

Nama : Rifky Ihsanul Mahda

NIM : 234308109

Kelas : TKA 6D

Mata Kuliah : Praktikum Kontrol Cerdas

Akun Github : Rifky Ihsanul Mahda (Rifky109)

A. Pendahuluan

Python adalah salah satu bahasa pemrograman yang sering direkomendasikan untuk pemula karena cara penulisannya sederhana dan mudah dimengerti. Dengan sintaks yang tidak terlalu rumit, mahasiswa dapat lebih cepat memahami logika dan konsep dasar pemrograman tanpa harus kebingungan dengan aturan teknis yang kompleks. Selain itu, Python juga banyak digunakan dalam pengembangan *machine learning* dan *computer vision* karena memiliki berbagai library pendukung seperti OpenCV dan MediaPipe yang memudahkan proses pengolahan gambar dan video (Belajarpython.com, 2025).

Computer vision merupakan teknologi yang memungkinkan komputer untuk “melihat” dan memahami gambar maupun video secara otomatis. Melalui teknologi ini, komputer dapat mengenali objek, membaca pola, serta menganalisis pergerakan yang terdapat dalam suatu tampilan visual. Di Indonesia, penerapan *computer vision* telah banyak dilakukan dalam berbagai penelitian, seperti analisis gerakan squat untuk latihan kebugaran menggunakan metode estimasi pose berbasis MediaPipe (Dyansyah et al., 2023). Selain itu, teknologi ini juga dimanfaatkan untuk mendeteksi jarak wajah pengguna terhadap layar komputer sebagai bentuk peringatan kesehatan (Suradi et al., 2025).

Penggunaan MediaPipe juga berkembang dalam bidang pengenalan gestur dan bahasa isyarat. Penelitian mengenai deteksi alfabet BISINDO menggunakan MediaPipe Holistic menunjukkan bahwa sistem dapat mengenali gerakan tangan secara *real-time* (Kartaputra et al., 2023). Bahkan, pengembangan lebih lanjut dilakukan dengan mengombinasikan MediaPipe dan algoritma Long Short-Term Memory (LSTM) untuk meningkatkan akurasi deteksi bahasa isyarat Indonesia (Utomo et al., 2025). Selain itu, MediaPipe juga digunakan dalam pengembangan sistem interaksi manusia-komputer, seperti implementasi mouse virtual berbasis pengenalan gerakan tangan (Ratnawati & Pangestu, 2025).

Pada praktikum ini digunakan MediaPipe sebagai framework untuk mendeteksi dan melacak gerakan tubuh secara *real-time* melalui kamera. MediaPipe bekerja dengan mengenali titik-titik penting (*landmark*) pada tubuh manusia, seperti bahu, siku, dan lutut, sehingga posisi dan pergerakan sendi dapat dianalisis. Oleh karena itu, penggunaan Python dan MediaPipe dalam praktikum ini menjadi pilihan yang tepat untuk memahami penerapan sistem *computer vision* dalam analisis gerakan tubuh secara langsung dan aplikatif.

B. Tujuan

1. Memahami penggunaan Python sebagai bahasa pemrograman dalam pengembangan sistem *computer vision*.
2. Mempelajari MediaPipe untuk mendeteksi dan melacak titik-titik penting (*landmark*) pada tubuh manusia secara *real-time*.
3. Menganalisis gerakan tubuh menggunakan data *landmark* untuk memahami pose dan pergerakan sendi.
4. Menerapkan konsep computer vision dalam pengembangan aplikasi praktis, seperti deteksi gerakan atau interaksi manusia-komputer.

C. Manfaat

1. Meningkatkan pemahaman mahasiswa tentang bahasa pemrograman Python dan penerapannya dalam computer vision.
2. Melatih keterampilan penggunaan MediaPipe untuk mendeteksi dan menganalisis gerakan tubuh secara real-time.
3. Memberikan pengalaman praktis dalam memproses data *landmark* untuk analisis pose dan pergerakan tubuh.
4. Mendukung pengembangan aplikasi nyata, seperti interaksi manusia-komputer, pengawasan kebugaran, dan sistem pendekripsi gerakan.

D. Kode Program

1. Deteksi Tubuh

```
import cv2
import mediapipe as mp

mpose = mp.solutions.pose
pose = mpose.Pose()
cap = cv2.VideoCapture(0)
```

```

while True:
    success, img = cap.read()
    if not success:
        break

    imgrgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    hasil = pose.process(imgrgb)

    if hasil.pose_landmarks:
        print("terdeteksi")
    else:
        print("tidak terdeteksi")

    cv2.imshow("webcam", img)

    if cv2.waitKey(1) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()

```

2. Body Landmark

```

import cv2
import mediapipe as mp

mpose = mp.solutions.pose
pose = mpose.Pose()
mdraw = mp.solutions.drawing_utils
cap = cv2.VideoCapture(0)

while True:
    success, img = cap.read()
    if not success:
        break

    imgrgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
    hasil = pose.process(imgrgb)
    if hasil.pose_landmarks:
        for id, lm in enumerate(hasil.pose_landmarks.landmark):
            print(id, lm.x, lm.y)

            mdraw.draw_landmarks(img, hasil.pose_landmarks,
                                 mpose.POSE_CONNECTIONS)

    cv2.imshow("webcam", img)
    if cv2.waitKey(10) & 0xFF == ord('q'):
        break

cap.release()
cv2.destroyAllWindows()

```

3. Deteksi Pose Angkat Tangan Kanan/Kiri

```

import cv2
import mediapipe as mp
mpose = mp.solutions.pose
pose = mpose.Pose()
mdraw = mp.solutions.drawing_utils
cap = cv2.VideoCapture(0)

while True:
    success, img = cap.read()
    if not success:
        break

```

```

imgrgb = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
hasil = pose.process(imgrgb)
if hasil.pose_landmarks:
    mdraw.draw_landmarks(img, hasil.pose_landmarks,
mpose.POSE_CONNECTIONS)
    lm = hasil.pose_landmarks.landmark
    pundak_kiri = lm[mpose.PoseLandmark.LEFT_SHOULDER.value]
    tangan_kiri = lm[mpose.PoseLandmark.LEFT_WRIST.value]
    pundak_kanan =
lm[mpose.PoseLandmark.RIGHT_SHOULDER.value]
    tangan_kanan = lm[mpose.PoseLandmark.RIGHT_WRIST.value]

    if tangan_kiri.y < pundak_kiri.y:
        cv2.putText(img, "Tangan Kiri Terangkat",
                    (10, 30),
                    cv2.FONT_HERSHEY_SIMPLEX,
                    0.7, (0, 255, 0), 2)

    if tangan_kanan.y < pundak_kanan.y:
        cv2.putText(img, "Tangan Kanan Terangkat",
                    (10, 60),
                    cv2.FONT_HERSHEY_SIMPLEX,
                    0.7, (255, 0, 0), 2)

cv2.imshow("Deteksi Tangan", img)
if cv2.waitKey(10) & 0xFF == ord('q'):
    break

cap.release()
cv2.destroyAllWindows()

```

E. Analisis

Ketiga program ini menunjukkan langkah-langkah bertahap dalam membuat sistem deteksi pose tubuh menggunakan MediaPipe dan OpenCV.

Program pertama berfokus pada hal dasar, yaitu membaca video dari webcam, memproses gambar dengan MediaPipe Pose, dan mengecek apakah tubuh terdeteksi atau tidak. Pada tahap ini, sistem hanya mengetahui ada atau tidaknya pose tanpa menunjukkan struktur tubuh secara detail. Program ini menjadi dasar untuk memahami bagaimana deteksi pose bekerja secara real-time.

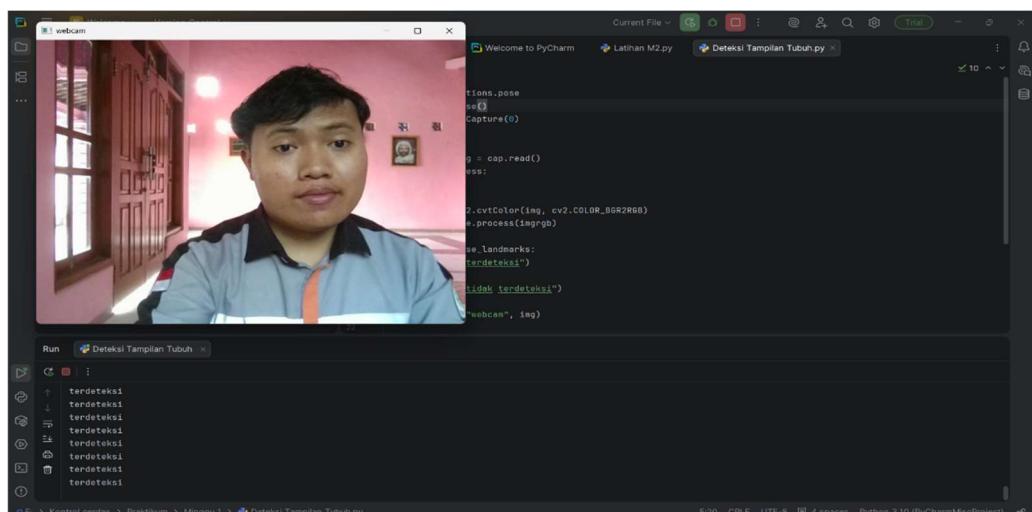
Program kedua adalah pengembangan dari program pertama. Selain mendeteksi pose, program ini menampilkan titik-titik tubuh (landmark) dan garis penghubung antar titik. Koordinat setiap titik juga dicetak ke layar, sehingga posisi setiap bagian tubuh bisa diketahui. Ini membuat sistem lebih canggih karena data landmark bisa digunakan untuk menghitung sudut, menganalisis gerakan, atau mengenali pose tertentu. Tahap ini membantu pengguna memahami posisi 33 titik tubuh pada MediaPipe Pose secara lebih jelas.

Program ketiga lebih aplikatif, karena sistem sudah bisa mengenali pose tertentu, yaitu mendeteksi tangan kiri atau kanan terangkat. Logikanya sederhana:

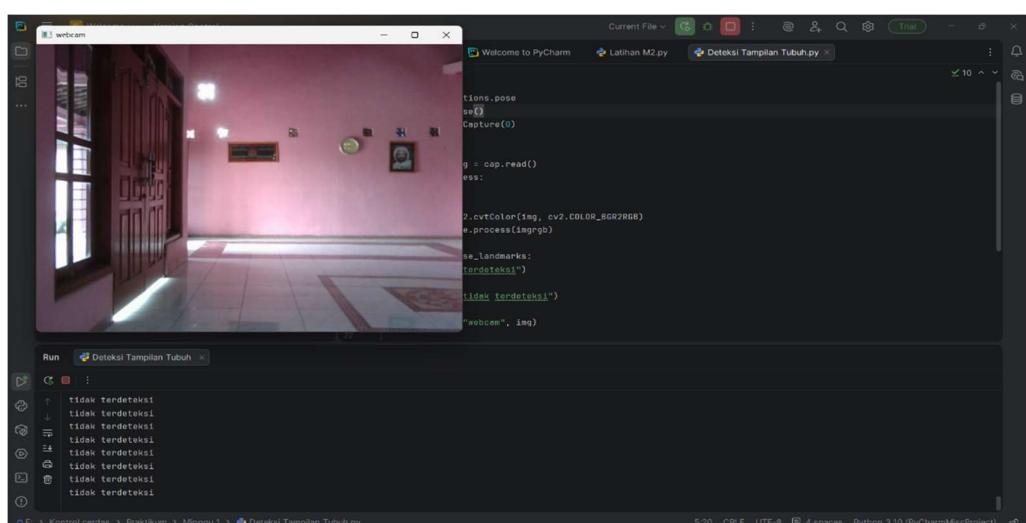
jika pergelangan tangan lebih tinggi daripada bahu, sistem menandai tangan tersebut terangkat. Fitur mirror membuat gerakan terasa alami saat dilihat di layar. Program ini menunjukkan bahwa data landmark tidak hanya bisa dianalisis, tapi juga bisa digunakan untuk memberi umpan balik atau informasi secara langsung.

Secara keseluruhan, ketiga program ini menunjukkan perkembangan dari deteksi sederhana, visualisasi titik tubuh, hingga analisis pose tertentu. Awalnya sistem hanya tahu ada pose, kemudian bisa memahami struktur tubuh, dan akhirnya bisa mengambil keputusan berdasarkan posisi titik-titik tertentu. Praktikum ini tidak hanya mengajarkan cara menggunakan library Python, tapi juga melatih logika dalam mengolah data menjadi informasi yang berguna.

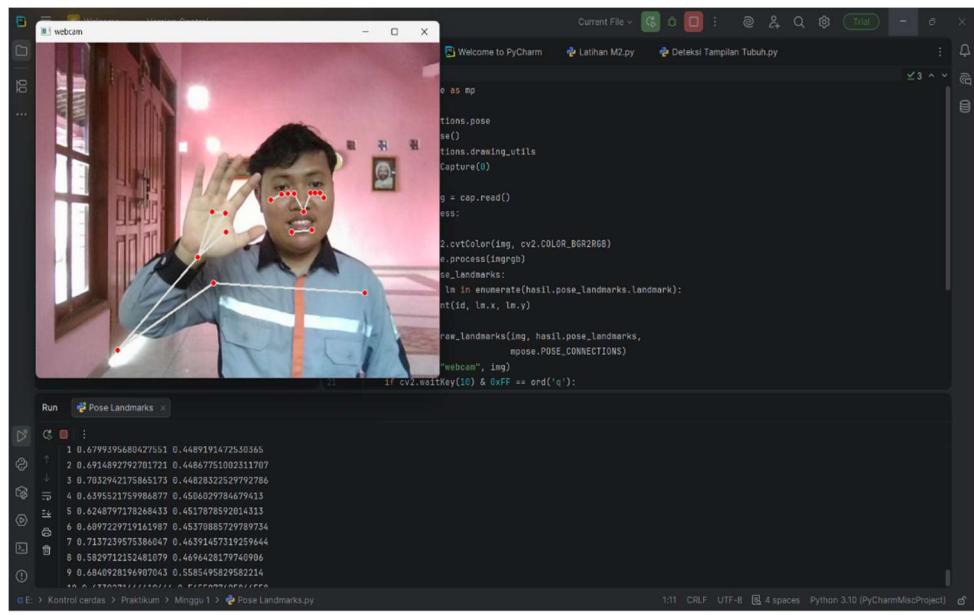
F. Hasil



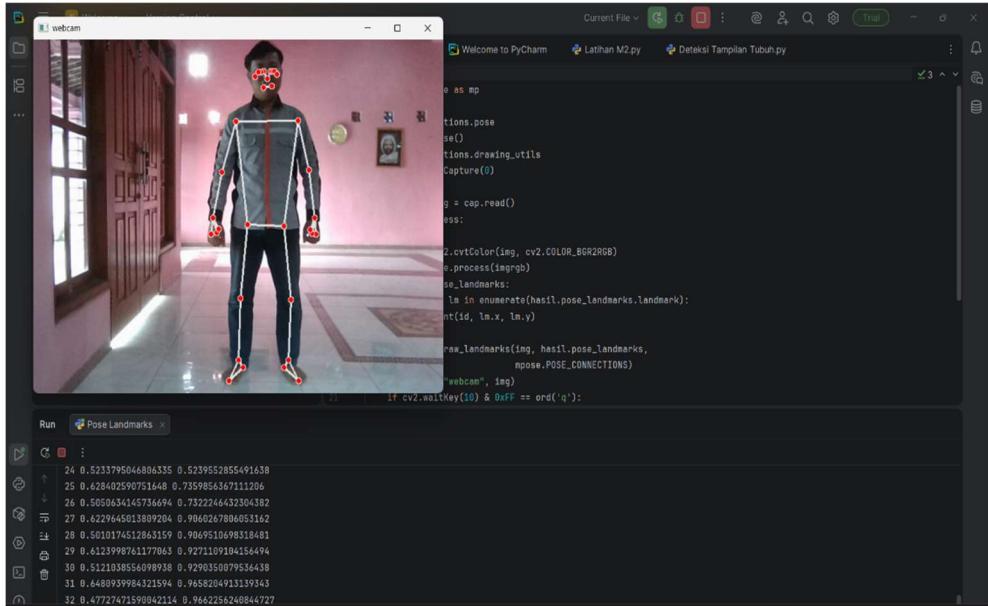
Gambar 1 Tubuh Terdeteksi



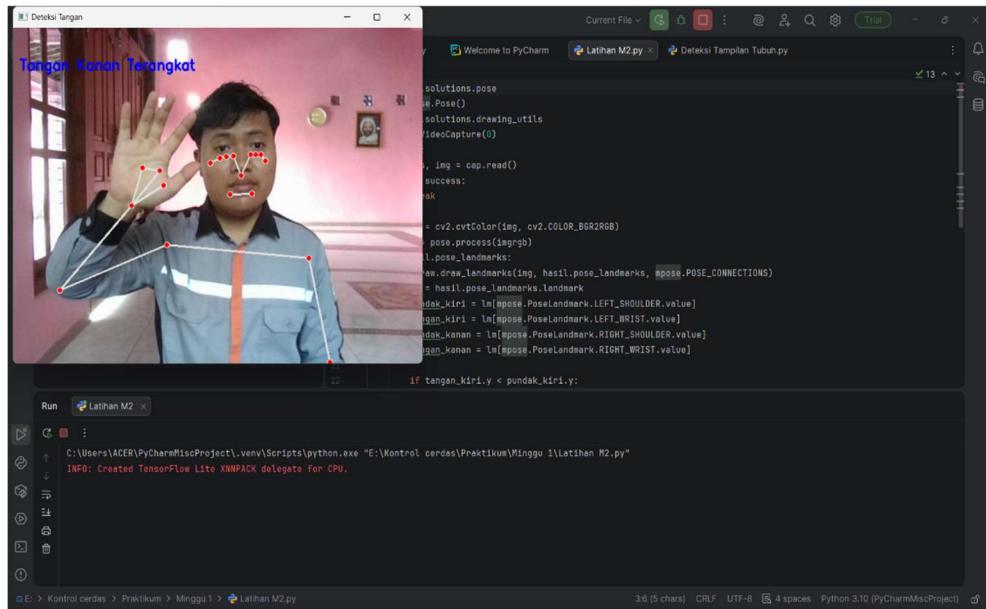
Gambar 2 Tubuh Tidak Terdeteksi



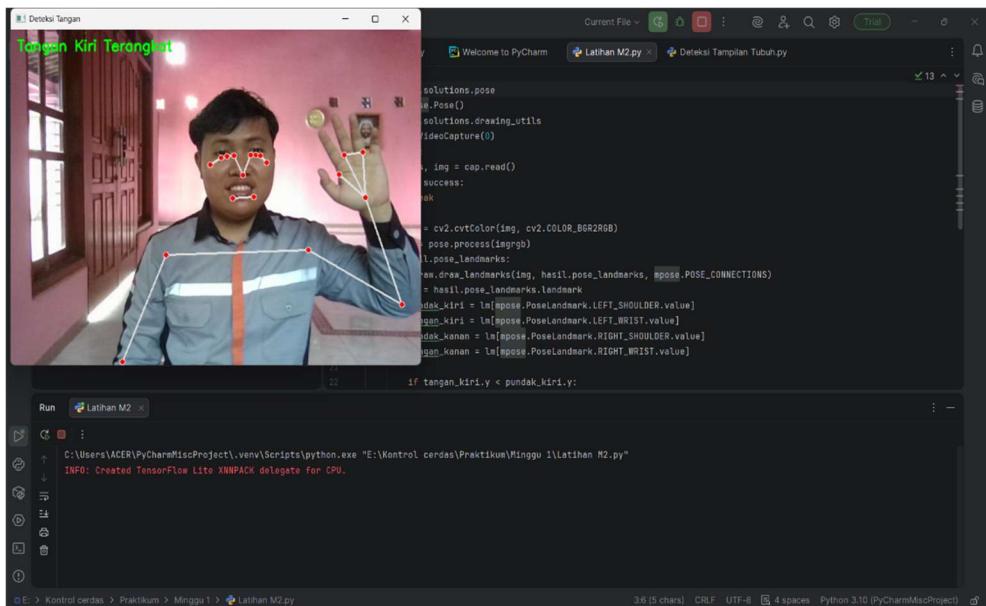
Gambar 3 Landmark pada Wajah dan tangan



Gambar 4 Landmark pada Seluruh Tubuh



Gambar 5 Deteksi Pose Tangan Kanan Terangkat



Gambar 6 Deteksi Pose Tangan Kiri Terangkat

G. Kesimpulan

1. Setelah melakukan praktikum dan menganalisisnya, dapat diperoleh beberapa kesimpulan sebagai berikut :
2. Python dan MediaPipe efektif digunakan untuk deteksi pose tubuh secara real-time, karena sintaks Python sederhana dan MediaPipe menyediakan landmark tubuh lengkap yang mudah diolah.
3. Proses deteksi pose dapat dikembangkan secara bertahap, mulai dari deteksi dasar (ada/tidaknya pose), visualisasi landmark, hingga analisis pose tertentu, seperti tangan terangkat.
4. Visualisasi landmark dan koneksi antar titik tubuh memudahkan pemahaman struktur pose dan memungkinkan analisis lebih lanjut, seperti pengukuran sudut atau klasifikasi gerakan.
5. Deteksi pose sederhana berbasis koordinat landmark memungkinkan sistem mengambil keputusan real-time, misalnya menampilkan indikator tangan terangkat, sehingga aplikasi bisa interaktif.

H. Referensi

Belajarpython.com. (2025). Tutorial Pemrograman Python Bahasa Indonesia Dasar Sampai Advance Belajarpython.com. Belajarpython.com.
<https://belajarpython.com/>.

Dyansyah, K. R. K., Purwantoro, S. D., Ilmi, M., & Wulanningrum, R. (2023). Penggunaan computer vision untuk estimasi pose squat sebagai solusi alternatif latihan kebugaran di gym. Proceedings Seminar Nasional Teknologi & Sains.

Kartaputra, D. P., Gunawan, H., & Lestari, A. E. (2023). Deteksi alfabet BISINDO menggunakan MediaPipe Holistic secara real-time. Jurnal Penelitian dan Pengembangan Teknologi Informasi dan Komunikasi, 12(1), 46–52.

Ratnawati Djuwitaningrum, & Pangestu, D. R. (2025). Implementasi mouse virtual berbasis pengenalan gerakan tangan menggunakan MediaPipe dan OpenCV. Jurnal Ilmu Pengetahuan dan Teknologi.

Suradi, A. A. M., Manguma, T. T. F., Alam, S., & Affifah, A. N. N. (2025). Deteksi dan peringatan jarak wajah otomatis menggunakan MediaPipe dan computer vision untuk kesehatan pengguna komputer. JUKI: Jurnal

Komputer dan Informatika, 7(2), 127–135.

Utomo, W., Suhanda, Y., Ar-Rasyid, H., & Dharmalau, A. (2025). Indonesian language sign detection using MediaPipe with Long Short-Term Memory (LSTM) algorithm. Journal of Informatics and Web Engineering, 4(3), 245–258.