

# Laporan Praktikum Kontrol Cerdas

Nama : Aureyza Pandu Qinara

NIM : 224308078

Kelas : TKA - 7D

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

Student Lab Assistant : -

## 1. Judul Percobaan

Week 1: Deteksi Objek Warna menggunakan OpenCV

## 2. Tujuan Percobaan

Tujuan dari praktikum “Deteksi Objek Warna menggunakan OpenCV”, mahasiswa diharapkan mampu:

- a) Mengimplementasikan deteksi objek berbasis warna menggunakan OpenCV pada Python.
- b) Menggunakan ruang warna HSV untuk segmentasi warna pada citra digital.
- c) Melakukan masking untuk mendeteksi beberapa warna (merah, biru, hijau, kuning) secara *real-time*.
- d) Mengidentifikasi dan menandai objek berbentuk persegi berdasarkan hasil deteksi warna.

## 3. Landasan Teori

- OpenCV (*Open Source Computer Vision Library*)

OpenCV adalah pustaka *open source* yang banyak digunakan untuk pengolahan citra dan visi komputer. Pustaka ini mendukung berbagai fungsi seperti deteksi objek, pengenalan pola, segmentasi, hingga pelacakan gerakan (Zidanul Akbar et al., 2025). Pada percobaan ini, OpenCV digunakan untuk mengakses kamera, melakukan konversi warna, segmentasi, serta mendeteksi

bentuk persegi pada objek berwarna tertentu.

- Ruang Warna HSV (*Hue, Saturation, Value*)

Citra yang ditangkap kamera awalnya berada pada ruang warna BGR (*Blue, Green, Red*). Untuk mempermudah segmentasi warna, citra diubah ke ruang HSV dengan fungsi `cv2.cvtColor()`. HSV memisahkan corak warna (*hue*), tingkat kejenuhan (*saturation*), dan kecerahan (*value*), sehingga lebih efektif dalam mendeteksi warna meski pencahayaan berubah (James et al., 2025).

- *Masking* dan Segmentasi Warna

Segmentasi dilakukan dengan menentukan rentang HSV untuk setiap warna target (merah, biru, hijau, kuning). Fungsi `cv2.inRange()` menghasilkan *mask* biner, di mana piksel sesuai warna target bernilai putih, dan lainnya hitam. Dengan ini, objek berwarna dapat dipisahkan dari latar belakang.

- Deteksi Kontur dan Pengenalan Bentuk

Fungsi `cv2.findContours()` digunakan untuk menemukan tepi objek dari hasil *masking*. Selanjutnya, `cv2.approxPolyDP()` digunakan untuk mendekati kontur menjadi poligon sederhana. Jika jumlah sisi poligon adalah empat dan memiliki rasio panjang-lebar mendekati 1, objek dikenali sebagai persegi.

- Bounding Box dan Pemberian Label

Untuk menandai objek, digunakan fungsi `cv2.boundingRect()` yang menggambar kotak di sekitar persegi terdeteksi. Fungsi `cv2.putText()` kemudian menambahkan label berupa nama warna sesuai objek. Dengan kombinasi ini, program mampu menampilkan persegi berwarna lengkap dengan identitas warnanya.

## 4. Analisis dan Diskusi

Program ini merupakan implementasi deteksi persegi berwarna menggunakan OpenCV. Proses dimulai dengan pengambilan frame kamera secara real-time yang kemudian dikonversi ke format HSV agar

deteksi warna lebih stabil. Rentang HSV ditentukan untuk merah, biru, hijau, dan kuning, lalu dilakukan masking dan penggabungan hasil untuk memperoleh area objek sesuai warna.

Deteksi persegi dilakukan dengan mencari kontur pada citra hasil segmentasi, kemudian memeriksa jumlah sisi dan rasio aspek. Jika memenuhi kriteria, program menggambar bounding box dan memberi label warna pada objek tersebut. Hasilnya ditampilkan pada tiga jendela: frame asli dengan bounding box, gabungan hasil segmentasi, dan citra masking per warna.

Program cukup efektif pada kondisi pencahayaan baik dan objek berukuran besar, namun masih terbatas pada rentang HSV statis dan sulit mendeteksi objek kecil. Pengembangan selanjutnya dapat mencakup penggunaan operasi morfologi untuk mengurangi noise, penyesuaian HSV adaptif, serta deteksi bentuk geometris lain.



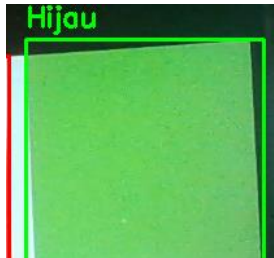
## 5. Assignment


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.

## 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	Merah	
2	Biru	
3	Hijau	

No	Variabel	Hasil Pengamatan
4	Kuning	

## 7. Kesimpulan

1. Program berhasil mengimplementasikan deteksi objek berbasis warna menggunakan OpenCV pada Python, sehingga objek dengan warna tertentu dapat dikenali secara *real-time*.
2. Konversi citra ke ruang warna HSV terbukti efektif digunakan dalam segmentasi warna, karena lebih stabil terhadap perubahan pencahayaan dibandingkan ruang warna BGR.
3. Proses *masking* mampu mendeteksi dan memisahkan beberapa warna target (merah, biru, hijau, dan kuning) dari latar belakang secara simultan.
4. Objek berbentuk persegi dapat diidentifikasi dan ditandai dengan *bounding box* serta label warna, sehingga hasil deteksi lebih informatif dan mudah dipahami.

## 8. Saran

1. Program dapat dikembangkan lebih lanjut untuk mendeteksi bentuk lain seperti segitiga, lingkaran, atau persegi panjang.
2. Segmentasi warna dapat diperluas dengan teknik *adaptive thresholding* atau *color calibration* agar lebih tahan terhadap perubahan pencahayaan.
3. Sistem dapat diintegrasikan dengan algoritma *tracking* sehingga objek berwarna dapat dilacak pergerakannya secara *real-time*.
4. Disarankan menggunakan kamera dengan resolusi lebih tinggi untuk meningkatkan akurasi deteksi bentuk dan warna.

## **10. Daftar Pustaka**

- James, A., Anto, A., Pauly, A.M., Anto, S.E., 2025. Color Recognition Methods In Digital Images: A Review Of Classical And Deep Learning Approaches 11.
- Zidanul Akbar, Asrul Suwondo, Rizky Ramadhan, Abdul Halim Hasugian, 2025. Deteksi Warna Dasar Menggunakan Metode Thresholding HSV dengan OpenCV. *Neptunus J. Ilmu Komput. Dan Teknol. Inf.* 3, 256–264. <https://doi.org/10.61132/neptunus.v3i3.1020>