

Laporan Praktikum Kontrol Cerdas

Nama : Aureyza Pandu Qinara
NIM : 224308078
Kelas : TKA - 7D
Akun Github (Tautan) : <https://github.com/aureyzapandu>
Student Lab Assistant :-

Judul Percobaan

Week 1: Machine Learning for Control Systems

1. Tujuan Percobaan

Tujuan dari praktikum “*Machine Learning for Control Systems*”, mahasiswa diharapkan mampu:

- a) Memahami dasar-dasar Machine Learning pada sistem kendali.
- b) Menggunakan KNN untuk klasifikasi warna objek berdasarkan data RGB.
- c) Mengintegrasikan model ML dengan *Computer Vision* (OpenCV) untuk deteksi objek secara real-time.
- d) Mengevaluasi performa model serta mendiskusikan cara meningkatkan akurasi.

2. Landasan Teori

- *Machine Learning*

Machine Learning adalah bidang ilmu komputer yang memungkinkan sistem “belajar” dari data tanpa diprogram secara eksplisit untuk setiap kondisi. Sebagai contoh, sistem diberikan data *input-output* yang sudah diberi label, kemudian belajar memprediksi *output* untuk data baru (Mitchell, 1997).

- *Supervised Learning*

Supervised learning adalah pendekatan dalam *machine learning* di mana model dilatih menggunakan data yang sudah berlabel (memiliki *input* dan *output* yang jelas). Algoritma berusaha menemukan fungsi pemetaan dari *input* ke *output* sehingga dapat memprediksi label untuk data baru yang belum pernah dilihat. *Unsupervised learning* adalah pendekatan *machine learning* dimana model dilatih menggunakan data tanpa label. Tujuannya adalah

menemukan pola, struktur, atau hubungan tersembunyi di dalam data (Nurhalizah et al., 2024).

- **Reinforcement Learning (RL)**

Reinforcement Learning (RL) merupakan salah satu cabang dari *machine learning* yang berfokus pada proses pengambilan keputusan melalui interaksi langsung dengan lingkungan. Berbeda dengan *supervised learning* yang menggunakan data berlabel, maupun *unsupervised learning* yang mencari pola dari data tanpa label, RL bekerja dengan mekanisme *trial-and-error*. Dalam pendekatan ini, sebuah agen belajar menentukan tindakan (*action*) terbaik dengan cara mencoba berbagai kemungkinan, menerima umpan balik dalam bentuk *reward* atau *punishment*, kemudian menyesuaikan strategi untuk memaksimalkan total *reward* di masa depan (Andreanus and Kurniawan, 2018).

- **K-Nearest Neighbor (KNN)**

Algoritma *K-Nearest Neighbor* (KNN) merupakan salah satu metode klasifikasi yang sederhana namun cukup efektif dalam penerapan *machine learning*. Prinsip utama dari KNN adalah mencari kedekatan data baru terhadap data latih yang sudah ada. KNN digolongkan sebagai metode *lazy learning* karena tidak memerlukan proses pelatihan model yang kompleks, melainkan langsung menggunakan seluruh data latih sebagai acuan. Oleh sebab itu, performa KNN sangat bergantung pada kualitas dan jumlah data latih yang digunakan. Selain itu, pemilihan nilai *k* juga berpengaruh signifikan terhadap hasil klasifikasi: nilai *k* yang terlalu kecil dapat menyebabkan model *overfitting*, sedangkan nilai *k* yang terlalu besar berisiko menghasilkan *underfitting* (Farokhah, 2020).

3. Analisis dan Diskusi

a. Performa Model dalam Mendeteksi Warna

Program ini mampu mendeteksi warna dari piksel tengah pada *frame webcam* dengan menggunakan algoritma KNN. Hasil deteksi cukup baik pada warna-warna dasar yang tersedia dalam dataset. Namun, pada warna yang memiliki kemiripan intensitas atau gradasi, model masih berpotensi

melakukan kesalahan prediksi. Hal ini menunjukkan bahwa akurasi sangat bergantung pada representasi data dalam dataset.

b. Perbedaan Akurasi Jika Jumlah Dataset Ditambah

Penambahan jumlah dataset berpengaruh positif terhadap akurasi model. Dengan semakin banyak data yang dimasukkan, model akan lebih mampu mengenali pola RGB dan membedakan warna dengan tingkat kesalahan yang lebih kecil. Dataset yang bervariasi dan seimbang antara tiap kelas warna juga dapat membantu meningkatkan keandalan model dalam situasi nyata.

c. Cara Meningkatkan Kinerja Model Klasifikasi

Kinerja model dapat ditingkatkan dengan beberapa strategi. Pertama, memperbanyak variasi dan jumlah data agar model lebih representatif. Kedua, mencoba konversi ke ruang warna lain seperti HSV atau LAB yang lebih mendekati persepsi visual manusia. Ketiga, melakukan *hyperparameter tuning* pada algoritma KNN, misalnya dengan mengubah nilai '`n_neighbors`', atau bahkan membandingkan hasilnya dengan algoritma lain seperti *Decision Tree*, *Random Forest*, atau SVM. Selain itu, menjaga kualitas dataset agar tidak terdapat label yang salah atau data duplikat juga penting untuk meminimalkan kesalahan prediksi.

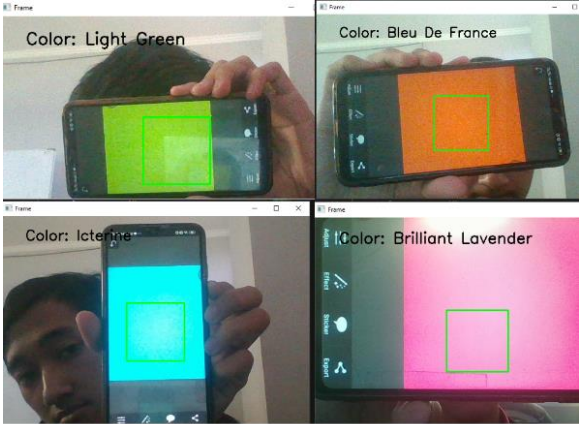
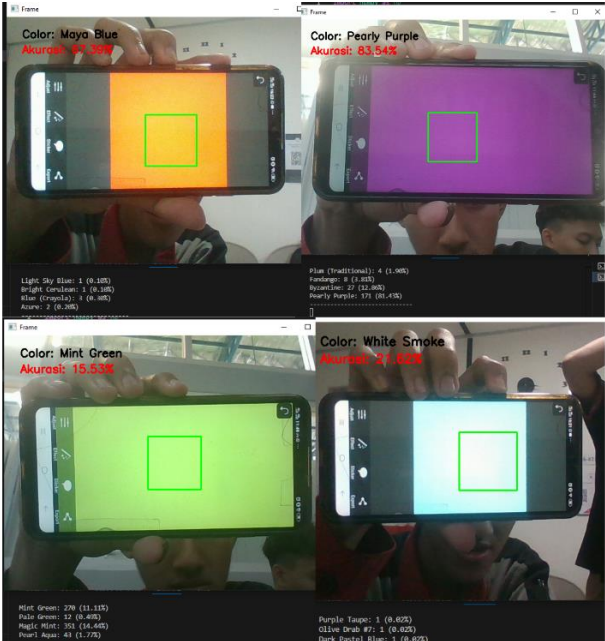
4. Assignment

1. Melakukan studi literatur mengenai konsep dasar yang mendukung percobaan, antara lain OpenCV sebagai pustaka pengolahan citra, ruang warna HSV untuk segmentasi warna yang lebih akurat, teknik *masking* dan segmentasi untuk memisahkan objek berwarna, deteksi kontur serta pengenalan bentuk untuk identifikasi persegi, serta penggunaan *bounding box* dan labeling untuk menandai serta memberi identitas pada objek yang terdeteksi.
2. Membuat akun GitHub sebagai sarana penyimpanan kode program berbasis *version control system*.
3. Menginstal perangkat lunak pendukung, yaitu VSCode, Python, dan Git.
4. Membuat repository di GitHub kemudian melakukan *clone repository* ke laptop.

5. Membuat kode program deteksi objek persegi berwarna merah, biru, hijau, dan kuning menggunakan OpenCV.
6. Menjalankan program untuk menampilkan hasil deteksi pada tiga jendela yaitu *Frame*, *Mask*, dan *Result*.
7. Menghentikan program dengan menekan tombol q pada keyboard.
8. Melakukan *commit* dan *push* kode program ke *repository* GitHub sebagai dokumentasi akhir.

5. 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 model <i>Machine Learning K-Nearest Neighbors</i> (KNN)	
2	Pengujian model <i>Machine learning Support Vector Machine</i> (SVM)	

6. Kesimpulan

- a) Mahasiswa berhasil memahami dasar-dasar *machine learning* dalam konteks sistem kendali, khususnya pada penerapan klasifikasi warna menggunakan data RGB.
- b) Algoritma KNN mampu mengklasifikasikan warna dasar dengan baik, meskipun masih berpotensi keliru pada warna dengan gradasi atau kemiripan intensitas.
- c) Integrasi dengan OpenCV memungkinkan deteksi warna secara *real-time* melalui kamera, sehingga mendemonstrasikan hubungan antara *machine learning* dan *computer vision*.
- d) Penambahan jumlah serta variasi dataset terbukti meningkatkan akurasi model, sehingga data yang representatif sangat penting dalam membangun sistem klasifikasi.
- e) Hasil percobaan menunjukkan bahwa kinerja model dapat ditingkatkan dengan strategi seperti pemilihan ruang warna yang lebih sesuai (misalnya HSV atau LAB), pengaturan parameter (*hyperparameter tuning*), maupun membandingkan performa dengan algoritma lain seperti *Decision Tree*, *Random Forest*, atau SVM.

7. Saran

- a) Dataset yang digunakan sebaiknya diperluas dengan lebih banyak variasi warna dan kondisi pencahayaan agar model lebih robust di situasi nyata.
- b) Perlu dilakukan uji coba menggunakan ruang warna selain RGB, seperti HSV atau LAB, yang lebih mendekati persepsi manusia terhadap warna.
- c) Disarankan melakukan *hyperparameter tuning* secara sistematis, misalnya menggunakan *Grid Search* atau *Cross Validation*, untuk mendapatkan konfigurasi model yang optimal.
- d) Perbandingan antara KNN, SVM, dan algoritma lain sebaiknya dilengkapi dengan metrik evaluasi tambahan seperti *precision*, *recall*, dan *F1-score*, agar hasil analisis lebih komprehensif.
- e) Implementasi *real-time* dapat diperluas untuk mendeteksi lebih dari satu objek sekaligus, serta mengintegrasikan deteksi bentuk (*shape detection*) agar aplikasi lebih praktis di dunia nyata.

8. Daftar Pustaka

- Andreanus, J., Kurniawan, A., 2018. Sejarah, Teori Dasar dan Penerapan Reinforcement Learning: Sebuah Tinjauan Pustaka. *J. Telematika* 12, 113–118. <https://doi.org/10.61769/telematika.v12i2.193>
- Farokhah, L., 2020. Implementasi K-Nearest Neighbor untuk Klasifikasi Bunga Dengan Ekstraksi Fitur Warna RGB. *J. Teknol. Inf. Dan Ilmu Komput.* 7, 1129–1136. <https://doi.org/10.25126/jtiik.2020722608>
- Mitchell, T.M., 1997. *Machine Learning*, McGraw-Hill series in computer science. McGraw-Hill, New York.
- Nurhalizah, R.S., Ardianto, R., Purwono, P., 2024. Analisis Supervised dan Unsupervised Learning pada Machine Learning: Systematic Literature Review. *J. Ilmu Komput. Dan Inform.* 4, 61–72. <https://doi.org/10.54082/jiki.168>