada Gambar 3. Pada pengujian jarak optimal kamera, kamera yang sudah diletakkan pada ketinggian 115 dari permukaan lantai lokasi pengujian, alat percobaan yang kemudian diatur jarak alat terhadap objek sesuai dengan variasi jarak (L) yang diuji. Hasil pengujian jarak optimal kamera ditunjukkan pada Tabel 2.



Gambar 3 Skema pengujian jarak optimal kamera

Tabel 2 Pengujian variasi jarak kamera

No.	Jarak (cm)	Hasil
1.	50	Berhasil mendeteksi objek
2.	100	Berhasil mendeteksi objek
3.	150	Berhasil mendeteksi objek
4.	200	Berhasil mendeteksi objek dengan akurasi rendah

Berdasarkan hasil pengujian jarak optimal kamera pada Gambar 4.3 dapat dilihat bahwa program deteksi berhasil mendeteksi objek pada variasi jarak 50 cm, sistem mendeteksi objek dengan sangat baik, namun jarak objek terhadap alat pada 50 cm yang terhitung sangat dekat dengan alat menyebabkan berkurangnya jangkauan kemampuan sistem yang dapat lebih dimaksimalkan. Pada variasi jarak 100 cm sistem dapat dengan konsisten dalam mendeteksi. Pada variasi jarak 150 cm, program dapat mendeteksi objek dengan kualitas yang konsisten dalam mendeteksi objek dengan jangkauan yang luas. Pada variasi jarak 200 cm program hanya dapat mendeteksi objek dengan akurasi yang sangat rendah sehingga jarak yang dipilih sebagai dasar pengujian adalah jarak kamera 150 cm dari objek.

3.3 Pengujian Penggunaan Nilai *Threshold*

Pada pengujian ini, dilakukan pengujian untuk menentukan nilai *threshold* yang tepat untuk menguji keandalan rancangan sistem yang telah dibuat. *Threshold* digunakan untuk membuat toleransi piksel yang membatasi jika simpangan piksel sudah melewati nilai *threshold* maka servo pada sumbu x maupun sumbu y akan bergerak sesuai dengan arah simpangan yang dibutuhkan untuk memosisikan objek untuk selalu pada bidang gambar. Besaran nilai *threshold* dapat ditentukan secara bebas sebagai parameter pengujian yang akan mempengaruhi hasil akurasi dan kualitas sistem deteksi objek. Data hasil pengujian nilai *threshold* ditunjukkan pada Tabel 4

Berdasarkan hasil pengujian pemilihan nilai *threshold* yang disajikan pada Tabel 4, dapat diketahui bahwa pada pengujian dengan menggunakan nilai *threshold* sebesar 25, gerak sistem yang dihasilkan tidak stabil karena toleransi piksel yang terlalu kecil, sehingga dengan adanya fluktuasi piksel pendeteksian objek pada sumbu x maupun sumbu y membuat hasil gerak yang tidak stabil. Selanjutnya pada pengujian dengan menggunakan nilai *threshold* sebesar 50, hasil gerak sistem yang dihasilkan cukup stabil pada sumbu x, tetapi terjadi sedikit fluktuasi pada sumbu y yang menyebabkan adanya noise pada gerak sistem pendeteksian pada sumbu y. Pada pengujian dengan menggunakan nilai *threshold* sebesar 75, gerakan yang dihasilkan sangat stabil dan memiliki akurasi tinggi untuk menempatkan objek pada tengah bidang gambar sesuai dengan tujuan penelitian. Pada pengujian dengan menggunakan nilai *threshold* sebesar 100 piksel, hasil gerak yang dihasilkan sangat stabil namun memiliki akurasi yang rendah untuk menempatkan objek pada tengah bidang gambar karena adanya toleransi piksel yang terlalu besar sebanyak 100 piksel dari pusat bidang gambar. Maka, pada penelitian ini digunakan nilai *threshold* sebesar 75 piksel untuk mendapatkan stabilitas dan akurasi yang tinggi.

Tabel 3 Data hasil pengujian nilai *threshold*

No.	Nilai <i>Threshold</i>	Hasil
1.	25	Tidak stabil
2.	50	Cukup stabil
3.	75	Stabil
4.	100	Stabil

3.3.1 Kesimpulan Hasil Pengujian

Setelah melakukan rangkaian pengujian untuk menguji keandalan sistem penjejak wajah, penulis mendapatkan data yang dirangkum kedalam sebuah tabel yang meliputi seluruh hasil pengujian yang didapat. Data kesimpulan rangkaian pengujian ditunjukkan pada Tabel 4.

Tabel 4 Hasil pengujian implementasi deteksi wajah dengan variasi kondisi

Kondisi	Hasil
Objek (wajah) berada pada jarak 50 cm	Berhasil mendeteksi objek
Objek (wajah) berada pada jarak 100 cm	Berhasil mendeteksi objek
Objek (wajah) berada pada jarak 150 cm	Berhasil mendeteksi objek
Objek (wajah) berada pada jarak 200 cm	Berhasil mendeteksi objek dengan akurasi rendah

4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan maka dapat diambil beberapa kesimpulan, diantaranya adalah sebagai berikut:

1. Perancangan dan pembuatan sistem visual servoing dengan robot manipulator dengan dua derajat kebebasan gerak mampu menjalankan fungsi prinsip dasar visual servoing dengan baik. perancangan produk dilakukan dengan hasil yang baik. Perancangan sistem pemrograman dilakukan dengan menggunakan bahasa Python pada media aplikasi Visual Studio Code. Pemrograman dilakukan dengan melakukan import library yang diperlukan yang dilanjutkan dengan pembuatan algoritma deteksi hingga pergerakan masing-masing servo yang lengkap dengan notasi gambar untuk memberikan keterangan proses deteksi hingga menghasilkan sistem visual servoing secara keseluruhan.
2. Rangkaian pengujian yang telah dilakukan menghasilkan kualitas program deteksi dengan reliabilitas dan akurasi yang tinggi untuk mendeteksi objek. Pemilihan penggunaan alat dan bahan dengan spesifikasi yang sesuai dengan tujuan penelitian dapat menghasilkan program deteksi objek yang dapat mendeteksi wajah dengan baik pada objek diam dan bergerak.
3. Sistem object detection yang dirancang dapat beroperasi dengan baik sehingga dapat menghasilkan gerak lengan robot manipulator yang dengan tingkat akurasi yang tinggi dalam setiap gerakan yang dilakukan oleh objek pada setiap variasi pengujian.
4. Gerak lengan robot manipulator dapat beroperasi dengan baik pada setiap tahapan pengujian yang menghasilkan hasil yang optimal dengan jarak pengujian optimal 150 cm dan nilai threshold sebesar 75 piksel. Sistem dapat memanipulasi objek agar selalu di tengah bidang gambar tanpa adanya noise pada sumbu x maupun sumbu y di dalam toleransi simpangan (threshold).

DAFTAR PUSTAKA

- [1] Kang, M ., Chen, H., & Dong, J. (2020). Adaptive visual servoing with an uncalibrated camera using extreme learning machine and Q-learning. *Neurocomputing journal*. 384-394.
- [2] Gorethai, P., Hartoyo, A., & Suryadi, D. (2019). IMPLEMENTASI VISUAL SERVO MENGGUNAKAN PENGENDALI PID PADA ROBOT MANIPULATOR 2 DOF UNTUK APLIKASI PENJEJAKAN WAJAH. *Jurnal Fakultas Teknik Universitas Tanjungpura*.
- [3] Corke, P. I. (1996). *VISUAL CONTROL OF ROBOTS: High-Performance Visual Servoing*. CSIRO Division of Manufacturing Technology, Australia.
- [4] Chaumette, F., & Hutchinson, S. (2007). Visual Servo Control Part I : Basic Approaches. *IEEE Robotics & Automation Magazine*. 82-90.
- [5] Burlacu, A., Cosmin, C., Panainte, A., Pascal, C., & Lazar, C. (2011). REAL-TIME IMAGE BASED VISUAL SERVOING ARCHITECHTURE FOR MANIPULATOR ROBOTS. DOI : 10.5220/0003367005020510.
- [6] Chen, K., Ye, C., Wu, C., Wang, H., Jin, L., Zhu, F., & Hong, H., (2023). Novel Open-Closed-Loop Control Strategy for Quadrotor Trajectory Tracking on Real-Time Control and Acquisition Platform. *Appl. Sci.* 2023, 13, 3251. <https://doi.org/10.3390/app13053251>.
- [7] Gorethai, P., Hartoyo, A., & Suryadi, D. (2019). IMPLEMENTASI VISUAL SERVO MENGGUNAKAN PENGENDALI PID PADA ROBOT MANIPULATOR 2 DOF UNTUK APLIKASI PENJEJAKAN WAJAH. *Jurnal Fakultas Teknik Universitas Tanjungpura*
- [8] Varjão, J., Almeida, T. Citrus Fruit Quality Classification using Support Vector Machines, *International Journal of Advanced Engineering Research and Science (IJAERS)* Vol-6, Issue-7, Jul- 2019, <https://dx.doi.org/10.22161/ijaers.678> ISSN: 2349-6495(P) | 2456-1908(O)
- [9] Gajjar, H., Sanyal, S., Shah, M. (2023). A comprehensive study on lane detecting autonomous car using computer vision. *Expert Systems With Applications* 233 (2023) 120929. DOI: <https://doi.org/10.1016/j.eswa.2023.120929>.
- [10] He, Y., Gao, J., & Chen, Y. (2022). Deep learning-based pose prediction for visual servoing of robotic manipulators using image similarity. *Neurocomputing* 491 (2011) 343-352. <https://doi.org/10.1016/j.neucom.2022.03.045>. 0925-2312 2022 Elsevier B.V.