

Laporan Praktikum Kontrol Cerdas

Nama : Alam Syah Putra Aulia
NIM : 224308003
Kelas : TKA 6A
Akun Github (Tautan) : <https://github.com/Alamsyah003>
Student Lab Assistant : Yulia Brillianty

1. Judul Percobaan

Deteksi Warna dengan Machine Learning (ML)

2. Tujuan Percobaan

Tujuan dari praktikum ini adalah sebagai berikut

1. Mahasiswa dapat memahami dasar-dasar Machine Learning dalam sistem kendali.
2. Mahasiswa dapat mengimplementasikan model ML sederhana untuk klasifikasi objek.
3. Mahasiswa dapat menggunakan Scikit-learn untuk membuat model ML dasar.
4. Mahasiswa dapat mengintegrasikan model ML dengan Computer Vision untuk deteksi objek.
5. Mahasiswa dapat mengelola dataset dan melakukan pelatihan model sederhana.

3. Landasan Teori

Machine Learning merupakan salah satu cabang dari ilmu Kecerdasan Buatan, khususnya yang mempelajari tentang bagaimana komputer mampu belajar dari data untuk meningkatkan kecerdasannya (Wahyono, 2018)

Metode K-Nearest Neighbor (KNN) merupakan metode klasifikasi terhadap obyek berdasarkan data pembelajaran dengan jarak yang terdekat. Metode KNN diterapkan untuk mengatasi permasalahan identifikasi yang diukur secara kualitatif maupun kuantitatif. Sebelum mencari jarak data ke tetangga adalah menentukan nilai K tetangga (neighbor). Lalu, untuk mendefinisikan jarak antara dua titik pada data training dan titik pada data testing, yaitu menggunakan rumus Euclidean Distance. Hasil dari perhitungan jarak, kemudian diurutkan dari

data dengan nilai terkecil hingga terbesar sesuai dengan nilai k yang telah ditentukan sebelumnya (Putri & Sodik, 2019).

Decision tree Merupakan algoritma yang digunakan untuk proses klasifikasi maupun prediksi melalui tahap pembentukan pohon keputusan. Pohon keputusan digunakan untuk prosedur penalaran untuk mendapatkan jawaban dari sebuah masalah (Syarief et al., 2017). Sedangkan menurut (Triscowati et al., 2019) Decision tree merupakan metode klasifikasi dan prediksi yang sangat kuat dan terkenal. Metode decision tree mengubah fakta yang sangat besar menjadi pohon keputusan yang mempresentasikan aturan.

4. Analisis dan Diskusi

Analisis :

- **Bagaimana performa model dalam mendeteksi warna?**

Performa model dalam mendeteksi warna bergantung pada beberapa faktor seperti jumlah data terlatih, fitur yang digunakan (RGB, HSV, atau fitur lainnya), serta parameter model yang dipilih. Model Decision Tree digunakan karena model ini memiliki beberapa keunggulan seperti cepat dalam pelatihan dan prediksi, mudah dipahami dan diinterpretasikan, dan tidak terlalu sensitif terhadap skala fitur sehingga cocok untuk dataset dengan pola sederhana dan jumlah data terbatas.

- **Bagaimana perbedaan akurasi jika jumlah dataset ditambah?**

Penambahan jumlah dataset akan memengaruhi akurasi karena model mempelajari lebih banyak data. Secara umum, semakin banyak dataset yang representatif, maka model Machine Learning dapat mengenali lebih banyak variasi warna yang mungkin muncul, menghindari overfitting pada dataset kecil yang tidak mencerminkan kondisi nyata, menyesuaikan lebih baik terhadap distribusi warna di dunia nyata, serta meningkatkan generalisasi model, sehingga performa di data uji lebih stabil. Namun, efeknya tergantung pada model yang digunakan dan jumlah dataset yang ditambahkan.

- **Bagaimana cara meningkatkan kinerja model klasifikasi?**

Peningkatan kinerja model klasifikasi dapat dilakukan dengan cara menambahkan dataset yang tujuannya untuk menambah referensi sehingga hasilnya memiliki akurasi yang lebih tinggi. Selain itu, juga dapat dilakukan penambahan fitur yang sesuai untuk meningkatkan kinerja model.

Diskusi :

- **Apa keuntungan Machine Learning dibandingkan metode berbasis aturan (rule-based)?**

Machine Learning dapat menangani data yang kompleks dan tidak terstruktur karena rule-based hanya bekerja dengan data yang didefinisikan secara eksplisit. Machine Learning juga lebih adaptif dan fleksibel karena rule-based harus diperbarui manual setiap ada perubahan. Machine Learning meningkatkan akurasi seiring waktu sedangkan rule-based memiliki batasan akurasi karena hanya mengandalkan aturan yang sudah ditentukan

- **Bagaimana ML dapat diintegrasikan lebih lanjut dalam sistem kendali?**

Machine Learning bisa digunakan dalam sistem kontrol untuk membuatnya lebih pintar dan lebih mudah beradaptasi. Misalnya, model prediktif berbasis ML dapat menganalisis bagaimana suatu sistem bekerja dan secara otomatis menyesuaikan pengaturan kontrol agar lebih optimal. Ini sangat berguna dalam bidang seperti robotika dan otomasi industri. Selain itu, sistem kontrol bisa belajar dari pengalaman langsung, sehingga dapat menemukan cara terbaik untuk menangani situasi yang rumit, seperti mengendalikan mobil tanpa pengemudi.

- **Apa saja tantangan dalam penerapan ML dalam sistem real-time?**

Menggunakan Machine Learning dalam sistem nyata memiliki beberapa tantangan. Salah satu masalah utama adalah keterbatasan daya komputasi, terutama untuk model yang kompleks seperti deep learning. Model seperti ini membutuhkan banyak sumber daya, sehingga sulit untuk digunakan dalam aplikasi yang membutuhkan respons cepat. Selain itu, data yang dikumpulkan secara real-time bisa terbatas atau memiliki keterlambatan, yang dapat mengurangi akurasi dan kecepatan sistem. Keamanan juga menjadi tantangan, karena model ML rentan terhadap serangan yang bisa mengganggu kinerjanya, terutama dalam aplikasi penting seperti mobil tanpa pengemudi atau proses industri.

5. Assignment

Assignment dimodifikasi dengan menggunakan model Decision Tree. Proses pengerjaan diawali dengan studi literatur terkait Machine Learning, K-Nearest Neighbors (KNN), dan Decision Tree (DT). Selanjutnya, repository pada GitHub dibuat dan di-clone ke laptop. Dilanjutkan dengan mencari dataset pada Kaggle dan membuat program dengan model Decision Tree (DT). Percobaan diakhiri dengan Commit ke akun GitHub masing-masing.

6. Data dan Output Hasil Pengamatan

Sajikan data dan hasil yang diperoleh selama percobaan. Gunakan tabel untuk menyajikan data jika diperlukan.

No	Variabel	Hasil Pengamatan
1	Pengujian menggunakan model K-Nearest Neighbor (KNN) Machine Learning	
2	Pengujian menggunakan model Data Tree (DT) Machine Learning dengan Dataset lain	

7. Kesimpulan

Dari beberapa percobaan yang telah dilakukan, dapat disimpulkan bahwa program yang dibuat dapat digunakan untuk mendeteksi warna secara real-time meskipun hasilnya masih dipengaruhi beberapa faktor seperti pencahayaan, kualitas kamera, dan dataset. Akurasi dari program ini juga dapat ditingkatkan dengan beberapa cara seperti menambah dataset yang lebih variatif dan penambahan fitur yang sesuai serta kamera eksternal yang memiliki kualitas gambar lebih baik daripada kamera default laptop.

8. Saran

Untuk percobaan selanjutnya dapat dilakukan penggunaan model yang lebih kompleks seperti Convolutional Neural Networks (CNN) serta penambahan dataset yang lebih beragam sehingga data yang dihasilkan lebih variatif juga informatif.

9. Daftar Pustaka

- Putri, A. Y. P., & Sodik, A. (2019). *Identifikasi Penyakit Tanaman Kopi Arabika dengan Metode K-Nearest Neighbor (K-NN)*.
- Syarief, M., Mukminin, A., Prastiti, N., & Setiawan, W. (2017). *PENERAPAN METODE NAÏVE BAYES CLASSIFIER UNTUK DETEKSI PENYAKIT PADA TANAMAN JAGUNG*. 3.
- Triscowati, D. W., Sartono, B., Kurnia, A., & Domiri, D. D. (2019). *Klasifikasi Fase Tanam Padi Menggunakan Supervised Random Forest Pada Data Multitemporal Citra Landsat-8*.
- Wahyono, T. (2018). *Python for Machine Learning (Dasar-dasar Pemrograman Python untuk Machine Learning dan Kecerdasan Buatan)*.