

# Laporan Praktikum Kontrol Cerdas

Nama : Rhehan Adi Prakoso  
NIM : 224308094  
Kelas : TKA-7D  
Akun Github (Tautan) : <https://github.com/RhehanAdi>  
Student Lab Assistant : -

## 1. Judul Percobaan

Week 2: Machine Learning for Control Systems.

## 2. Tujuan Percobaan

Tujuan dari dilakukannya praktikum “Machine Learning for Control Systems” adalah:

- Memahami dasar-dasar *Machine Learning* (ML) dalam sistem kendali.
- Mengimplementasikan model ML sederhana untuk klasifikasi objek.
- Menggunakan *Scikit-learn* untuk membuat model ML dasar.
- Mengintegrasikan model ML dengan *Computer Vision* untuk deteksi objek.
- Mengelola dataset dan melakukan pelatihan model sederhana.

## 3. Landasan Teori

A. ***Machine Learning* (ML)** adalah cabang dari *Artificial Intelligence* (AI) yang berfokus pada pengembangan sistem atau algoritma yang mampu belajar dari data untuk membuat prediksi maupun pengambilan keputusan tanpa harus diprogram secara eksplisit. Konsep dasarnya adalah bagaimana komputer dapat mengenali pola dari sejumlah besar data, kemudian menggunakan pola tersebut untuk menyelesaikan masalah baru (Shalev-Shwartz and Ben-David, 2014). Proses pembelajaran *machine learning* melibatkan tiga tahapan utama, yaitu pelatihan (*training*) menggunakan dataset, pengujian (*testing*) untuk mengevaluasi kemampuan model, serta penerapan (*deployment*) ketika model sudah dianggap cukup akurat (Heymann et al., 2022). *Machine learning* sendiri terbagi ke dalam beberapa jenis, di antaranya *supervised learning* yang memerlukan data berlabel, *unsupervised learning* yang digunakan untuk menemukan pola tersembunyi pada data tidak berlabel, serta *reinforcement learning* yang berfokus pada proses pembelajaran melalui interaksi dengan

lingkungan (Bintoro et al., 2024).

- B. ***Supervised Learning*** adalah metode pendekatan *machine learning* yang melakukan pembelajaran dengan menggunakan data yang sudah memiliki label atau jawaban yang benar. Setiap data masukan (*input*) dipasangkan dengan keluaran (*output*) yang sesuai, sehingga algoritma dapat belajar dari hubungan antara keduanya. Selama tahap *training*, model mencoba menemukan pola dari data berlabel tersebut agar mampu memprediksi keluaran yang tepat untuk data baru yang belum pernah dilihat. Metode *supervised learning* menghasilkan output dalam bentuk klasifikasi dan regresi berupa nilai numeriknya. *Supervised learning* sangat akurat dalam mengidentifikasi, dikarenakan memiliki label yang jelas pada setiap datasetnya namun dataset yang dibutuhkan juga banyak dan harus memiliki label yang benar (Dinata and Hasdyna, 2020).
- C. ***Unsupervised learning*** adalah metode pendekatan *machine learning* yang bekerja dengan data yang tidak memiliki label. Pada *unsupervised learning* sistem harus menemukan pola, struktur, atau hubungan tersembunyi dalam data secara mandiri. Sehingga model akan melakukan pengelompokan data yang mirip atau menemukan keteraturan yang sebelumnya tidak diketahui. *Unsupervised learning* sering diaplikasikan pada beberapa sistem seperti sistem rekomendasi produk, segmentasi pasar, deteksi anomali (seperti aktivitas keuangan mencurigakan), hingga analisis pola dalam data biologi. Kelemahan yang dimiliki *unsupervised learning* adalah sulit untuk mengevaluasi hasil karena tidak ada label pembandingan sehingga model memungkinkan untuk menemukan wawasan baru dari data mentah yang sebelumnya belum terstruktur (Dinata and Hasdyna, 2020).
- D. ***Reinforcement learning*** adalah metode pendekatan *machine learning* di mana proses pembelajaran dilakukan melalui interaksi antara agen (program/algoritma) dengan lingkungan. Pada *reinforcement learning* agen tidak diberikan label data, melainkan belajar dari pengalaman dengan cara mencoba berbagai tindakan (*actions*) dan menerima **umpan balik** berupa *reward* (hadiah) atau *punishment* (hukuman) sehingga model mampu memaksimalkan total *reward* yang diperoleh dalam jangka panjang dengan

menyusun strategi atau terbaik. Contoh penerapannya dapat ditemukan pada robot yang belajar berjalan, sistem permainan (game AI) yang mampu mengalahkan pemain manusia, hingga sistem navigasi kendaraan otonom. Proses belajar ini mirip dengan cara manusia atau hewan belajar melalui *trial and error*—mencoba, gagal, lalu memperbaiki tindakannya. Keunggulan *reinforcement learning* adalah kemampuannya beradaptasi pada lingkungan yang dinamis, namun kelemahannya adalah membutuhkan banyak percobaan dan waktu komputasi yang besar agar model bisa berjalan efektif (Dinata and Hasdina, 2020).

- E. **RGB** adalah singkatan dari Red, Green, Blue, yaitu model warna yang paling umum digunakan dalam berbagai perangkat digital seperti monitor, kamera, hingga sensor gambar. Konsep dasarnya adalah warna dapat dibentuk dari kombinasi tiga warna primer cahaya: merah (R), hijau (G), dan biru (B). Setiap komponen biasanya direpresentasikan dengan nilai intensitas tertentu, misalnya dalam rentang 0–255 pada sistem digital 8-bit. Model RGB bersifat *additive color model*. Semakin tinggi intensitas cahaya yang digabungkan, maka warnanya semakin terang (Annas, 2019).

#### 4. Analisis dan Diskusi

##### A. Analisis Hasil

Berdasarkan hasil praktikum yang dilakukan, model *machine learning* yang digunakan mampu mengidentifikasi warna utama dengan cukup akurat, terutama warna yang memiliki perbedaan kontras yang tinggi seperti merah, hijau dan biru. Namun model mengalami kesulitan dalam mengidentifikasi warna yang memiliki tingkat kemiripan yang tinggi seperti fuchsia dan magenta atau warna dengan saturasi rendah yang mendekati abu-abu sehingga tingkat akurasi pada model menurun. Selain itu, faktor resolusi yang digunakan pada webcam mempengaruhi model dalam mengidentifikasi warna karena resolusi menentukan seberapa detail informasi warna yang ditangkap. Semakin tinggi resolusi, semakin banyak piksel yang tersedia untuk merepresentasikan objek, sehingga perbedaan warna yang halus atau gradasi dapat terdeteksi dengan lebih akurat. Sebaliknya, resolusi rendah dapat menyebabkan informasi warna menjadi kurang jelas, piksel tercampur, atau detail warna hilang, yang berpotensi menurunkan akurasi model. Untuk

meningkatkan nilai akurasi dari model bisa dilakukan penambahan jumlah dataset. Jika jumlah dataset ditambah, akurasi model cenderung meningkat. Penambahan dataset memungkinkan model untuk belajar dari lebih banyak variasi warna. Selain itu, untuk meningkatkan akurasi model dilakukan teknik augmentasi data, seperti mengubah kecerahan, kontras, atau sedikit rotasi, untuk memperluas kemampuan model tanpa harus menambah dataset manual. Penggunaan representasi warna yang lebih efektif seperti HSV atau LAB, mengoptimalkan parameter model, serta mengurangi noise pada data input sehingga tingkat akurat yang dihasilkan model meningkat dan stabil dalam berbagai kondisi.

Selanjutnya dilakukan percobaan pada ketiga model *machine learning* yakni K-Nearest Neighbors (KNN), Decision Tree (DT), dan Support Vector Machine (SVM). Setiap model menunjukkan performa yang berbeda dalam mendeteksi warna. KNN mampu memberikan prediksi yang cukup akurat untuk warna-warna dengan perbedaan kontras tinggi karena prinsip kerjanya yang mengklasifikasikan data baru berdasarkan kedekatan dengan data tetangga terdekat. Namun, KNN cenderung lebih lambat pada dataset yang besar karena harus menghitung jarak ke semua data saat prediksi. Decision Tree menunjukkan performa yang baik dan lebih cepat dibanding KNN, serta mudah diinterpretasikan karena model membagi data berdasarkan aturan berbentuk pohon. Namun, DT cenderung *overfitting* (mampu memberikan prediksi yang sangat akurat pada data *training*, tetapi gagal memprediksi data baru dengan baik) pada dataset yang kompleks atau berisik, sehingga akurasinya bisa menurun pada data baru. SVM memberikan akurasi yang tinggi terutama pada dataset yang memiliki batas pemisah jelas antar warna, karena SVM mencari hyperplane terbaik untuk memisahkan kelas. Namun, SVM membutuhkan penyesuaian parameter dan komputasi yang lebih tinggi untuk dataset besar.

## B. Diskusi

- 1) Apa keuntungan *Machine Learning* dibandingkan metode berbasis aturan (*rule-based*)?
  - \* Mampu belajar dari data secara otomatis sehingga menghasilkan prediksi yang lebih akurat tanpa perlu membuat aturan manual satu per satu.
- 2) Bagaimana ML dapat diintegrasikan lebih lanjut dalam sistem kendali?

- \* Diintegrasikan dengan memprediksi perilaku dari sistem, mengoptimalkan keputusan dan menyesuaikan kontrol secara otomatis berdasarkan data sehingga kendali menjadi lebih adaptif dan efisien

3) Apa saja tantangan dalam penerapan ML dalam sistem real-time?


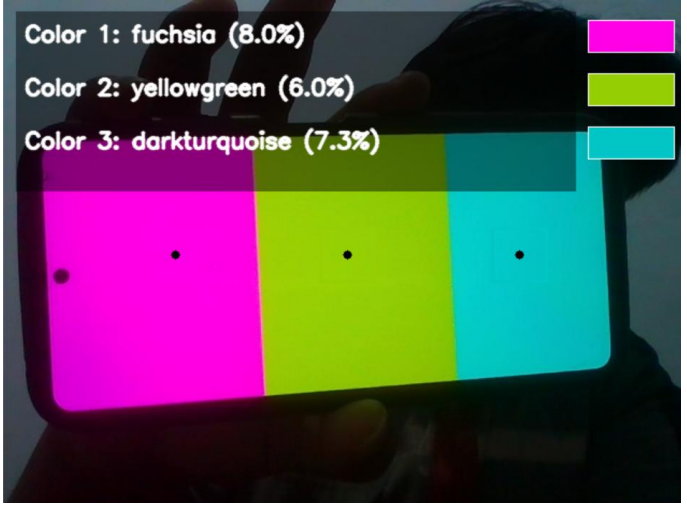
- \* Latensi (waktu respon) yang tinggi, kebutuhan komputasi yang kompleks, keterbatasan data, risiko overfitting, dan kesulitan adaptasi pada perubahan yang cepat

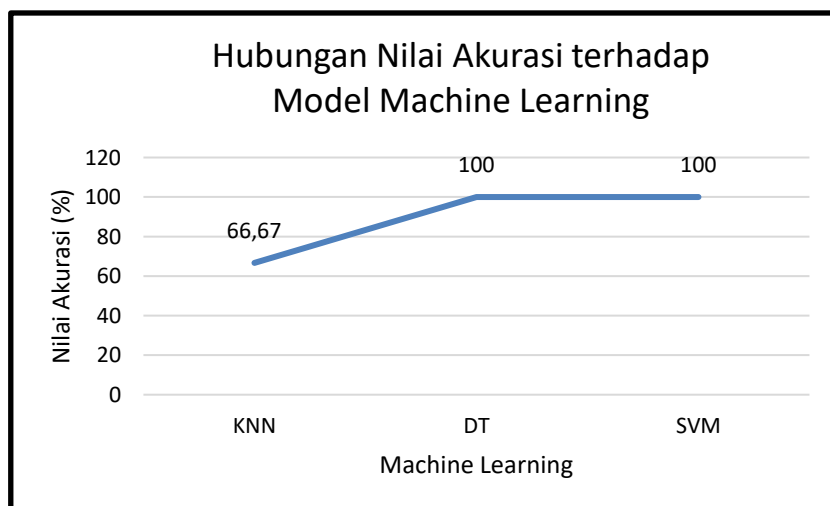
## 5. Assignment

- Studi literatur terkait *machine learning* serta metode pendekatannya seperti *supervised learning*, *unsupervised learning* dan *reinforcement learning* serta beberapa model *machine learning* seperti KNN, DT, dan SVM
- Membuat pemograman pada python dengan model machine learning yang digunakan adalah KNN
- Membuat program bisa mendeteksi beberapa warna secara simultan
- Membuat program yang sama namun model yang digunakan adalah DT
- Membuat program yang sama namun model yang digunakan adalah DT
- Melakukan uji coba hasil pemograman menggunakan Vscode
- Melakukan upload hasil praktikum pada repository Github

## 6. Data dan Output Hasil Pengamatan

No.	Keterangan	Hasil Percobaan
1.	Percobaan menggunakan KNN	<p>Color 1: magenta (100.0%)</p> <p>Color 2: yellowgreen (100.0%)</p> <p>Color 3: darkturquoise (100.0%)</p>

2.	Percobaan Menggunakan DT	 <p>Color 1: fuchsia (100.0%)</p> <p>Color 2: yellowgreen (100.0%)</p> <p>Color 3: darkturquoise (100.0%)</p>
3.	Percobaan menggunakan SVM	 <p>Color 1: fuchsia (8.0%)</p> <p>Color 2: yellowgreen (6.0%)</p> <p>Color 3: darkturquoise (7.3%)</p>



## 7. Kesimpulan

Berdasarkan praktikum yang telah dilakukan, disimpulkan bahwa penggunaan *machine learning* untuk klasifikasi warna menunjukkan hasil yang

cukup efektif. Model mampu mengenali warna utama dengan akurasi yang baik, meskipun masih terdapat kesulitan pada warna yang mirip atau memiliki saturasi rendah. Perbandingan beberapa model menunjukkan karakteristik yang berbeda; KNN sederhana dan efektif untuk dataset kecil, *Decision Tree* cepat tetapi rentan *overfitting*, sementara SVM memberikan akurasi tinggi terutama untuk data dengan batas kelas yang jelas. Faktor-faktor seperti jumlah dataset, resolusi kamera, dan kualitas data input berpengaruh signifikan terhadap kinerja model. Secara keseluruhan, praktikum ini menunjukkan bahwa machine learning dapat menjadi alat yang powerful untuk deteksi warna, dengan catatan perlu dilakukan optimasi dataset dan parameter model sehingga performa lebih optimal.

## **8. Saran**

Menambahkan jumlah dan variasi dataset sehingga model dapat mengenali warna dengan lebih akurat, termasuk variasi pencahayaan dan saturasi. Menggunakan resolusi kamera yang lebih tinggi untuk menangkap detail warna secara lebih jelas dan mengurangi noise. Menggunakan representasi warna lain seperti HSV atau LAB yang lebih sesuai dengan persepsi manusia, sehingga mempermudah pemisahan warna yang mirip.

## **9. Daftar Pustaka**

- Bintoro, P., Ratnasari, Wihardjo, E., Putri, I.P., Asari, A., 2024. Pengantar Machine Learning. PT MAFY MEDIA LITERASI INDONESIA, Solok.
- Dinata, R.K., Hasdyna, N., 2020. Machine Learning. Unimal Press, Lhokseumawe.
- Heymann, H., Kies, A.D., Frye, M., Schmitt, R.H., Boza, A., 2022. Guideline for Deployment of Machine Learning Models for Predictive Quality in Production. *Procedia CIRP* 107, 815–820. <https://doi.org/10.1016/j.procir.2022.05.068>
- S, T.A.T., 2019. Perbandingan Model Warna RGB, HSL dan HSV Sebagai Fitur dalam Prediksi Cuaca pada Citra Langit menggunakan K-Means.
- Shalev-Shwartz, S., Ben-David, S., 2014. Understanding Machine Learning: From Theory to Algorithms, 1st ed. Cambridge University Press. <https://doi.org/10.1017/CBO9781107298019>