

Laporan Praktikum Kontrol Cerdas

Minggu ke-2

Nama : Kresensia Meita Indar Mayaningsih

NIM : 224308087

Kelas : TKA 7D

Akun Github (Tautan) : <https://github.com/kzmeita>

Student Lab Assistant :

1. Judul Percobaan :

Machine Learning for Control Systems

2. Tujuan Percobaan

Tujuan dari praktikum minggu ke-2 :

1. Mahasiswa dapat memahami konsep dasar machine learning dalam sistem kendali.
2. Mahasiswa dapat memahami cara mengklasifikasi warna objek menggunakan library numpy pandas.
3. Mahasiswa dapat mengimplementasikan model machine learning untuk mengklasifikasi warna.
4. Mahasiswa dapat mengintegrasikan model machine learning dengan computer vision untuk deteksi objek terutama klasifikasi.
5. Mahasiswa dapat memahami cara mengambil datasheet dan mengimplementasikan ke program sebagai pelatihan model machine learning yang digunakan.
6. Mahasiswa dapat mengelola dan membuat datasheet untuk pelatihan sederhana.

3. Landasan Teori

Pembelajaran mesin (*machine learning*) merupakan cabang dari kecerdasan buatan (*artificial intelligence*) yang memungkinkan komputer untuk belajar dari data, mengenali pola, dan membuat prediksi tanpa diprogram secara eksplisit untuk setiap kondisi yang mungkin terjadi. Sistem pembelajaran mesin bekerja dengan cara membangun model matematis berdasarkan data historis, sehingga dapat melakukan generalisasi pada data baru yang sebelumnya belum pernah ditemui (Shaveta, 2023; Raj, 2019). Sebagai bidang yang interdisipliner, pembelajaran mesin menggabungkan konsep-konsep dari ilmu komputer, statistik, teori optimasi, serta inspirasi dari ilmu kognitif. Keunggulan dari pembelajaran mesin adalah setelah algoritma mempelajari pola dari data, sistem dapat bekerja secara

otomatis dan melakukan klasifikasi atau prediksi dengan tingkat akurasi tertentu (Batta, 2020).

Salah satu implementasi pembelajaran mesin yang populer adalah pada bidang pengolahan citra digital (*digital image processing*). Pengolahan citra adalah proses manipulasi gambar menggunakan komputer untuk mengekstraksi informasi yang berguna, misalnya warna, bentuk, maupun tekstur (Gonzalez & Woods, 2018). Dalam konteks ini, warna merupakan fitur visual yang penting karena sering digunakan sebagai dasar pengenalan objek, segmentasi citra, maupun analisis visual lainnya. Representasi warna dapat dinyatakan dalam berbagai model, seperti RGB (Red, Green, Blue) maupun HSV (Hue, Saturation, Value). Model HSV lebih sering digunakan dalam deteksi warna karena sesuai dengan cara manusia memandang warna, sehingga mempermudah segmentasi citra berbasis rentang warna tertentu.

Untuk mendukung proses pengolahan citra, salah satu pustaka (*library*) yang banyak digunakan adalah OpenCV (Open Source Computer Vision Library). OpenCV menyediakan berbagai fungsi untuk konversi ruang warna, deteksi tepi, segmentasi objek, hingga pengenalan pola visual (Bradski, 2000). Pada deteksi warna, OpenCV dapat digunakan untuk mengubah citra dari format RGB ke HSV, kemudian menyeleksi rentang nilai hue, saturation, dan value tertentu agar sistem dapat mengenali warna yang sesuai.

Selain OpenCV, dalam implementasi pembelajaran mesin modern, sering digunakan pula Convolutional Neural Network (CNN). CNN merupakan salah satu jenis jaringan syaraf tiruan (*artificial neural network*) yang dirancang khusus untuk mengolah data berbentuk grid, seperti citra. CNN bekerja melalui lapisan konvolusi yang mengekstraksi fitur spasial, lapisan pooling yang mereduksi dimensi, dan lapisan fully connected yang berfungsi sebagai klasifikator (LeCun et al., 2015). Dengan kemampuan tersebut, CNN sangat efektif dalam mendeteksi pola visual, termasuk pengenalan warna maupun objek kompleks.

Dalam mendukung pengolahan data, library Pandas juga memegang peran penting. Pandas merupakan pustaka Python yang menyediakan struktur data efisien seperti *DataFrame* dan *Series*, yang memudahkan analisis serta manipulasi data dalam skala besar. Pada proses deteksi warna berbasis machine learning, Pandas dapat digunakan untuk mengelola dataset warna, menyimpan informasi berupa nilai RGB atau HSV beserta label warna, serta memfasilitasi tahap pra-pemrosesan data sebelum model dilatih. Dengan demikian, Pandas menjadi alat yang esensial dalam pipeline analisis data modern (McKinney, 2011).

Sumber data yang digunakan dalam penelitian berbasis machine learning seringkali berasal dari dataset publik, salah satunya yang banyak tersedia di platform Kaggle. Kaggle merupakan sebuah komunitas sains data global yang menyediakan berbagai dataset terbuka, kompetisi analitik, serta forum diskusi. Dataset warna dari Kaggle biasanya terdiri atas ribuan entri yang memuat nilai RGB/HSV serta label warna yang sesuai, yang dapat digunakan untuk melatih model klasifikasi. Melalui dataset ini, sistem dapat belajar mengenali warna berdasarkan pola nilai numerik yang direpresentasikan dalam format tertentu. Tahap pelatihan model melibatkan pembagian data ke dalam subset data latih (*training set*) dan data uji (*testing set*) untuk memastikan model memiliki kemampuan generalisasi yang baik.

Evaluasi sistem deteksi warna dilakukan menggunakan metrik seperti akurasi, presisi, recall, dan F1-score. Akurasi merupakan metrik paling sederhana, yang mengukur persentase prediksi yang benar dibandingkan jumlah total data uji. Misalnya, jika sistem mendeteksi warna “Ivory” dengan akurasi 18,36%, berarti hanya sebagian kecil dari prediksi yang sesuai dengan label sebenarnya. Nilai akurasi ini dapat ditingkatkan dengan menambah jumlah data latih, melakukan augmentasi data, atau menggunakan arsitektur model yang lebih kompleks.

Dengan memadukan konsep pembelajaran mesin, pengolahan citra, pustaka pendukung seperti OpenCV dan Pandas, serta pemanfaatan dataset publik dari Kaggle, sistem deteksi warna dapat dikembangkan untuk berbagai aplikasi. Aplikasi ini mencakup sistem inspeksi kualitas produk di industri, pengenalan objek berbasis warna dalam sistem transportasi cerdas, hingga interaksi manusia-komputer berbasis visual. Landasan teori ini menjadi dasar penting dalam perancangan sistem deteksi warna yang mampu berfungsi secara akurat dan efisien pada lingkungan nyata.

4. Analisis dan Diskusi

A. Analisis

Pada praktikum ini, kami menggunakan pendekatan machine learning untuk melakukan klasifikasi warna objek. Library utama yang digunakan adalah NumPy, Pandas, dan OpenCV dengan bahasa pemrograman Python. NumPy berperan dalam perhitungan numerik yang efisien, Pandas digunakan untuk pengelolaan dataset warna dalam bentuk DataFrame, sedangkan OpenCV digunakan untuk pengolahan citra dan deteksi warna secara langsung dari kamera atau gambar.

Proses analisis dimulai dengan menginstall library yang digunakan yaitu numpy panda dengan memasukkan sintak *"pip install numpy pandas matplotlib scikit-learn opencv-python"* pada command prompt. Kemudian setelah terinstall, kami menuju ke Kaggle dan melakukan registrasi terlebih dahulu. Setelah selesai, kami menuju setting dan membuat API token agar datasheet yang kita download bisa digunakan. Kemudian, kami menggunakan datasheet yang tersedia. Namun terdapat problem dimana datasheet bersifat private tidak public, sehingga datasheet tersebut tidak dapat diakses. Maka, sebagai solusi dari permasalahan tersebut kami menggunakan datasheet lain. Dengan memanfaatkan dataset warna yang diambil dari Kaggle dan kemudian dikelola menggunakan Pandas agar data lebih terstruktur. Data warna dalam format RGB atau HSV kemudian dijadikan dasar bagi model machine learning untuk mengenali pola warna tertentu. Model dilatih menggunakan data latih (training set) dan diuji dengan data uji (testing set) untuk mengukur akurasi klasifikasi.

Selanjutnya, sistem diuji pada objek nyata dengan berbagai warna. Hasilnya ditampilkan dalam jendela frame berupa bounding box yang mengelilingi objek serta teks yang menunjukkan klasifikasi warna beserta tingkat akurasinya. Hasil praktikum menunjukkan bahwa sistem mampu mengenali warna dasar (merah, hijau, biru) dan dapat diperluas ke warna sekunder seperti ungu, putih, hitam, hingga warna kulit. Nilai akurasi yang diperoleh bervariasi, tergantung pada kualitas pencahayaan dan jumlah data pelatihan yang tersedia.

B. Diskusi

Hasil praktikum memperlihatkan bahwa integrasi machine learning dengan computer vision memberikan peluang besar dalam pengembangan sistem deteksi berbasis warna. Keunggulan utama terletak pada fleksibilitas algoritma machine learning yang mampu beradaptasi dengan data baru, sehingga sistem dapat mendeteksi berbagai variasi warna di luar dataset awal. Namun, keterbatasan masih muncul terutama pada kondisi pencahayaan yang tidak konsisten, di mana akurasi klasifikasi menurun.

Penggunaan OpenCV terbukti sangat membantu dalam proses segmentasi dan konversi ruang warna dari RGB ke HSV, karena HSV lebih stabil terhadap perubahan intensitas cahaya. Pandas juga memberikan kontribusi signifikan dalam pengelolaan dataset, mempermudah tahap pra-pemrosesan, normalisasi, hingga evaluasi model. Dengan demikian, kombinasi kedua library ini menciptakan alur kerja yang efektif untuk mendukung machine learning dalam kontrol sistem.

Dari sisi dataset, penggunaan sumber dari Kaggle memudahkan proses pelatihan model. Namun, dataset publik tidak selalu sesuai dengan kondisi nyata yang dihadapi di laboratorium. Oleh karena itu, diperlukan pembuatan dataset tambahan dengan mengambil data langsung dari lingkungan percobaan untuk meningkatkan representasi data. Strategi ini dapat meningkatkan akurasi sistem dalam kondisi pencahayaan yang beragam dan objek dengan warna campuran.

Secara keseluruhan, praktikum ini membuktikan bahwa metode machine learning dapat diimplementasikan secara praktis pada sistem kontrol berbasis computer vision. Tantangan utama yang masih perlu diatasi adalah optimasi akurasi dalam kondisi nyata serta efisiensi komputasi untuk penerapan real-time, terutama jika dijalankan pada perangkat keras dengan sumber daya terbatas seperti Raspberry Pi.

5. Assignment

Pada tugas praktikum minggu ini, dilakukan pengembangan lebih lanjut dari program deteksi warna berbasis machine learning dengan beberapa modifikasi dan eksplorasi tambahan. Rincian assignment adalah sebagai berikut:

A. Modifikasi Model Machine Learning

- Program awal yang menggunakan metode klasifikasi sederhana dimodifikasi dengan mengganti model machine learning menjadi Decision Tree Classifier dan Support Vector Machine (SVM).
- Decision Tree digunakan untuk membangun aturan klasifikasi yang jelas berdasarkan nilai RGB/HSV, sedangkan SVM digunakan untuk membentuk hyperplane yang memisahkan kelas warna dengan margin optimal.
- Hasil pengujian menunjukkan bahwa SVM memberikan akurasi lebih baik pada dataset warna dengan jumlah kelas yang banyak, sementara Decision Tree lebih mudah dipahami dan divisualisasikan.

B. Deteksi Multi-Warna Simultan

- Program diperluas agar dapat mendeteksi lebih dari satu objek berwarna dalam satu frame kamera.
- Setiap objek yang terdeteksi diberikan bounding box dengan label warna dan tingkat akurasi klasifikasi.
- Fitur ini memungkinkan sistem untuk bekerja pada kondisi nyata, misalnya ketika ada beberapa objek dengan warna berbeda yang muncul secara bersamaan.

C. Eksplorasi Dataset Lain di Kaggle

- Selain dataset warna standar, dilakukan eksplorasi pada dataset lain dari Kaggle untuk menguji generalisasi model.

- Dataset yang dipilih memiliki variasi warna lebih banyak dan kondisi pencahayaan yang beragam.
- Implementasi pada dataset baru ini menunjukkan pentingnya data preprocessing seperti normalisasi, balancing, dan augmentasi data agar model tetap stabil.

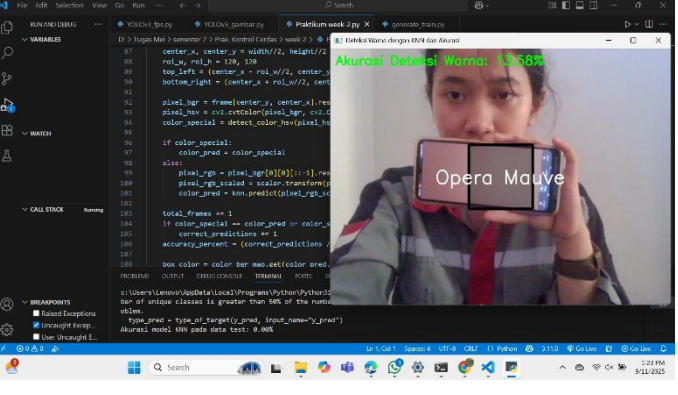
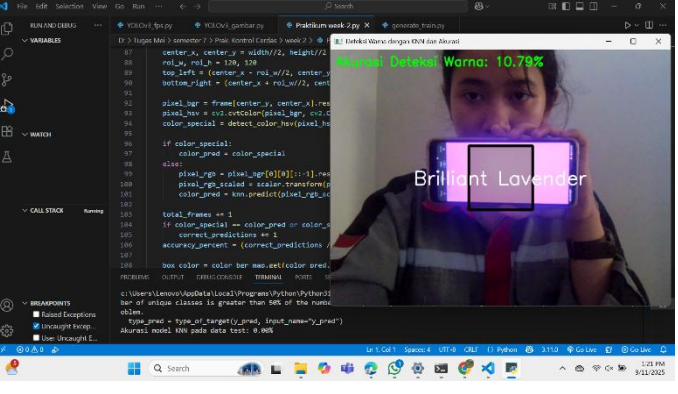
D. Integrasi GitHub dan Dokumentasi

- Hasil eksperimen, termasuk kode program, dataset yang digunakan, dan grafik evaluasi model diunggah ke repository GitHub pribadi.
- Sebuah laporan singkat dibuat untuk mendokumentasikan langkah-langkah modifikasi, hasil pengujian, serta perbandingan kinerja model.
- Laporan ini menjadi bukti kolaborasi serta transparansi eksperimen yang memudahkan replikasi oleh mahasiswa lain.

E. Data dan Output Hasil Pengamatan

Data yang diperoleh pada praktikum minggu ke-1 disajikan dalam tabel dibawah ini:

No	Variabel	Hasil Pengamatan
1.	Pada percobaan pertama, saya menggunakan warna orange untuk dilakukan uji coba. Ternyata program mendeteksi warna tersebut sebagai warna ivory dengan akurasi warna 18,36%.	
2.	Pada percobaan kedua, saya menggunakan warna putih dan melakukan uji coba. Didapatkan hasil warna tersebut termasuk dalam warna ghost white dengan akurasi warna 19.11%.	

3.	<p>Pada percobaan ketiga, saya menggunakan warna coklat untuk dilakukan uji coba. Didapatkan hasil warna tersebut termasuk dalam warna Opera Mauve dengan akurasi warna 13.58%.</p>	
4.	<p>Pada percobaan terakhir, saya menggunakan warna ungu untuk dilakukan uji coba. Hasil yang didapatkan warna tersebut termasuk dalam Brilliant Lavender dengan akurasi warna 10.79%.</p>	

1. Kesimpulan

1. Praktikum minggu ke-2 berhasil memperlihatkan bagaimana machine learning dapat digunakan untuk klasifikasi warna dengan bantuan library Python seperti NumPy, Pandas, dan OpenCV.
2. Sistem deteksi warna mampu mengenali warna dasar (merah, hijau, biru) dan dimodifikasi untuk mendeteksi warna sekunder maupun kompleks (putih, hitam, ungu, warna kulit).
3. Penggunaan dataset dari Kaggle membantu dalam proses pelatihan model, namun hasil akurasi sangat dipengaruhi oleh kondisi pencahayaan dan jumlah data latih yang tersedia.
4. Pandas terbukti mempermudah pengelolaan dataset, sedangkan OpenCV efektif dalam segmentasi dan ekstraksi fitur warna.
5. Praktikum ini menunjukkan bahwa integrasi machine learning dengan computer vision dapat menjadi dasar bagi pengembangan sistem kontrol yang lebih kompleks, seperti deteksi objek real-time dalam transportasi atau industri.

2. Saran

1. Perlu dilakukan penambahan jumlah dataset baik dari Kaggle maupun hasil akuisisi langsung agar model lebih robust terhadap variasi warna dan kondisi pencahayaan.

2. Lakukan augmentasi data (rotasi, perubahan brightness/contrast) untuk meningkatkan generalisasi model.
3. Gunakan arsitektur CNN yang lebih dalam atau model ringan seperti MobileNet/YOLOv5s untuk mendukung deteksi real-time dengan akurasi tinggi.
4. Uji sistem pada berbagai kondisi pencahayaan untuk mengetahui stabilitas model di lingkungan nyata.
5. Implementasi lebih lanjut dapat diarahkan pada integrasi dengan perangkat keras seperti Raspberry Pi agar sistem dapat diaplikasikan di lapangan secara langsung.

3. Daftar Pustaka

- Saldaña, E., Siche, R., Huamán, R., Luján, M., Castro, W., & Quevedo, R. (2013). Computer vision system in real-time for color determination on flat surface food. *Scientia Agropecuaria*, 55-63.
- Aryeni, I., Maulidiah, H., Toar, H., Wicaksono, M., & Gunawan, I. (2023). Application of Computer Vision for Real-Time Detection of Fruit Color and Size in Fruit Sorter. *Journal of Applied Electrical Engineering*, 2548-9682.
- Batta, M. (2020). Machine Learning Algorithms - A Review. *International Journal of Computer Applications*.
- Diers, J., & Pigorsch, C. (2023). A Survey of Methods for Automated Quality Control Based on Images. *International Journal of Computer Vision*, 2553–2581.
- Joy, D., Kaur, G., Chugh, A., & Bajaj, S. (2021). Computer Vision for Color Detection. *International Journal of Innovative Research in Computer Science & Technology*, 53-59.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep Learning. *Nature*, 436-444.
- Lynden, S., & Taveekarn, W. (2019). *Semi-automated Augmentation of Pandas DataFrames*. Singapore: Data Mining and Big Data.
- Maitlo, N., Noonari, N., Ahmed, S., Duraisamy, S., & Ahmed, F. (2024). Color Recognition in Challenging Lighting Environments: CNN Approach. *Institute of Electrical and Electronics Engineers*.