

# Laporan Praktikum Kontrol Cerdas

Nama : Bintang Anugerah B

NIM : 224308030

Kelas : TKA-6B

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

Student Lab Assistant : Mas Ali

## 1. Judul Percobaan : Deteksi Objek Sederhana dengan OpenCV

## 2. Tujuan Percobaan :

- Memahami konsep dasar *Intelligent Control System*.
- Memahami penerapan *Artificial Intelligence* (AI) dalam system kendali.
- Menggunakan Python dan OpenCV untuk mendeteksi objek secara sederhana.
- Memahami bagaimana OpenCV digunakan untuk analisis gambar.
- Menampilkan hasil deteksi secara *real-time* menggunakan kamera.

## 3. Landasan Teori :

*Artificial Intelligence* (AI), atau dalam bahasa Indonesia dikenal sebagai Kecerdasan Buatan, adalah cabang ilmu komputer yang bertujuan untuk mengembangkan sistem dan mesin yang mampu melakukan tugas yang biasanya memerlukan kecerdasan manusia. AI melibatkan penggunaan algoritma dan model matematika untuk memungkinkan komputer dan sistem lainnya untuk belajar dari data, mengenali pola, dan membuat keputusan yang cerdas (**Eriana et al., n.d.**). *Intelligent System* atau sistem cerdas adalah sebuah sistem yang dihasilkan melalui sebuah rekayasa untuk menghasilkan mesin yang memiliki kecerdasan tertentu yang disusun dari program-program komputer sehingga menghasilkan sebuah mesin yang cerdas (**Budi & Santiyadnya, n.d.**). Sementara, *software* yang digunakan dalam praktikum ini yaitu *python* , *Visual Studio Code* dan OpenCV.

*Python* adalah bahasa pemrograman dengan tujuan umum (*general*). Itu berarti *python* dapat digunakan untuk menulis kode pada segala kebutuhan pemrograman. *Python* diciptakan oleh Guidovan Rossum di Belanda pada tahun 1990. *Python* menjadi Bahasa

pemrograman populer yang digunakan di industry dan akademik karena mudah, ringkas, dan memiliki library yang luas seperti 'NumPy' dan 'Pandas' untuk analisis data (Cahyo & Kholis, 2019). Sementara itu, *Visual Studio Code* adalah *Software* yang sangat ringan, namun kuat *editor* kode sumbernya yang berjalan dari *desktop*. *Visual Studio Code* digunakan untuk pembuatan kode-kode program dibutuhkan sebuah aplikasi yang mumpuni. *Visual Studio Code* dapat digunakan untuk berbagai Bahasa pemrograman seperti Java Script, HTML, CSS, PHP, Python, C++, dan masih banyak lagi (Syarif et al., 2023). *OpenCV (Open Source Computer Vision Library)* adalah sebuah pustaka perangkat lunak khusus untuk computer vision, yang ditujukan untuk pengolahan citra dinamis secara *real-time*. *OpenCV* memiliki lebih dari 25000 algoritma yang dioptimalkan untuk mendeteksi, mengidentifikasi dan mengenali wajah (Mutasil et al., 2021).

#### 4. Analisis dan Diskusi :

##### A. Analisis

Pada Mata Kuliah Praktikum Kontrol Cerdas minggu pertama ini kami melakukan percobaan mendeteksi objek berwarna biru secara *real-time* dengan menggunakan *OpenCV*. Untuk menampilkan kamera kita menginisialisasi menggunakan fungsi `cv2.VideoCapture`. Kemudian, untuk mengkonversi warna menggunakan fungsi `cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)`, pada program ini kita menggunakan HSV karena warnanya lebih stabil terhadap perubahan daripada BGR Sementara untuk rentang warna biru didefinisikan dengan nilai HSV yaitu `lower_blue = np.array([100, 150, 50])` dan `upper_blue = np.array([140, 255, 255])` Selanjutnya, untuk hasilnya akan ditampilkan dalam tiga *tab* yang berbeda yaitu *frame* untuk menampilkan kamera asli, *mask* untuk menampilkan hasil dari warna biru, dan *result* untuk menampilkan hasil dari penyaringan warna. Program ini akan terus berjalan hingga pengguna menekan tombol 'q' yang akan melepaskan akses kamera dan menutup tiga tab.

Kemudian, terdapat fitur tambahan berupa *bounding box* untuk menandai objek warna biru yang terdeteksi. Setelah melakukan konversi dari BGR ke HSV, dengan menggunakan fungsi `cv2.findContours()` akan mendeteksi kontur area/benda yang memiliki warna biru, sedangkan untuk memunculkan fitur *bounding box* menggunakan fungsi `cv2.boundingRect()` untuk memberikan koordinat dan ukuran kotak pada objek yang berwarna biru tersebut. *Bounding box* ini akan muncul pada frame asli

menggunakan fungsi `cv2.rectangle()` dengan nilai biru (255, 0, 0). Adanya penambahan fitur *bounding box* ini meningkatkan akurasi deteksi benda, karena tidak hanya menampilkan area yang memiliki warna biru, tetapi juga menandai objek dengan jelas.

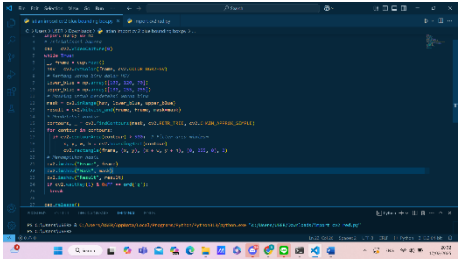

## B. Diskusi


Metode deteksi objek berwarna biru dengan OpenCV dan fitur *bounding box* memiliki kelebihan dan kekurangan. Salah satu kelebihan adalah kemudahan dalam mendeteksi objek berdasarkan warna tertentu. Selain itu, penggunaan HSV membuat sistem lebih stabil terhadap perubahan pencahayaan dibandingkan dengan BGR. Selain itu, *bounding box* memungkinkan objek yang terdeteksi ditandai dengan jelas, sehingga memudahkan proses identifikasi dan verifikasi dalam suatu *frame*. Namun, metode ini memiliki keterbatasan, seperti kesalahan deteksi akibat perubahan pencahayaan atau keberadaan objek lain dengan warna serupa. Selain itu, metode berbasis warna saja tidak mampu membedakan objek berdasarkan bentuk, sehingga dapat mengurangi tingkat keakuratan sistem.

## 5. Assignment :

Pada praktikum kali ini dengan percobaan mendeteksi objek sederhana berbasis warna menggunakan OpenCV. Selain itu, menggunakan beberapa *software* lain seperti *python* dan *Visual Studi Code*. Sistem ini dirancang untuk mendeteksi warna khususnya warna biru dengan tambahan fitur *bounding box*. Pertama- tama kita menginisialisasi kamera, yang kemudian dikonversi ke warna HSV, setelah itu dilakukan masking berdasarkan rentang warna biru dan ditambahkan fitur *bounding box* menggunakan fungsi `cv2.boundingRect()` dan label teks “biru” untuk memberikan informasi visual mengenai hasil deteksi. Dengan adanya *bounding box*, sistem menjadi lebih informatif, karena pengguna dapat dengan jelas melihat objek yang dideteksi dalam kamera. Akan tetapi, terdapat beberapa kekurangan seperti kemungkinan deteksi objek yang tidak diinginkan, seperti latar belakang atau pencahayaan memiliki warna yang serupa.

## 6. Data dan Output Hasil Pengamatan :

No.	Variabel	Hasil Pengamatan
1.	<ul style="list-style-type: none"> <li>- Menginisialisasi kamera, yang akan dikonversi ke warna HSV, dengan fungsi <code>cv2.cvtColor(frame, cv2.COLOR_BGR2HSV)</code></li> <li>- Rentang warna biru dalam HSV yaitu : <ul style="list-style-type: none"> <li>• <code>lower_blue = np.array([100, 150, 50])</code></li> <li>• <code>upper_blue = np.array([140, 255, 255])</code></li> </ul> </li> <li>- Masking mendeteksi warna biru yaitu dengan fungsi <code>mask = cv2.inRange(hsv, lower_blue, upper_blue)</code>  <code>result = cv2.bitwise_and(frame, frame, mask=mask)</code></li> <li>- Kemudian untuk menampilkan hasil yaitu : <ul style="list-style-type: none"> <li>• <code>cv2.imshow("Frame", frame)</code></li> <li>• <code>cv2.imshow("Mask", mask)</code></li> <li>• <code>cv2.imshow("Result", result)</code></li> </ul> </li> <li>- Menekan tombol 'q' yang akan menghentikan akses kamera dan menutup tiga tab <code>cv2.waitKey(1) &amp; 0xFF == ord('q')</code></li> </ul>	 

2.	<p>- Menemukan kontur warna biru yaitu <code>cv2.findContours(mask, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)</code></p> <p>- Untuk memunculkan fitur <i>bounding box</i> dengan menggunakan fungsi <code>vc2.boundingRect(contour)</code></p>	
----	--	--

## 7. Kesimpulan :

Berdasarkan praktikum dan analisis yang telah dilakukan, maka dapat diambil kesimpulan yaitu:

- Menunjukkan kemampuan sistem dalam mendeteksi objek berwarna biru secara *real-time* dengan menggunakan OpenCV.
- Fitur *bounding box* sangat meningkatkan keakuratan deteksi dengan menandai objek yang terdeteksi, sehingga memudahkan proses identifikasi dan verifikasi.
- Metode ini memiliki kelebihan dalam mendeteksi objek berdasarkan warna tertentu, namun juga memiliki keterbatasan seperti kemungkinan kesalahan deteksi akibat perubahan pencahayaan atau keberadaan objek lain dengan warna serupa.

## 8. Saran :

Saran untuk pengembangan sistem deteksi objek ini adalah agar dilakukan integrasi teknik deteksi yang lebih canggih, seperti *machine learning*, untuk membedakan objek berdasarkan bentuk dan warna, serta melakukan pengujian dalam berbagai kondisi pencahayaan. Selain itu, penggunaan kamera berkualitas tinggi dapat meningkatkan hasil gambar, dan penerapan sistem umpan balik dapat membantu memperbaiki algoritma berdasarkan kesalahan deteksi yang ditemukan.

## 9. Daftar Pustaka :

- Budi, C. S., & Santiyadnya, N. (n.d.). PENGEMBANGAN MEDIA PEMBELAJARAN SMART CONTROL SYSTEM BERBASIS IoT YANG BERSIFAT ADAPTIF UNTUK MENGATUR SUHU DAN KELEMBABAN TANAH PADA MATA KULIAH SISTEM KENDALI CERDAS.
- Cahyo, O. B. D., & Kholis, N. (2019). RANCANG BANGUN SIMULATOR ELEKTROKARDIOGRAM MENGGUNAKAN FPGA YANG TERINTEGRASI DENGAN SOFTWARE PYTHON. 08.
- Eriana, E. S., Kom, S., Kom, M., Zein, D. A., & Kom, M. (n.d.). ARTIFICIAL INTELLIGENCE (AI).
- Mutasil, A., Irsan, M., & Sujana, D. (2021). Pengenalan Wajah Menggunakan Opencv Untuk Validasi Peserta Ujian Penerimaan Mahasiswa Baru. Jurnal SISKOM-KB (Sistem Komputer dan Kecerdasan Buatan), 5(1), 21–28. <https://doi.org/10.47970/siskom-kb.v5i1.221>
- Syarif, M. N., Pambudiyatno, N., & Utomo, W. (2023). RANCANGAN SISTEM PRESENSI DAN REKAPITULASI JURNAL KEGIATAN OJT MENGGUNAKAN VISUAL STUDIO CODE BERBASIS WEB DI AIRNAV CABANG MATSC.