



E-ISSN 3032-601X & P-ISSN 3032-7105

Vol. 1, No. 4, Tahun 2024

MISTER

**Journal of Multidisciplinary Inquiry in Science,
Technology and Educational Research**

**Jurnal Penelitian Multidisiplin dalam Ilmu
Pengetahuan, Teknologi dan Pendidikan**

**UNIVERSITAS SERAMBI MEKKAH
KOTA BANDA ACEH**

mister@serambimekkah.ac.id

Journal of Multidisciplinary Inquiry in Science Technology
and Educational Research

Journal of MISTER

Vol. 1, No. 4, Tahun 2024

Pages: 2120–2134

Full-Body Tracking Berbasis OpenCV dan MediaPipe untuk Interaksi Objek Virtual

Ria Amelia Shinta Putricia Hendra, Velian Prapatoni, Bahiskara Ananda
Arryanto, Anggraini Puspita Sari

Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan
Nasional “Veteran” Jawa Timur, Surabaya, Indonesia

Article in Journal of MISTER

Available at : <https://jurnal-serambimekkah.org/index.php/mister/index>

DOI : <https://doi.org/10.32672/mister.v1i4.2212>

How to Cite this Article

APA : Putricia Hendra, R. A. S., Prapatoni, V., Arryanto, B. A., & Sari, A. P. (2024). Full-Body Tracking Berbasis OpenCV dan MediaPipe untuk Interaksi Objek Virtual. *Journal of Multidisciplinary Inquiry in Science, Technology and Educational Research*, 1(4), 2120 – 2134. <https://doi.org/10.32672/mister.v1i4.2212>

Others Visit : <https://jurnal-serambimekkah.org/index.php/mister/index>

MISTER: *Journal of Multidisciplinary Inquiry in Science, Technology and Educational Research* is a scholarly journal dedicated to the exploration and dissemination of innovative ideas, trends and research on the various topics include, but not limited to functional areas of Science, Technology, Education, Humanities, Economy, Art, Health and Medicine, Environment and Sustainability or Law and Ethics.

MISTER: *Journal of Multidisciplinary Inquiry in Science, Technology and Educational Research* is an open-access journal, and users are permitted to read, download, copy, search, or link to the full text of articles or use them for other lawful purposes. Articles on Journal of MISTER have been previewed and authenticated by the Authors before sending for publication. The Journal, Chief Editor, and the editorial board are not entitled or liable to either justify or responsible for inaccurate and misleading data if any. It is the sole responsibility of the Author concerned.



ISSN 3032-7105

ISSN 3032-601X



Full-Body Tracking Berbasis OpenCV dan MediaPipe untuk Interaksi Objek Virtual

**Ria Amelia Shinta Putricia Hendra¹, Velian Prapatoni², Bahiskara Ananda Arryanto³,
Anggraini Puspita Sari^{4*}**

Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional “Veteran”
Jawa Timur, Surabaya, Indonesia^{1,2,3,4}

*Email Korespondensi: anggraini.puspita.if@upnjatim.ac.id

Diterima: 03-09-2024

| Disetujui: 04-09-2024

| Diterbitkan: 05-09-2024

ABSTRACT

This research introduces a system that tracks full body movements in real-time to interact with virtual objects by combining OpenCV and MediaPipe in the Unity3D game engine. The system aims to overcome the drawbacks of current tracking solutions, which typically need unique hardware and are complicated, thus restricting their usability. The suggested method uses OpenCV for capturing and preparing images, while MediaPipe Pose is chosen for its precise and efficient real-time body landmark detection. The information on the user's body position is sent to Unity3D through a named pipe system, allowing accurate management of a 3D character's actions. Tests on two devices with varying hardware specifications indicated that the system successfully monitors body positions and movements in real-time, enabling interactive interaction with virtual objects in Unity3D. An examination of performance indicated that the processing speed, accuracy of landmark detection, and frame rate are all notably affected by the hardware specifications, especially the processor and GPU. Devices with higher specifications significantly offered a more seamless and speedy user experience. The study suggests that merging OpenCV and MediaPipe provides a precise and effective method for tracking full body movements, suitable for different interactive settings like virtual and augmented reality.

Keywords: Full-body tracking; Object interaction; OpenCV; MediaPipe; Unity3D.

ABSTRAK

Penelitian ini memperkenalkan sistem yang melacak gerakan seluruh tubuh secara *real-time* untuk berinteraksi dengan objek virtual dengan menggabungkan OpenCV dan MediaPipe dalam *game engine* Unity3D. Sistem ini dirancang untuk mengatasi kekurangan solusi pelacakan yang ada, yang biasanya membutuhkan perangkat keras khusus dan rumit, sehingga membatasi penggunaannya. Metode yang diusulkan memanfaatkan OpenCV untuk menangkap dan mempersiapkan gambar, sementara MediaPipe Pose dipilih karena kemampuannya dalam mendeteksi *landmark* tubuh secara *real-time* dengan akurat dan efisien. Informasi posisi tubuh pengguna kemudian dikirim ke Unity3D melalui sistem *named pipe*, yang memungkinkan kontrol gerakan karakter 3D yang presisi. Pengujian pada dua perangkat dengan spesifikasi perangkat keras yang berbeda menunjukkan bahwa sistem ini berhasil melacak posisi dan gerakan tubuh secara *real-time*, memungkinkan interaksi interaktif dengan objek virtual di Unity3D. Evaluasi kinerja menunjukkan bahwa kecepatan pemrosesan, akurasi deteksi *landmark*, dan *frame rate* dipengaruhi secara signifikan oleh spesifikasi perangkat keras, terutama prosesor dan GPU. Perangkat dengan spesifikasi yang lebih tinggi memberikan pengalaman pengguna yang lebih lancar dan cepat. Penelitian ini menyimpulkan bahwa menggabungkan OpenCV dan MediaPipe menyediakan metode yang akurat dan efektif untuk melacak gerakan seluruh tubuh, cocok untuk berbagai lingkungan interaktif seperti *virtual reality* dan *augmented reality*.

Katakunci: Pelacakan seluruh tubuh; Interaksi objek; OpenCV; MediaPipe; Unity3D.

PENDAHULUAN

Perkembangan teknologi saat ini banyak ditemukan dalam bidang interaksi manusia-komputer, khususnya dalam pengembangan sistem yang memungkinkan pengguna untuk berinteraksi dengan komputer secara lebih alami dan intuitif. Oleh karena itu, pengembangan teknologi ini perlu melampaui aspek teoritis dan direalisasikan dalam aplikasi praktis. Salah satu teknologi yang semakin penting dalam interaksi antara manusia dan komputer adalah *full-body tracking*, yang memungkinkan pelacakan pergerakan seluruh tubuh secara *real-time*. Peran signifikan teknologi ini semakin krusial dengan adanya kemajuan pesat dalam bidang *virtual reality* (VR) dan *augmented reality* (AR), terutama dalam industri game dan animasi (Anthes, 2016). Dalam game dan animasi, *full-body tracking* memungkinkan karakter virtual untuk bergerak dan berinteraksi dengan cara yang lebih alami dan realistis (Aristidou et al., 2018). Dalam konteks AR/VR, *full-body tracking* memungkinkan pengguna untuk berinteraksi dengan lingkungan virtual seolah-olah mereka benar-benar berada di dalamnya sehingga menciptakan pengalaman yang lebih imersif bagi pengguna.

Kemajuan signifikan yang dialami oleh teknologi *full-body tracking* memiliki beberapa keterbatasan yang perlu diatasi. Sistem *tracking* yang ada umumnya memerlukan perangkat keras khusus seperti sensor inframerah atau pakaian dengan sensor, yang meningkatkan biaya dan kompleksitas penggunaan. Selain itu, banyak teknologi pelacakan tubuh yang ada memiliki keterbatasan dalam hal akurasi dan kinerja di berbagai kondisi lingkungan. Kebutuhan akan solusi *full-body tracking* yang akurat, efisien, mudah digunakan, dan tanpa memerlukan perangkat keras khusus menjadi krusial.

Tujuan utama dari penelitian ini adalah mengembangkan metode *full-body tracking* yang lebih akurat dan efisien menggunakan kombinasi OpenCV dan MediaPipe. Penelitian ini merupakan pengembangan dari project open-source berjudul "*Multithreaded Unity Python MediaPipe Body/Pose*" yang menggunakan MediaPipe untuk pelacakan pose tubuh ke dalam Unity (Ganeshsar, 2023). Penelitian ini bertujuan untuk melampaui fungsionalitas sistem sebelumnya dengan mengevaluasi performa sistem secara komprehensif dan memperluas fungsinya untuk mendukung interaksi objek virtual yang lebih kompleks, yang memungkinkan interaksi antara pengguna dan objek virtual dalam lingkungan AR/VR. Dengan demikian, penelitian ini diharapkan dapat mengatasi keterbatasan teknologi tracking dan menyediakan solusi yang lebih terjangkau dan mudah diimplementasikan.

Tinjauan terhadap penelitian sebelumnya menunjukkan bahwa berbagai teknologi telah digunakan untuk *full-body tracking*, termasuk Microsoft Kinect, *motion capture suits*, dan metode visi komputer berbasis kamera. Kinect menggunakan sensor inframerah dan kedalaman, namun memerlukan perangkat keras khusus (Clark et al., 2013). *Motion capture suits* menawarkan akurasi tinggi tetapi memerlukan biaya yang signifikan dan persiapan yang rumit. Metode *computer vision* yang menggunakan kamera biasa dengan bantuan algoritma pemrosesan citra, menawarkan fleksibilitas tinggi dan biaya rendah, tetapi sering menghadapi tantangan dalam hal akurasi dan kinerja di berbagai kondisi lingkungan. Terlebih lagi, menjalankan *object detection* berbasis Machine Learning pada frame rate penuh (misalnya, 30 FPS) seringkali membutuhkan sumber daya komputasi yang tinggi, yang dapat menjadi kendala pada perangkat dengan sumber daya terbatas (Lugaresi et al., 2019).

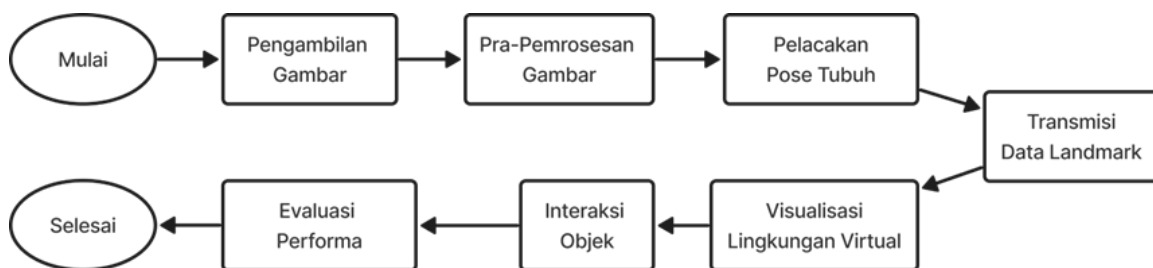
Dalam studi terkini, empat pustaka utama yang sering dibandingkan adalah OpenPose, PoseNet, MoveNet, dan MediaPipe Pose. MoveNet memiliki performa paling unggul secara keseluruhan untuk citra dan video, diikuti oleh OpenPose, PoseNet, dan MediaPipe Pose. Namun, MediaPipe Pose menunjukkan keunggulan khusus dalam pemrosesan video dengan nilai PDJ maksimum dan median tertinggi (Chung et

al., 2022). MediaPipe, dengan kemampuannya dalam deteksi keypoint yang presisi dan efisiensi pemrosesan, diakui sebagai solusi yang unggul dalam pemrosesan video *real-time*.

Selain penelitian mengenai pelacakan seluruh tubuh, penelitian di bidang deteksi mata juga memiliki relevansi dengan kemajuan teknologi visi komputer. Penelitian terkini menunjukkan potensi deteksi kedipan mata yang akurat dan responsif dengan menggunakan landmark wajah, khususnya Dlib's 68 *Facial Landmark*. Metode ini mampu mencapai akurasi tinggi dan respons cepat dalam kondisi nyata, tanpa kompleksitas komputasi dan persyaratan data besar yang terkait dengan jaringan saraf tiruan (Dewi et al., 2022). Penelitian lainnya menunjukkan hasil yang menjanjikan dengan memanfaatkan landmark MediaPipe *Face Mesh*. Pengujian sistem pendeteksi mata dilakukan dengan menggunakan video yang diambil menggunakan smartphone, pengujian tersebut dilakukan pada 56 video dengan durasi sekitar 5-10 detik. Penelitian ini menunjukkan akurasi 100% dalam mendeteksi mata terbuka dan tertutup, bahkan dalam kondisi pencahayaan yang kurang optimal (Rasyid et al., 2023). Hasil ini mengindikasikan bahwa deteksi mata berbasis landmark wajah telah menjadi alat yang efektif untuk aplikasi yang membutuhkan pengenalan gerakan mata, seperti deteksi kedipan mata.

Penelitian ini akan menggabungkan keunggulan OpenCV dan MediaPipe untuk menciptakan solusi full-body tracking yang efektif. OpenCV adalah pustaka pemrosesan citra yang serbaguna, sementara MediaPipe adalah kerangka kerja yang efisien untuk pemrosesan media multi-modal yang dikembangkan oleh Google. Kombinasi kedua pustaka ini diharapkan dapat menghasilkan sistem pelacakan tubuh yang tidak hanya akurat tetapi juga dapat diimplementasikan secara *real-time* dan efisien dalam penggunaan sumber daya. Kontribusi utama dari penelitian ini adalah pengembangan metode yang lebih efisien untuk pelacakan tubuh yang dapat diterapkan dalam berbagai aplikasi interaktif, terutama di platform Unity3D.

METODE PENELITIAN



Gambar 1. Alur Penelitian

Penelitian dimulai dengan memanfaatkan kamera digital sebagai sensor visual utama untuk menangkap gambar secara *real-time*. Data visual diproses menggunakan OpenCV dan MediaPipe. OpenCV bertugas untuk mengambil dan mengolah gambar, sedangkan MediaPipe menangani ekstraksi pose tubuh secara *real-time* dan melakukan segmentasi gambar. Pada tahap pelacakan pose tubuh, MediaPipe mendeteksi dan melacak titik-titik kunci (*keypoints*) pada tubuh manusia. *Keypoints* ini mewakili titik-titik penting pada tubuh, seperti sendi bahu, siku, pergelangan tangan, pinggul, lutut, dan pergelangan kaki. MediaPipe kemudian menghubungkan *keypoints* tersebut untuk membentuk kerangka tubuh (*skeleton*), yang merepresentasikan pose tubuh pengguna secara *real-time*.

Data landmark tubuh yang dihasilkan dari proses pelacakan pose tubuh kemudian dikirimkan ke *game engine* Unity3D melalui *named pipe*, memungkinkan pemetaan gerakan pengguna ke dalam lingkungan virtual. Data yang dikirimkan secara *real-time* dari program Python ke Unity3D memungkinkan karakter tiga dimensi bergerak sesuai dengan gerakan tubuh pengguna, menciptakan representasi virtual yang akurat dari gerakan fisik pengguna. Interaksi antara pengguna dan objek virtual di Unity3D diimplementasikan melalui deteksi *collision* dan komponen *Rigidbody*, yang dirancang untuk menciptakan pengalaman yang alami dan intuitif dalam lingkungan virtual tiga dimensi. Dengan begitu, pengguna dapat berinteraksi dengan objek virtual secara realistis seolah-olah berada dalam dunia virtual tersebut.

1) Pengambilan Gambar

Proses pengambilan data secara *real-time* untuk pelacakan gestur tubuh dilakukan menggunakan kamera digital. Kamera ini berfungsi sebagai sensor visual utama, yang bertugas untuk menangkap rangkaian gambar secara kontinu selama program dijalankan. Kemampuan kamera dalam merekam pergerakan tubuh secara terus-menerus menjadi krusial, karena memungkinkan program untuk memperoleh representasi yang akurat dari setiap gerakan yang dilakukan.

2) Pra-pemrosesan Citra

Tahap pra-pemrosesan citra bertujuan untuk meningkatkan kualitas data visual yang diperoleh dari kamera, sehingga memudahkan proses identifikasi dan pelacakan gestur tubuh. Dalam penelitian ini, OpenCV diimplementasikan untuk mendeteksi dan mengolah gambar. Tujuan utama penggunaan library ini adalah untuk mendapatkan gambar gestur tubuh melalui kamera. Selain OpenCV, program *full-body tracking* ini juga memanfaatkan kapabilitas framework MediaPipe untuk ekstraksi pose tubuh secara *real-time*. *Open-Source Computer Vision* (OpenCV) merupakan *library open-source* yang dirancang untuk pengolahan gambar dan video, dengan tujuan memberikan kemampuan pengolahan visual pada komputer yang mirip dengan cara kerja penglihatan manusia. OpenCV menyediakan banyak algoritma visi komputer dasar serta modul pendeteksian objek yang menggunakan metode *computer vision* (Makahaube et al., 2021).

Modul pra-pemrosesan MediaPipe berperan penting dalam mempersiapkan data citra sebelum diproses oleh model machine learning untuk deteksi pose. Modul ini secara otomatis menangani berbagai aspek, seperti penormalan citra, penyesuaian pencahayaan, dan transformasi geometri, untuk memastikan keseragaman data dan meningkatkan akurasi model. Penggunaan framework MediaPipe yang terintegrasi memungkinkan program untuk fokus pada ekstraksi pose tanpa perlu mengimplementasikan tahapan pra-pemrosesan secara manual. Hal ini menyederhanakan pengembangan dan meningkatkan efisiensi program secara keseluruhan. Sebagai contoh, studi oleh Ullah et al. membandingkan algoritma pelacakan orang menggunakan OpenCV dan menunjukkan keunggulan dalam berbagai kondisi pengamatan (Ullah et al., 2019).

3) Pelacakan Pose Tubuh

Pelacakan pose tubuh dalam sistem ini memanfaatkan kemampuan *Human Pose Estimation*, sebuah bidang krusial dalam *computer vision* yang berfokus pada deteksi dan analisis postur serta gerakan tubuh manusia (Josyula, 2021). Dalam penelitian ini, deteksi dan pelacakan pose tubuh dilakukan dengan memanfaatkan MediaPipe *Pose*, sebuah *framework machine learning open-source* yang dikembangkan

oleh Google (Pranav, 2024). MediaPipe Pose menyediakan solusi siap pakai untuk deteksi pose tubuh manusia secara *real-time* dari gambar atau video. Dalam gambar dan video, ada beberapa jenis estimasi pose, termasuk tubuh, wajah, dan tangan (Zulkhaidi et al., 2020). Informasi mengenai posisi landmark tubuh ini kemudian digunakan sebagai titik acuan untuk mendeteksi gerakan tracking oleh pengguna. Dengan adanya *library* MediaPipe, komputer dapat merespons pergerakan atau posisi tubuh secara cepat dan akurat, sehingga memungkinkan interaksi yang lebih efisien antara pengguna dan sistem.

MediaPipe memanfaatkan *Convolutional Neural Networks* (CNNs) dalam model-model pre-trained untuk membantu menyediakan model yang sudah dilatih dan siap digunakan, sehingga mempercepat proses pengembangan dan meningkatkan akurasi hasil. Selain itu, CNNs juga digunakan secara luas dalam berbagai bidang lain termasuk pengolahan gambar, *computer vision*, diagnostik medis, serta dalam kombinasi dengan LSTM untuk prediksi kecepatan dan arah angin (Sari et al., 2020) [15].

Tabel 1. Parameter Tracking MediaPipe Pose

Attribute	Value
Minimum Detection Confidence	0.8
Minimum Tracking Confidence	0.5
Model Complexity	1
Segmentation	True
Landmark Visibility Threshold	0.5
Capture Resolution	640 x 480 px (OpenCV default resolution)
Minimum Detection Confidence	0.8

MediaPipe, sebagai kerangka kerja machine learning yang digunakan untuk pelacakan tubuh penuh, membutuhkan konfigurasi parameter yang optimal untuk mencapai akurasi dan performa yang diharapkan. Dalam pengujian yang akan dilakukan, konfigurasi parameter MediaPipe diatur dengan mempertimbangkan keseimbangan antara akurasi, ketahanan pelacakan, dan beban komputasi. Parameter *Minimum Detection Confidence* diatur ke nilai 0.8, yang berarti sistem membutuhkan kepercayaan minimal 80% untuk menganggap suatu pose terdeteksi. Nilai yang lebih tinggi meningkatkan akurasi deteksi tetapi meningkatkan risiko deteksi terlewat pada kondisi ambigu. Parameter *Minimum Tracking Confidence* diatur ke 0.5 untuk menyeimbangkan antara ketahanan pelacakan dan akurasi. Nilai yang lebih rendah meningkatkan ketahanan pelacakan terhadap oklusi atau gerakan cepat, tetapi meningkatkan potensi kesalahan pelacakan.

Untuk menyeimbangkan antara akurasi dan beban komputasi, parameter *Model Complexity* diatur ke nilai 1. Model dengan kompleksitas lebih tinggi dapat meningkatkan akurasi tetapi membutuhkan sumber daya komputasi yang lebih besar. Segmentasi gambar diaktifkan (*Segmentation = True*) untuk memisahkan objek manusia dari latar belakang, meningkatkan akurasi deteksi, terutama pada lingkungan yang kompleks. Parameter *Landmark Visibility Threshold* diatur ke 0.5 sebagai nilai untuk pengujian *Confidence Score* dari sensitivitas deteksi landmark, dengan mempertimbangkan potensi kesalahan deteksi pada kondisi oklusi. Resolusi citra yang ditangkap kamera (*Capture Resolution*) diatur ke 640 x 480 pixel sesuai dengan pengaturan bawaan OpenCV untuk menyeimbangkan antara akurasi deteksi dan beban komputasi sistem.

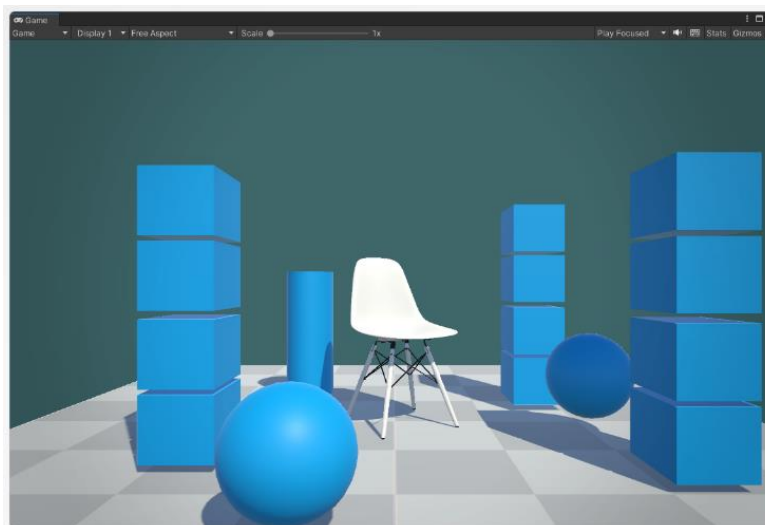
4) Pengiriman Pipe

Data landmark tubuh yang diperoleh dari MediaPipe Pose dapat ditransmisikan secara *real-time* ke game engine Unity3D melalui mekanisme *named pipe*, salah satu bentuk komunikasi antar-proses (*Inter-Process Communication* atau IPC) (Merwan,2024). Named pipe dipilih karena efisiensinya dalam menyediakan komunikasi inter-process *real-time* antara program Python yang menangani pemrosesan *full-body tracking* dengan *rendering* pada Unity3D. Skrip Python berperan sebagai server, membuka named pipe dan mengirimkan data landmark tubuh dalam format string yang telah ditentukan.

Di Unity, program C# yang berperan sebagai *client* (Vetter, 2019), membuka koneksi ke *named pipe* dan membaca data landmark tubuh. Data ini kemudian dipetakan ke model karakter tiga dimensi yang berupa kerangka tubuh manusia (*skeleton*). Pemetaan dilakukan dengan mencocokkan landmark tubuh dengan joint yang sesuai pada karakter di Unity3D. Program juga mengatur *transform* (posisi dan rotasi) dari *joint* pada model karakter berdasarkan pemetaan landmark, sehingga karakter bergerak sesuai dengan gerakan tubuh pengguna secara *real-time*.

5) Lingkungan Virtual Unity

Setelah gerakan tracking berhasil dipetakan dalam lingkungan Unity, langkah selanjutnya pada program ini adalah membuat lingkungan virtual untuk menampilkan data tracking yang terdeteksi sistem. Lingkungan virtual dalam Unity3D dirancang menyerupai ruang nyata tempat pengguna berinteraksi dengan sistem. Hal ini bertujuan untuk memudahkan pengguna dalam memahami hubungan antara gerakan tubuh mereka dengan objek virtual di dalam lingkungan tersebut. Dengan begitu Unity3D dapat digunakan untuk merekonstruksi dan memvisualisasikan gerakan pengguna dalam lingkungan virtual tiga dimensi.

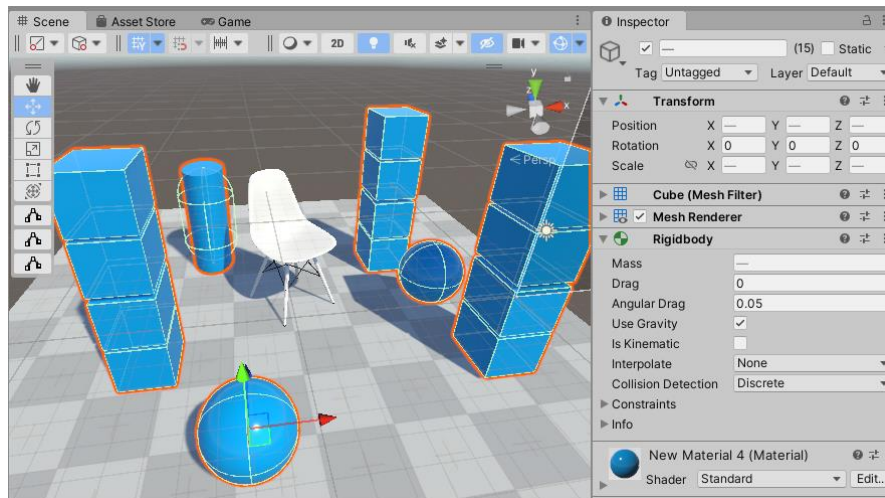


Gambar 2. Lingkungan virtual Unity3D

6) Komponen Rigidbody

Tahap selanjutnya adalah mengimplementasikan interaksi antara pengguna dan objek virtual di dalam lingkungan Unity3D. Gerakan tubuh pengguna, yang direkonstruksi dari data landmark pose tubuh, digunakan untuk mengontrol dan memanipulasi objek virtual. Objek-objek virtual dalam lingkungan

Unity3D diberikan komponen *Rigidbody* untuk mensimulasikan sifat fisik objek, seperti gravitasi, massa, dan koefisien gesek. Interaksi antara pengguna dan objek virtual diimplementasikan dengan mendeteksi *collision* antara model kerangka tubuh pengguna dengan objek virtual. Dengan menggabungkan *full-body tracking*, lingkungan virtual tiga dimensi, dan sifat fisik objek, sistem *full-body tracking* yang dibangun dapat menciptakan pengalaman interaksi yang natural dan intuitif antara pengguna dan objek virtual.



Gambar 3. Komponen Rigidbody pada objek

7) Perangkat Pengujian

Pengujian dilakukan dalam dua aspek utama. Pertama, mengevaluasi keberhasilan program dalam mendeteksi pose tubuh dan memperkirakan latensi sistem dalam menangkap, memproses, dan mengirimkan data ke Unity. Pengujian ini dilakukan pada dua perangkat berbeda untuk membandingkan performa sistem *full-body tracking* pada spesifikasi hardware yang berbeda. Spesifikasi kedua perangkat yang diuji dapat dilihat pada Tabel 2.

Tabel 2. Spesifikasi Perangkat Pengujian

Komponen	Perangkat 1	Perangkat 2
Processor	Intel Core i5-10400F CPU @ 2.9GHz (12CPUs), ~2,9GHz	Intel Core i5-8350U CPU @ 1.7GHz (8CPUs), ~1.9GHz
RAM	16384 MB	8192 MB
Graphic Card	NVIDIA GeForce GTX 1070	Intel UHD Graphics 620
VRAM	8067 MB	128 MB
Webcam	Logitech C922	Thinkpad T480 Webcam

8) Evaluasi Performa

Untuk menilai kinerja sistem *full-body tracking*, beberapa metrik kunci digunakan sebagai indikator kemampuan sistem dalam menangkap, memproses, dan menyampaikan gerakan pengguna secara real-time. Metrik ini memungkinkan peneliti untuk mendefinisikan tingkat responsivitas, akurasi, dan kelancaran sistem dalam melacak gerakan tubuh pengguna. Salah satu metrik utama adalah *capture latency*,

yang mengukur waktu yang dibutuhkan kamera webcam untuk menangkap gambar dan menyediakan data gambar tersebut ke prosesor untuk diproses. *Capture latency* yang lebih rendah mengindikasikan sistem yang lebih responsif terhadap gerakan pengguna, sehingga menghasilkan pengalaman interaksi yang lebih mulus.

Selain *capture latency*, *processing latency* juga menjadi fokus evaluasi. Metrik ini mengukur waktu yang dibutuhkan MediaPipe untuk mendeteksi dan melacak pose tubuh dari data gambar yang diterima. *Processing latency* yang lebih rendah menunjukkan sistem yang efisien dalam menganalisis gambar dan menghasilkan data landmark tubuh, yang pada akhirnya akan meningkatkan kecepatan dan ketepatan respon sistem terhadap gerakan pengguna. Kecepatan pengiriman data juga menjadi aspek penting yang diukur melalui *communication latency*, yaitu waktu yang dibutuhkan untuk mengirimkan data landmark tubuh dari modul pemrosesan program Python ke modul rendering Unity3D. Latency komunikasi yang rendah memastikan data dapat dikirimkan secara efisien tanpa menyebabkan jeda dalam visualisasi gerakan, sehingga pengalaman pengguna dalam berinteraksi dengan lingkungan virtual menjadi lebih lancar dan responsif.

Total latency, yang menggabungkan ketiga komponen latency di atas, mencerminkan keseluruhan waktu yang dibutuhkan dari momen kamera menangkap gambar hingga model karakter virtual dalam Unity3D merespon gerakan tersebut. Metrik ini memberikan gambaran lengkap tentang responsivitas sistem secara keseluruhan. Selain latency, akurasi deteksi juga menjadi aspek penting dalam evaluasi sistem. Akurasi deteksi diukur melalui dua metrik: *total visible landmarks*, yang menunjukkan jumlah landmark tubuh yang berhasil dideteksi dan dilacak oleh sistem, serta *average confidence score*, yang menyatakan tingkat keyakinan sistem terhadap akurasi deteksi landmark tubuh. Semakin tinggi jumlah *total visible landmarks* dan *confidence score*, semakin akurat sistem dalam mendeteksi dan melacak pose tubuh, sehingga gerakan pengguna dapat diinterpretasikan dengan lebih tepat.

Terakhir, *frame rate per second* merupakan metrik yang mengukur jumlah frame yang dapat diproses dan ditampilkan setiap detik. FPS yang tinggi mengindikasikan visualisasi gerakan yang lebih halus dan responsif, sehingga menciptakan pengalaman interaksi yang lebih realistis dan menyenangkan. Pengukuran ini penting untuk menentukan kelancaran dan kualitas visualisasi gerakan pengguna dalam lingkungan virtual. Dengan menganalisis semua metrik ini, peneliti dapat memperoleh pemahaman yang lebih komprehensif tentang performa sistem *full-body tracking*. Metrik *latency* menggambarkan responsivitas sistem terhadap gerakan pengguna. Akurasi deteksi, yang diukur melalui *total visible landmarks* dan *confidence score*, menentukan ketepatan sistem dalam mendeteksi dan melacak pose tubuh. *Frame rate*, pada gilirannya, mempengaruhi kelancaran visualisasi gerakan, dan berkontribusi terhadap kualitas pengalaman pengguna secara keseluruhan.

HASIL DAN PEMBAHASAN

Bagian ini menyajikan hasil implementasi dan pengujian sistem *full-body tracking* yang dikembangkan, serta analisis performa yang didapat. Pembahasan akan dilakukan berdasarkan hasil pengujian dan perbandingan performa sistem pada dua perangkat komputer dengan spesifikasi hardware yang berbeda. Representasi visual seperti tabel dan gambar akan digunakan untuk memperjelas penjelasan hasil pengujian.

1) Pengambilan Modul Library

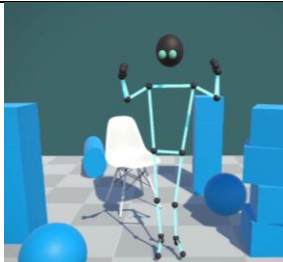
Sebelum memulai proses pengujian, tahap yang perlu dilakukan adalah mempersiapkan lingkungan pengembangan dan library yang dibutuhkan. Program *full-body tracking* dalam penelitian ini dikembangkan menggunakan bahasa pemrograman Python. Pengguna diharuskan menginstal versi Python yang kompatibel, direkomendasikan versi 3.8 ke atas, dan disarankan untuk menggunakan *virtual environment* Python seperti PyCharm agar pengelolaan library lebih terstruktur. Program ini bergantung pada beberapa library penting. MediaPipe, library inti dari sistem, menyediakan fungsionalitas untuk mendeteksi dan melacak landmark tubuh secara real-time. OpenCV digunakan untuk akuisisi dan pra-pemrosesan citra dari kamera. NumPy digunakan untuk manipulasi data numerik pada koordinat landmark tubuh. Library bawaan Python, csv dan struct, masing-masing digunakan untuk menyimpan data performa selama pengujian ke dalam file CSV dan membantu komunikasi data melalui named pipe dengan mengubah data menjadi format byte. Setelah lingkungan pengembangan siap, proses akuisisi dan pra-pemrosesan citra, dapat dilakukan.

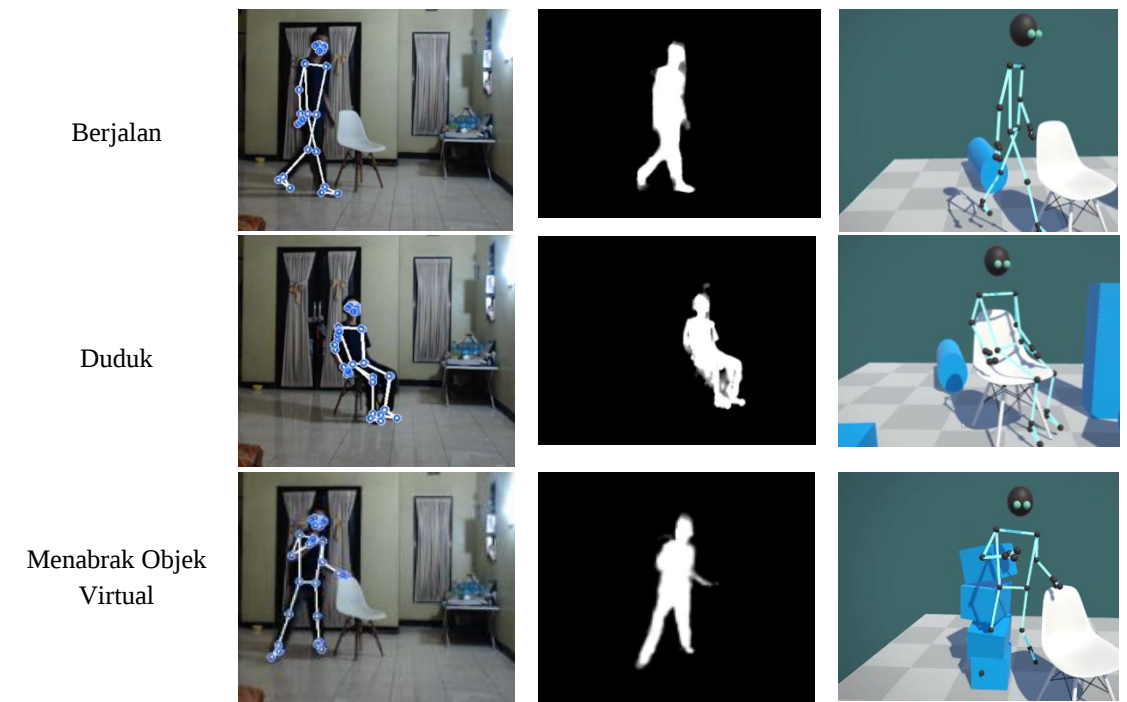
2) Pengujian Program

Tahap pengujian dilakukan dalam lingkungan pengembangan terintegrasi PyCharm untuk memastikan fungsionalitas program dan tidak adanya kesalahan yang menghambat eksekusi. Verifikasi ini bertujuan untuk memastikan bahwa program dapat berjalan dengan lancar dan tidak mengalami kendala teknis yang dapat mengganggu proses pengujian selanjutnya. Setelah memastikan fungsionalitas program, program *full-body tracking* kemudian diintegrasikan dengan perangkat lunak Unity untuk mengevaluasi kinerja *real-time*. Integrasi ini memungkinkan peneliti untuk menguji program dalam lingkungan yang lebih kompleks dan realistis, serta untuk melihat bagaimana program berinteraksi dengan komponen lain dalam sistem secara keseluruhan.

Pengujian pada lingkungan Unity melibatkan eksekusi langsung program untuk memverifikasi kemampuan sistem dalam mendeteksi dan melacak gestur tubuh secara *real-time*. Tujuan utama dari tahap ini adalah untuk memastikan bahwa sistem dapat secara akurat mengidentifikasi dan melacak landmark pose tubuh yang relevan untuk berbagai gerakan, yang merupakan prasyarat untuk fungsionalitas sistem secara keseluruhan. Proses pengujian ini dilakukan dengan melakukan berbagai gerakan tubuh yang telah ditentukan, dan hasilnya kemudian dianalisis untuk menentukan kemampuan sistem dalam mendeteksi dan melacak pose tubuh dengan tepat.

Tabel 3. Hasil Uji Coba Sistem *Full-body Tracking*

Gestur Tubuh	MediaPipe Pose	MediaPipe Segmentation	Unity3D
Berdiri			



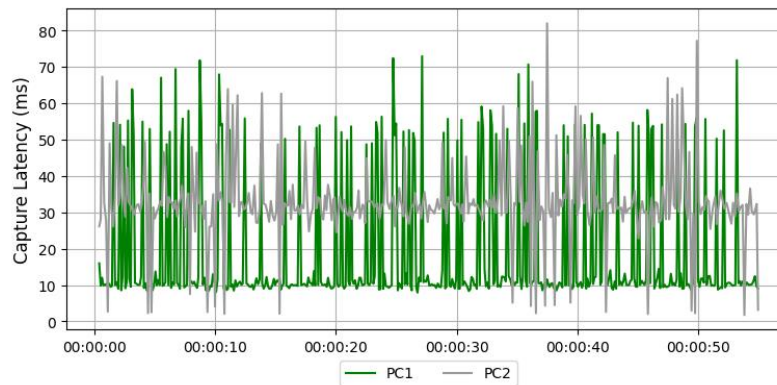
Dari hasil pengujian yang telah dijabarkan pada Tabel 4 menunjukkan bahwa MediaPipe mampu untuk melakukan *tracking* terhadap landmark pose dalam berbagai gestur tubuh yang dicoba secara langsung. Hasil *tracking* berupa pendeteksian gestur tubuh berdiri, berjalan, duduk, dan gerakan interaksi dengan objek virtual yang ada di Unity. Keberhasilan MediaPipe dalam mendeteksi dan melacak gestur tubuh ini menunjukkan bahwa sistem ini dapat digunakan untuk membangun berbagai aplikasi interaktif yang melibatkan interaksi antara pengguna dan objek virtual. Tahap selanjutnya melibatkan pengukuran latensi sistem, yang meliputi latensi pengambilan gambar, latensi pemrosesan gambar, dan latensi komunikasi data landmark dari MediaPipe ke Unity. Interval debugging untuk pengujian performa pelacakan disetel setiap 100 ms untuk mendapatkan pemahaman yang komprehensif tentang performa sistem *real-time*.

3) Laporan Pengujian

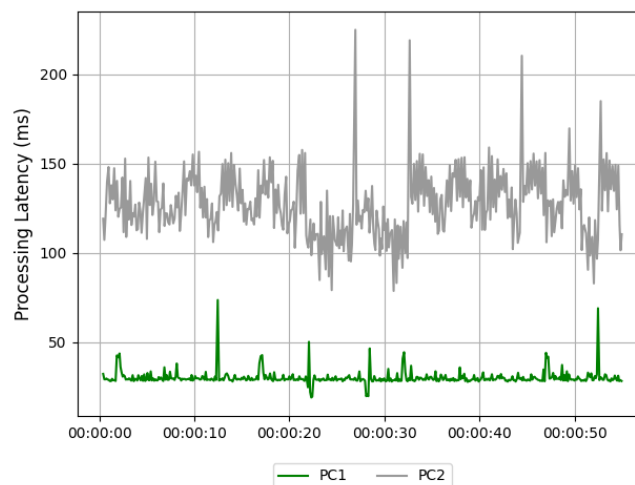
Sistem *full-body tracking* yang diimplementasikan menunjukkan performa yang cukup baik dalam mendeteksi pose tubuh untuk mendukung interaksi real-time, meskipun terdapat beberapa catatan penting yang perlu diperhatikan. Tabel 5 merangkum statistik performa sistem selama pengujian.

Tabel 4. Statistik Performa Sistem Full-body Tracking

Component	Minimum		Maximum		Average		Standard Deviation	
	PC 1	PC 2	PC 1	PC 2	PC 1	PC 2	PC 1	PC 2
Capture Latency (ms)	7.9	1.74	72.97	83.84	18.77	32.97	17.87	11.88
Processing Latency (ms)	19.13	78.77	73.7	224.91	30.17	128	3.94	18.54
Communication Latency (ms)	0.02	0.03	0.65	1.45	0.07	0.36	0.03	0.23
Total Latency (ms)	29.09	105.98	129.65	257.2	49	161.33	18.65	21.59
Total Visible Landmarks	0	0	33	33	31.2	24.95	4.45	12.47
Average Confidence	0	0	0.98	0.96	0.92	0.72	0.12	0.35
FPS	13.57	4.45	52.28	12.69	33.53	7.97	3.2	1.14



Gambar 4. Hasil Pengujian Capture Latency

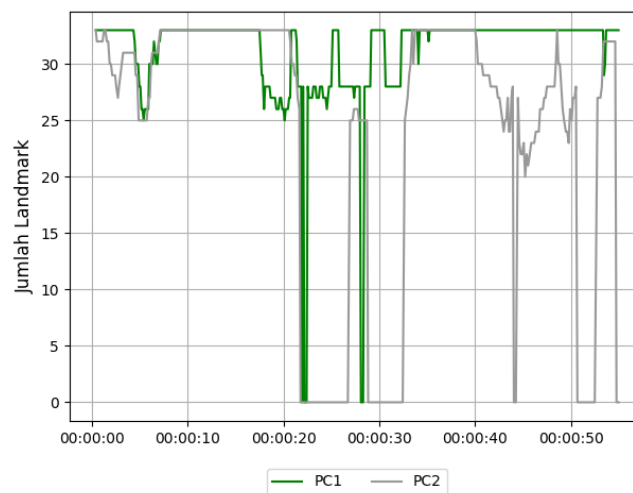


Gambar 5. Hasil Pengujian Processing Latency

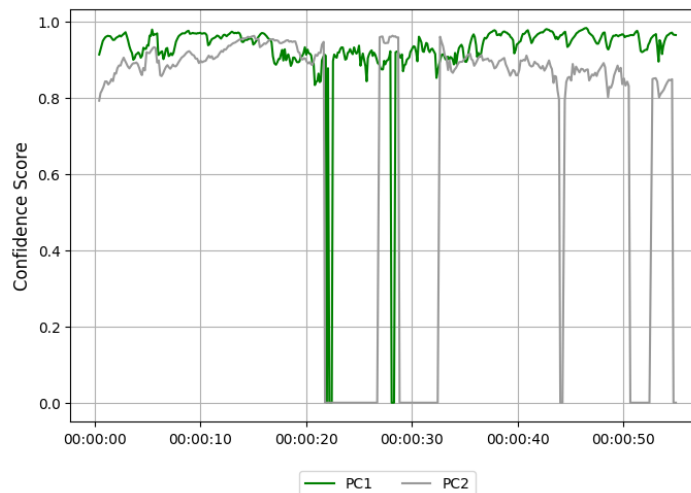
Pada pengujian *capture latency* yang ditunjukkan pada Gambar 4, PC1 menunjukkan rata-rata 18.77 ms dengan fluktuasi yang cukup tinggi ($SD = 17.87$ ms), mengindikasikan potensi inkonsistensi pada proses akuisisi data dari webcam. Meskipun menggunakan webcam dengan spesifikasi lebih rendah, PC2

menunjukkan rata-rata *capture latency* 32.97 ms dengan fluktuasi yang lebih stabil. Hasil ini menggarisbawahi bahwa selain faktor hardware, optimasi driver dan firmware juga berperan penting dalam meminimalisir *capture latency*.

Dari Gambar 5 di atas, rata-rata *processing latency* pada PC1 menunjukkan waktu pemrosesan yang jauh lebih rendah sebesar 30.17 ms, dibandingkan dengan PC2 sebesar 128 ms. Perbedaan signifikan ini menunjukkan bahwa prosesor dan GPU yang lebih tinggi pada PC1 mampu menangani kompleksitas komputasi dalam mendeteksi dan melacak landmark tubuh secara jauh lebih efisien. *Communication latency* yang mengukur waktu transmisi data landmark menunjukkan nilai yang sangat rendah dan serupa pada kedua perangkat, di bawah 0.37 ms. Hal ini mengindikasikan bahwa metode komunikasi *named pipe* yang diimplementasikan cukup efisien dan tidak menjadi suatu *bottleneck* dalam sistem.



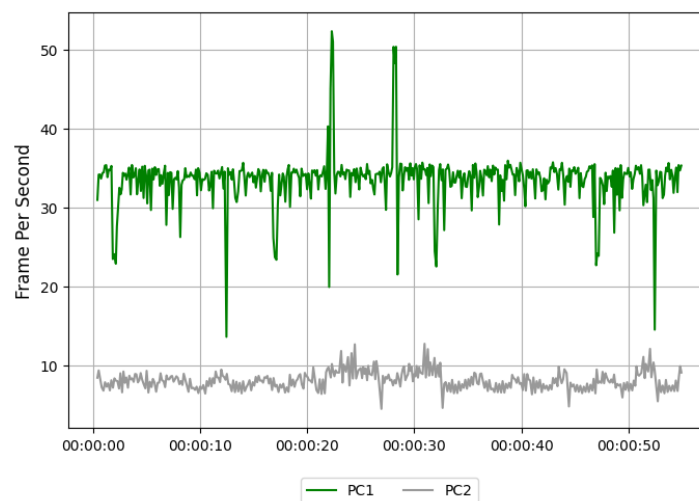
Gambar 6. Hasil Deteksi Landmark



Gambar 7. Hasil Pengujian Confidence Score

Gambar 6 menampilkan hasil deteksi landmark, sementara Gambar 7 menunjukkan hasil pengujian

confidence score. PC1 menunjukkan performa yang unggul dengan kemampuan mendeteksi rata-rata 31.2 dari 33 landmark tubuh yang didefinisikan oleh MediaPipe, sementara PC2 hanya mampu mendeteksi rata-rata 24.95 landmark. Kemampuan pemrosesan PC1 yang lebih baik memungkinkan algoritma MediaPipe untuk menganalisis data visual secara lebih komprehensif dan menghasilkan deteksi landmark yang lebih akurat. Skor kepercayaan (*Average Confidence Score*) yang mencerminkan tingkat keyakinan sistem terhadap akurasi deteksi landmark juga menunjukkan hasil yang konsisten. PC1 menunjukkan skor kepercayaan yang jauh lebih tinggi sebesar 92% dibandingkan PC2 yang hanya sebesar 72%, menguatkan bahwa PC1 tidak hanya mendeteksi lebih banyak landmark, tetapi juga lebih yakin terhadap keakuratan deteksinya. Dalam pengujian terdapat sedikit penurunan skor kepercayaan pada saat gerakan tubuh yang cepat, mengindikasikan ketidakpastian dalam deteksi landmark.



Gambar 8. Hasil Pengujian Frame Rate

Gambar 8 menampilkan hasil pengujian *frame rate* yang mengukur kelancaran visualisasi gerakan. Rata-rata *frame rate* yang dicapai oleh sistem *full-body tracking* pada PC1 mencapai 33.53 FPS, melampaui ambang batas minimal 24 FPS untuk menghasilkan visualisasi gerakan yang halus dan nyaman dipandang. Sebaliknya, PC2 hanya mencapai 7.97 FPS, menunjukkan *bottleneck* pada kemampuan pemrosesan yang mengakibatkan gerakan tersendat-sendat. Fluktuasi *frame rate* yang minimal pada PC1 (SD = 3.2 FPS) dan PC2 (SD = 1.14 FPS) menunjukkan bahwa sistem mampu mempertahankan performa yang relatif stabil selama pengujian.

KESIMPULAN

Berdasarkan tinjauan literatur dan pengujian yang dilakukan, MediaPipe Pose menunjukkan potensi yang signifikan sebagai solusi yang akurat dan efisien untuk sistem *full-body tracking real-time*, terutama untuk interaksi objek virtual dalam lingkungan Unity3D. MediaPipe Pose mampu mendeteksi dan melacak 33 landmark tubuh secara akurat dan efisien, bahkan dengan perangkat keras yang terbatas. Hal ini dibuktikan dengan hasil pengujian yang menunjukkan rata-rata 31.2 dari 33 landmark tubuh terdeteksi dengan skor kepercayaan (*confidence score*) mencapai 92% pada perangkat dengan spesifikasi menengah.

Kemampuannya menghasilkan frame rate yang relatif tinggi, mencapai rata-rata 33.53 FPS pada perangkat yang sama, semakin menegaskan efisiensinya. Kemudahan integrasinya dengan platform lain seperti OpenCV dan Unity3D dibuktikan dengan keberhasilan pengiriman data landmark tubuh melalui *named pipe* untuk mengendalikan gerakan model karakter tiga dimensi dalam Unity3D. Dibandingkan dengan sistem *full-body tracking* yang memerlukan perangkat keras khusus, MediaPipe Pose menawarkan alternatif yang lebih aksesibel dan hemat biaya karena hanya memerlukan kamera digital standar. Meskipun MediaPipe Pose menunjukkan performa yang baik pada perangkat dengan spesifikasi terbatas, perlu dicatat bahwa spesifikasi hardware yang lebih tinggi, terutama prosesor dan GPU, berkontribusi pada *processing latency* yang lebih rendah, akurasi deteksi landmark yang lebih tinggi, dan *frame rate* yang lebih halus, sehingga menghasilkan pengalaman pengguna yang lebih baik. Kesimpulannya, MediaPipe Pose merupakan solusi *full-body tracking* yang menjanjikan untuk berbagai aplikasi interaktif. Implementasinya yang relatif mudah, akurasi yang tinggi, dan kebutuhan hardware yang minimal menjadikannya pilihan yang ideal untuk pengembangan berbagai aplikasi AR/VR di masa depan.

DAFTAR PUSTAKA

- Anthes, C., García-Hernández, R. J., Wiedemann, M., & Kranzlmüller, D. (2016). State of the art of virtual reality technology. In *2016 IEEE Aerospace Conference* (pp. 1-19). Big Sky, MT, USA: IEEE. <https://doi.org/10.1109/AERO.2016.7500674>
- Aristidou, A., Lasenby, J., Chrysanthou, Y., & Shamir, A. (2018). Inverse kinematics techniques in computer graphics: A survey. *Computer Graphics Forum*, 37(6), 35-58. <https://doi.org/10.1111/cgf.13310>
- Chung, J.-L., Ong, L.-Y., & Leow, M.-C. (2022). Comparative analysis of skeleton-based human pose estimation. *Future Internet*, 14(380). <https://doi.org/10.3390/fi14120380>
- Clark, R. A., Pua, Y., Bryant, K., Hunt, P., & Clark, M. M. (2013). Validity and reliability of the Microsoft Kinect for providing lateral trunk lean feedback during gait retraining. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 21(4), 567-574. <https://doi.org/10.1109/TNSRE.2013.2260472>
- Dewi, C., Chen, R.-C., Jiang, X., & Yu, H. (2022). Adjusting eye aspect ratio for strong eye blink detection based on facial landmarks. *PeerJ Computer Science*, 8, e943. <https://doi.org/10.7717/peerj-cs.943>
- Ganeshsar. (2023). *Multithreaded Unity Python MediaPipe BodyPose*. GitHub. <https://github.com/ganeshsar/UnityPythonMediaPipeBodyPose>
- Josyula, R. (2021). *Human pose estimation* (Studi Independen Musim Panas). Northeastern University, Electrical and Computer Engineering Department, Augmented Cognition Laboratory (ACLab).
- Lugaresi, C., Tang, J., Nash, H., McClanahan, C., Uboweja, E., Hays, M., ... & Chang, M. (2019). MediaPipe: A framework for building perception pipelines. *Google Research*. arXiv preprint arXiv:1906.08172.
- Makahaube, S. S., Sambul, A. M., & Sompie, S. R. U. (2021). Implementation of gesture recognition technology for automated education service kiosk. *Universitas Sam Ratulangi*, 16(4), 465–472.
- Merwan, C. (2024, 27 Januari). Named pipes in .NET (C#). *Medium*. <https://medium.com/codenx/named-pipes-in-net-c-c0459e165371>
- Pranav, D. (2022, 1 Maret). MediaPipe: The ultimate guide to video

- processing. *LearnOpenCV*. <https://learnopencv.com/introduction-to-mediapipe/>
- Rasyid, F., Mustafa, M., Mahersatillah, A., Rizal, M., Mushaf, M., & Arifin, A. (2023). Deteksi mata di video smartphone menggunakan mediapipe python. *JOINTECS (Journal of Information Technology and Computer Science)*, 8(2), 49-56. <https://doi.org/10.31328/jointecs.v8i2.4562>
- Sari, A. P., Suzuki, H., Kitajima, T., Yasuno, T., Prasetya, D. A., & Nachrowie. (2020). Prediction model of wind speed and direction using convolutional neural network - long short term memory. In *2020 IEEE International Conference on Power and Energy (PECon)* (pp. 356-361). <https://doi.org/10.1109/PECon48942.2020.9314474>
- Ullah, K., Ahmed, I., Ahmad, M., & Khan, I. (2019). Comparison of person tracking algorithms using overhead view implemented in OpenCV. In *2019 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)* (pp. 15-20). Jaipur, India: IEEE. <https://doi.org/10.1109/IEMECONX.2019.8877025>
- Vetter, M. (2019). *A pattern for client/server communication via Named Pipes via C#* [Kode komputer]. GitHub. <https://github.com/malcomvetter/NamedPipes>
- Zulkhaidi, T. C. A.-S., Maria, E., & Yulianto, Y. (2020). Pengenalan pola bentuk wajah dengan OpenCV. *Jurnal Rekayasa Teknologi Informasi*, 3(2), 181. Samarinda: Politeknik Pertanian.