

**Nama** : Annisa Hanif Febrianti  
**NIM** : 234308092  
**Kelas** : TKA 6D  
**Mata Kuliah** : Praktikum Kontrol Cerdas  
**Akun Github** : AnnisaHanif-tech

## **I. Pendahuluan**

Scikit-learn atau yang sering disebut sklearn merupakan salah satu pustaka pada bahasa pemrograman Python yang banyak dimanfaatkan dalam pengembangan aplikasi berbasis *machine learning*. Library ini menyediakan berbagai metode yang dapat digunakan untuk mengolah data serta membangun model pembelajaran mesin, seperti algoritma klasifikasi, regresi, pengelompokan data (*clustering*), hingga teknik reduksi dimensi. Selain proses pelatihan model, sklearn juga dilengkapi dengan berbagai metode evaluasi performa sehingga hasil prediksi dapat dianalisis secara lebih sistematis dan mudah dipahami oleh pengguna.

Penggunaan sklearn relatif sederhana serta dokumentasi resmi yang lengkap dan mudah diikuti, sehingga cocok digunakan oleh pemula maupun peneliti dalam tahap eksperimen. Selain itu, sklearn dapat diintegrasikan dengan library lain dalam ekosistem Python, seperti NumPy, Pandas, maupun OpenCV, sehingga pengguna memiliki fleksibilitas tinggi dalam mengembangkan berbagai jenis proyek berbasis data. Namun, library ini memiliki beberapa keterbatasan, terutama pada pengolahan *deep learning* yang umumnya lebih optimal menggunakan framework khusus seperti TensorFlow atau PyTorch. Sklearn juga kurang efisien ketika menangani dataset berukuran sangat besar serta belum menyediakan dukungan komputasi GPU secara langsung.

Pada penerapan sistem *gesture detection*, misalnya pada pengenalan gerakan tangan atau pose tubuh menggunakan MediaPipe, sklearn berperan dalam tahap pengolahan dan analisis fitur hasil ekstraksi landmark. Titik-titik koordinat yang diperoleh dari hasil deteksi kemudian diolah menjadi data numerik yang dapat digunakan sebagai input model *machine learning*. Selanjutnya, algoritma klasifikasi pada sklearn dapat dilatih untuk mengenali pola gerakan tertentu berdasarkan data tersebut.

## II. Tujuan dan Manfaat

### A. Tujuan

1. Mampu menguji kemampuan sistem deteksi secara *real time* melalui kamera webcam.
2. Mampu mengintegrasikan NumPy, OpenCV, MediaPipe, dan Scikit-learn ke dalam program python,
3. Melatih model klasifikasi dengan memanfaatkan algoritma Random Forest agar dapat mengenali gerakan tangan serta posisi tubuh.

### B. Manfaat


1. Memungkinkan pengambilan serta pengolahan data visual secara langsung dari kamera.
2. Meningkatkan kemampuan praktis dalam analisis data, pelatihan model, serta evaluasi akurasi prediksi, sehingga pengguna dapat memahami performa model secara nyata.
3. Membuka peluang untuk menggabungkan teknologi *computer vision* dengan model machine learning sehingga sistem menjadi lebih cerdas.

## III. Hasil Percobaan


### A. Data Webcam

Name	Date modified	Type	Size
0	26/02/2026 10:38	File folder	
1	26/02/2026 10:38	File folder	

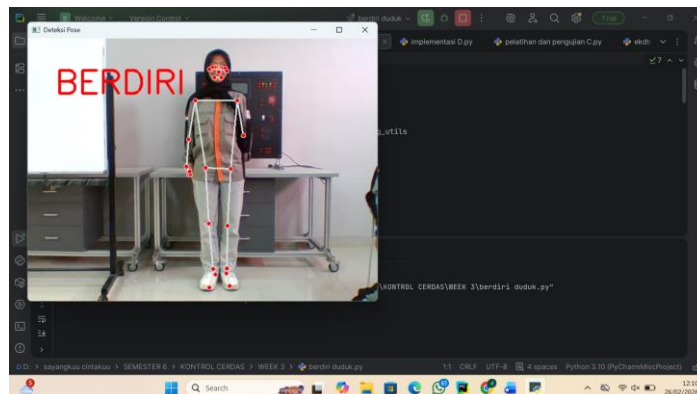
### B. Ekstraksi Datasheet

 data.pickle

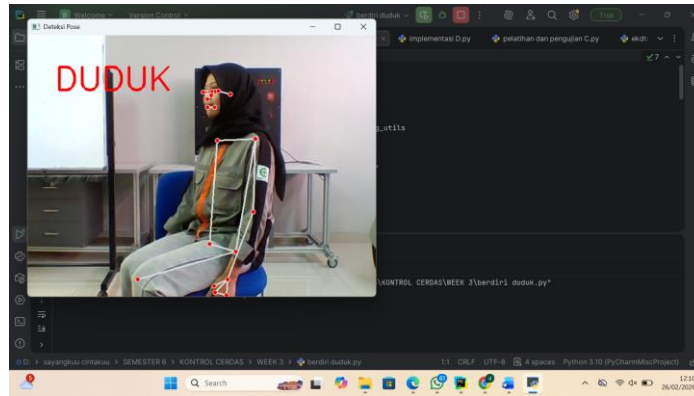
### C. Pelatihan dan Pengujian

 model.p

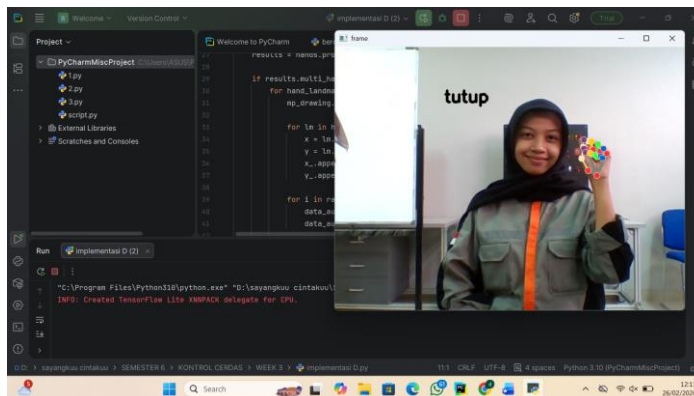
### D. Implementasi



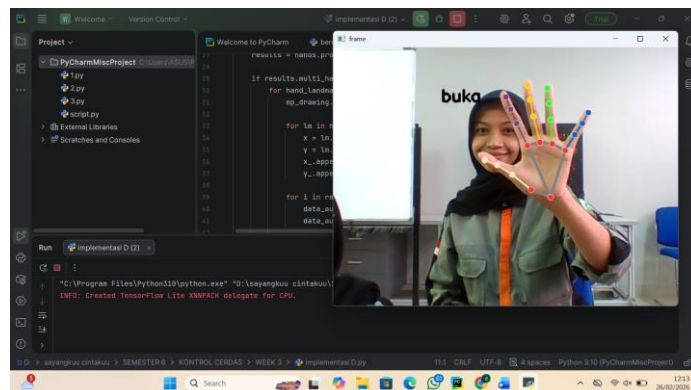
Gambar 1 Deteksi Objek Berdiri



Gambar 1 Deteksi Objek Duduk



Gambar 6 Deteksi Tangan Menutup



Gambar 7 Deteksi Tangan Terbuka

#### IV. Analisis Hasil Percobaan

Berdasarkan keseluruhan hasil pengujian, Hasil implementasi sistem klasifikasi gesture yang menggunakan algoritma Random Forest menunjukkan bahwa seluruh tahapan yang direncanakan berjalan dengan baik, mulai dari pengumpulan data hingga pengujian secara real-time. Proses pengambilan data dilakukan menggunakan webcam yang dipadukan dengan MediaPipe, sehingga titik-titik landmark pada tangan dan tubuh dapat dideteksi dengan stabil dan akurat. Landmark ini kemudian diubah menjadi data numerik (*feature*) yang digunakan sebagai input untuk pelatihan model dengan Scikit-learn. Berdasarkan hasil pelatihan, model mampu membedakan berbagai kategori gerakan, seperti tangan terbuka, tangan tertutup, posisi duduk, dan posisi

berdiri, dengan performa yang cukup baik. Hal ini menandakan bahwa fitur yang diekstraksi dari data visual telah cukup representatif untuk kebutuhan klasifikasi gesture.

Saat diuji secara langsung menggunakan kamera, sistem menunjukkan kemampuan untuk mengenali gerakan dengan respons yang cepat, selama kondisi pencahayaan memadai dan posisi objek relatif stabil di depan kamera. Integrasi antara pustaka NumPy, OpenCV, MediaPipe, dan Scikit-learn berjalan lancar tanpa hambatan berarti, sehingga prediksi dapat dilakukan secara real-time dan responsif. Meski demikian, beberapa faktor masih memengaruhi tingkat akurasi sistem, seperti sudut pengambilan gambar yang berbeda, jarak pengguna terhadap kamera, serta keterbatasan jumlah data latih yang tersedia.

Secara keseluruhan, program yang dikembangkan berhasil mencapai tujuan praktikum dengan baik. Namun, masih ada ruang untuk pengembangan lebih lanjut, terutama dalam memperluas variasi dataset, meningkatkan jumlah data latih, dan mengoptimalkan model agar klasifikasi gesture menjadi lebih konsisten dan akurat. Peningkatan ini juga dapat mencakup penyesuaian algoritma atau penggunaan teknik tambahan untuk menangani kondisi pencahayaan yang beragam dan posisi pengguna yang tidak selalu ideal. Dengan demikian, sistem ini memiliki potensi untuk diterapkan dalam berbagai aplikasi nyata, seperti interaksi berbasis gerakan tangan atau kontrol perangkat secara intuitif.

## **V. Referensi**

Belajar Python – Situs Open Source Tutorial Pemrograman Python Bahasa Indonesia –  
<https://belajarpython.com>