



Perancangan Sistem Deteksi Warna Real-Time Menggunakan Metode Gaussian Blur Dan Ruang Warna HSV

Ferdi Nur Rochim¹, Grivi Desky Sompie², Mua'mmar^{3*}, Rizky Imam Saputra⁴, Perani Rosyani⁵

^{1,2,3,4,5}Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Pamulang, Tangerang Selatan, Indonesia

Email: ¹ferdinurrochim@gmail.com, ²grivisompie189@gmail.com, ³muamar.idm008@gmail.com, ⁴imamsaput55@gmail.com, ⁵dosen00837@unpam.ac.id

(* : coresponding author : muamar.idm008@gmail.com)

Abstrak – Deteksi warna adalah proses dalam pengenalan objek berdasarkan karakteristik warna dalam sebuah citra. Penelitian ini merancang sistem deteksi warna real-time menggunakan OpenCV dan HSV color space untuk membantu individu mengenali warna dengan lebih mudah. Sistem ini memanfaatkan teknik pengolahan citra seperti Gaussian blur dan operasi morfologi untuk meningkatkan akurasi deteksi. Sistem ini dirancang untuk mendeteksi dan mengenali beberapa warna termasuk merah, hijau, biru, kuning, oranye, ungu, coklat, hitam, putih, dan abu-abu. Hasil deteksi ditampilkan dengan menggambar kotak pembatas di sekitar objek yang terdeteksi serta memberikan label warna pada objek tersebut dalam bentuk teks yang jelas dan mudah dibaca. Sistem ini diimplementasikan dengan mengakses kamera secara langsung, memproses setiap frame secara real-time, dan menampilkan hasil deteksi pada layar. Uji coba menunjukkan bahwa sistem ini memiliki kinerja yang memuaskan dalam kondisi pencahayaan yang baik dan mampu mendeteksi objek dengan berbagai ukuran, sehingga dapat memberikan bantuan visual yang signifikan dalam mengidentifikasi dan membedakan warna di lingkungan sekitar.

Kata Kunci: Deteksi Warna, Pengolahan Real-Time, OpenCV, Ruang Warna HSV, Pengolahan Citra

Abstract – Color detection is a crucial process in recognizing objects based on their color characteristics in an image. This research designs a real-time color detection system using OpenCV and HSV color space to aid individuals in easily identifying colors. The system employs image processing techniques such as Gaussian blur and morphological operations to enhance detection accuracy. Designed to detect and recognize multiple colors including red, green, blue, yellow, orange, purple, brown, black, white, and gray, the system displays detection results by drawing bounding boxes around detected objects and providing clear and readable color labels. Implemented by accessing the camera directly, processing each frame in real-time, and displaying detection results on-screen, the system demonstrates satisfactory performance in good lighting conditions and is capable of detecting objects of various sizes, thereby providing significant visual assistance in color identification and differentiation in the surrounding environment.

Keywords: Color Detection, Real-Time Processing, OpenCV, HSV Color Space, Image Processing

1. PENDAHULUAN

Deteksi warna merupakan proses penting dalam mengenali objek berdasarkan karakteristik warna dalam sebuah citra (Panggabean et al., 2021). Menggunakan OpenCV, sebuah library yang umum digunakan dalam bidang komputer vision, sistem deteksi warna dapat diimplementasikan secara real-time dengan memanfaatkan ruang warna HSV (Hue, Saturation, Value) (Rabbani et al., 2021). Ruang warna ini memungkinkan pemisahan informasi warna dari kecerahan (brightness), sehingga sistem lebih efektif dalam menangani variasi pencahayaan yang mempengaruhi kualitas citra. Ruang lingkup warna HSV terdiri dari tiga elemen yaitu Hue yang mewakili warna, Saturation yang mewakili tingkat dominasi warna, dan Value yang mewakili tingkat kecerahan. Dengan demikian, metode ini cenderung mendeteksi warna beserta tingkat dominasi dan kecerahannya (Utami, 2023).

Untuk mendapatkan akurasi yang baik dalam proses deteksi objek, diperlukan pre-processing seperti pembersihan noise (Rosyani et al., 2021). Teknik-teknik pre-processing seperti Gaussian blur sering digunakan untuk mengatasi masalah ini dengan mengaplikasikan fungsi Gaussian pada gambar (Rosyani et al., 2024). Gaussian Blur digunakan untuk mengurangi noise, sementara operasi morfologi membantu memperbaiki struktur objek yang terdeteksi, yang pada akhirnya meningkatkan akurasi dan keandalan deteksi warna dalam berbagai kondisi lingkungan (Yasir et al.,



2023). Implementasi sistem ini memungkinkan pengenalan berbagai warna seperti merah, hijau, biru, kuning, oranye, ungu, coklat, hitam, putih, dan abu-abu. Sistem dapat mengolah setiap frame secara real-time dari kamera langsung, menampilkan hasil deteksi dengan kotak pembatas yang jelas, dan memberikan label warna pada objek yang terdeteksi. Pendekatan ini tidak hanya meningkatkan keakuratan deteksi, tetapi juga meningkatkan efisiensi dalam mengidentifikasi objek dengan berbagai ukuran dan kompleksitas. Dengan demikian, deteksi warna menggunakan OpenCV dan ruang warna HSV menjadi pendekatan yang andal untuk memenuhi berbagai kebutuhan dalam analisis citra dan aplikasi pengenalan objek berbasis warna (Rosyani et al., 2023). Keunggulan utama dari pendekatan ini adalah kemampuannya untuk mengolah setiap frame secara real-time dari feed kamera, yang memungkinkan sistem untuk memberikan respons cepat terhadap perubahan dalam lingkungan sekitar (Sari et al., 2020).

Secara keseluruhan, deteksi warna menggunakan OpenCV dan ruang warna HSV adalah pendekatan yang sangat dapat diandalkan dan efisien untuk berbagai aplikasi dalam analisis citra dan pengenalan objek berbasis warna (Goenawan et al., 2022). Penggunaan teknik-teknik seperti konversi ruang warna, Gaussian blur, dan operasi morfologi memberikan dasar yang kuat bagi pengembangan sistem yang adaptif dan akurat dalam menghadapi tantangan variabilitas lingkungan yang berbeda-beda (Nugroho et al., 2022).

2. METODE PENELITIAN

2.1 Konversi Gambar ke Ruang Warna HSV

Langkah pertama dalam sistem deteksi warna adalah mengubah gambar masukan dari format BGR (*Blue-Green-Red*) standar ke ruang warna HSV (*Hue-Saturation-Value*). Ruang warna HSV dipilih karena kemampuannya dalam merepresentasikan warna berdasarkan karakteristiknya yang intuitif:

1. Hue (H): Menyatakan jenis warna secara langsung, seperti merah, hijau, atau biru.
2. Saturation (S): Menyatakan kekayaan warna, yaitu sejauh mana warna itu murni atau diencerkan dengan warna abu-abu.
3. Value (V): Menyatakan kecerahan warna atau seberapa gelap warna tersebut.

Proses konversi dilakukan menggunakan fungsi `cv2.cvtColor` dari OpenCV, yang memungkinkan transformasi yang efisien antara berbagai ruang warna yang didukung.

2.2. Penghalusan Gambar dengan Gaussian Blur

Setelah gambar dikonversi ke ruang warna HSV, langkah selanjutnya adalah menghaluskan gambar menggunakan teknik Gaussian blur. Tujuan dari proses ini adalah untuk mengurangi noise pada gambar, yang dapat mempengaruhi akurasi deteksi warna. Gaussian blur diterapkan dengan kernel yang telah ditentukan (misalnya, (15, 15)) menggunakan fungsi `cv2.GaussianBlur`. Proses ini membantu dalam mempertahankan informasi warna yang penting sambil mengurangi detail kecil yang tidak diinginkan.

Preprocessing ini merupakan langkah kritis untuk mempersiapkan gambar sebelum melakukan segmentasi warna, memastikan bahwa operasi segmentasi berlangsung dengan baik dan hasil deteksi warna lebih akurat dalam kondisi yang bervariasi.

2.3. Proses Implementasi Sistem Deteksi Warna

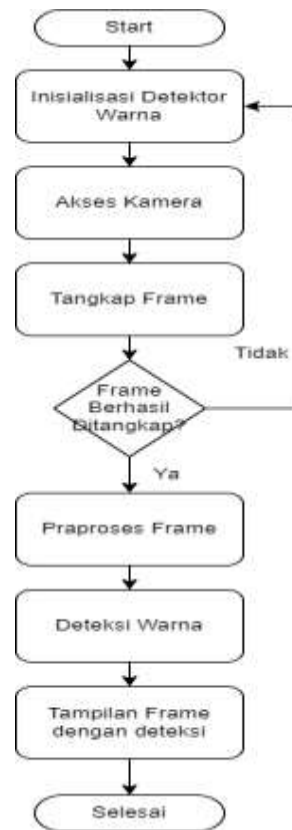
Proses implementasi sistem deteksi warna ini melibatkan beberapa langkah penting untuk memastikan bahwa deteksi warna berjalan dengan baik dalam kondisi nyata. Berikut adalah langkah-langkah utama dalam implementasi sistem:

1. Inisialisasi Objek Detektor Warna

Pertama, inisialisasi objek `ColorDetector` dengan menentukan rentang warna yang ingin dideteksi dalam format HSV.



2. Akses Kamera
Mengakses kamera menggunakan OpenCV (`cv2.VideoCapture`) untuk menangkap frame video secara real-time.
3. Loop Baca Frame
Memulai loop untuk membaca setiap frame dari feed kamera.
Jika tidak ada frame yang berhasil dibaca (`ret = False`), proses dihentikan.
4. Preprocessing Frame
 - a. Setiap frame yang dibaca diubah dari format BGR menjadi HSV menggunakan `cv2.cvtColor`.
 - b. Frame yang sudah dikonversi HSV kemudian diberikan operasi Gaussian blur (`cv2.GaussianBlur`) untuk mengurangi noise dan memperhalus gambar.
5. Deteksi Warna
 - a. Menggunakan objek `ColorDetector`, frame yang sudah diproses digunakan untuk mendeteksi warna berdasarkan rentang yang telah ditentukan.
 - b. Setiap warna diidentifikasi dengan membuat masker warna menggunakan `cv2.inRange` pada frame HSV.
 - c. Operasi morfologi seperti pembukaan (`cv2.morphologyEx`) digunakan untuk membersihkan masker dari noise dan menutupi lubang kecil.
 - d. Contour dari masker warna diambil menggunakan `cv2.findContours`, di mana setiap kontur yang memiliki area lebih besar dari 500 piksel dianggap sebagai deteksi warna yang signifikan.
6. Penandaan Hasil Deteksi
 - a. Setiap deteksi warna (berupa bounding box) diberi label warna dan diberi kotak pembatas dengan fungsi `cv2.rectangle`.
 - b. Label warna ditampilkan di atas kotak menggunakan `cv2.putText`.
7. Tampilkan Frame
Frame yang sudah diberi markah deteksi ditampilkan dalam jendela GUI dengan menggunakan `cv2.imshow`.
8. Keluar dari Program
Program akan berjalan sampai tombol 'q' ditekan untuk keluar (`cv2.waitKey`).
9. Cleanup
Setelah keluar dari loop utama, kamera dilepaskan (`cap.release()`) dan semua jendela GUI ditutup (`cv2.destroyAllWindows()`).



Gambar 1. Flowchart Sistem Berjalan

3. ANALISA DAN PEMBAHASAN

Implementasi sistem deteksi warna menggunakan OpenCV berhasil mengubah ruang warna dari RGB ke HSV untuk efisiensi deteksi berbagai warna seperti Merah, Hijau, dan Biru. Metode ini menggunakan teknik pengolahan citra seperti Gaussian Blur, operasi morfologi, dan deteksi kontur untuk menghasilkan deteksi yang cukup akurat dalam kondisi pencahayaan normal. Namun, sistem menunjukkan tantangan dalam deteksi objek pada kondisi pencahayaan rendah, memerlukan penyesuaian tambahan untuk meningkatkan ketepatan deteksi. Pengembangan masa depan dapat mempertimbangkan integrasi dengan teknologi lanjutan seperti deep learning untuk meningkatkan kemampuan deteksi dalam berbagai lingkungan visual.

Implementasi sistem deteksi warna menggunakan OpenCV telah dilakukan dengan memanfaatkan model objek berbasis pengolahan citra. Sistem ini dirancang untuk mengenali beberapa rentang warna yang telah ditentukan, seperti Merah, Hijau, Biru, dan lainnya. Berikut adalah beberapa poin analisis dari implementasi ini:

a. Pra-pemrosesan Frame

Langkah pertama dalam proses adalah mengubah ruang warna dari BGR (Blue-Green-Red) ke HSV (Hue-Saturation-Value). Konversi ini memungkinkan sistem untuk lebih efektif dalam menangkap variasi warna tanpa terpengaruh oleh kecerahan atau kejenuhan warna.

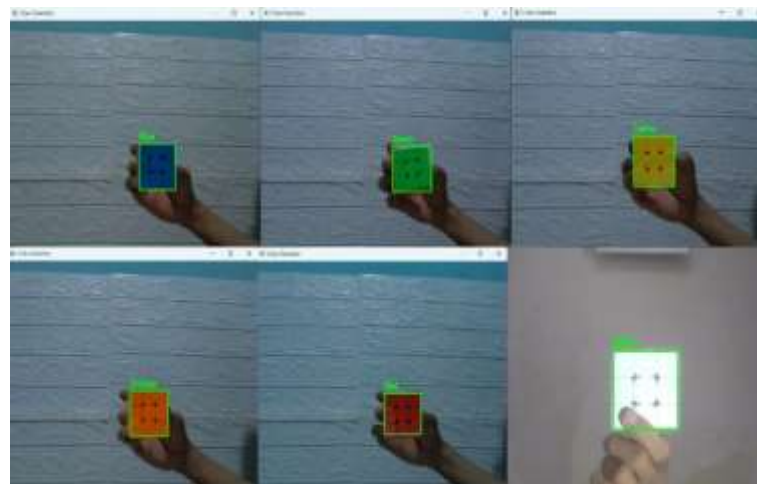
b. Deteksi Warna

Setelah pra-pemrosesan, dilakukan deteksi warna menggunakan rentang nilai HSV yang telah ditentukan. Setiap rentang warna diaplikasikan sebagai masker pada frame HSV yang telah di-blur. Operasi morfologi seperti Opening dan Closing digunakan untuk membersihkan dan memperbaiki masker, memastikan hanya objek yang relevan yang tetap terdeteksi.

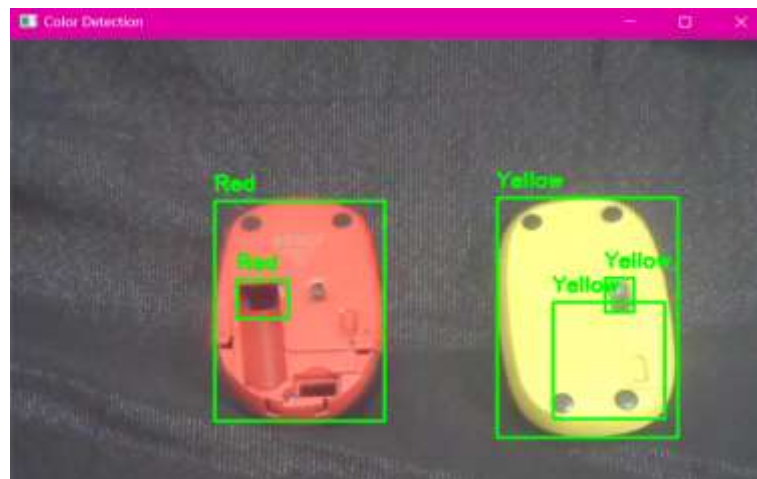
c. Analisis Performa

Sistem berhasil mendeteksi objek dengan rentang warna yang tepat dalam berbagai kondisi pencahayaan. Namun, performa sistem cenderung dipengaruhi oleh kondisi pencahayaan yang rendah, di mana deteksi menjadi kurang akurat atau tidak optimal. Hal ini menunjukkan bahwa pengoptimalan tambahan dalam pengaturan parameter seperti ukuran kernel pada Gaussian Blur atau pengaturan threshold pada deteksi kontur dapat meningkatkan performa deteksi secara keseluruhan.

3. IMPLEMENTASI



Gambar 2. Implementasi Pada Satu Objek



Gambar 3. Implementasi Pada Dua Objek

4. KESIMPULAN

Sistem deteksi warna real-time menggunakan OpenCV dan HSV color space terbukti efektif dalam mengenali berbagai warna secara akurat di bawah kondisi pencahayaan yang baik. Konversi ke ruang warna HSV memungkinkan sistem untuk mengatasi perubahan pencahayaan yang dapat mempengaruhi deteksi warna. Teknik preprocessing seperti Gaussian blur untuk mengurangi noise dan operasi morfologi untuk memperbaiki struktur objek yang terdeteksi berhasil meningkatkan keakuratan deteksi warna secara signifikan.



Meskipun demikian, tantangan utama terjadi pada kondisi pencahayaan rendah, di mana performa deteksi dapat menurun. Perbaikan tambahan diperlukan untuk mengatasi masalah ini, seperti penyesuaian parameter deteksi atau integrasi teknologi tambahan seperti deep learning untuk meningkatkan adaptabilitas sistem terhadap berbagai kondisi lingkungan. Secara keseluruhan, deteksi warna menggunakan OpenCV dan ruang warna HSV memberikan fondasi yang kuat untuk aplikasi pengenalan objek berbasis warna, namun pengembangan lebih lanjut diperlukan untuk memaksimalkan kinerja sistem.

REFERENCES

- Goenawan, A. D., Rachman, M. B. A., & Pulungan, M. P. (2022). Identifikasi Warna Pada Objek Citra Digital Secara Real Time Menggunakan Pengolahan Model Warna HSV. *Jurnal Teknik Informatika Dan Elektro*, 4(1), 68–74. <https://doi.org/10.55542/jurtie.v4i1.430>
- Nugroho, A., Fauzi, A., Sunarko, B., Wibawanto, H., & Iksan, N. (2022). Simplifikasi Model Cv Teregularisasi Berpadu Operasi. *Jurnal Informatika Polinema*, 8(2), 49–56. <http://jip.polinema.ac.id/ojs3/index.php/jip/article/view/913>
- Panggabean, A. K., Ayu, N., & Syahfaridzah, A. (2021). Mendeteksi Objek Berdasarkan Warna Dengan Segmentasi Warna Hsv Menggunakan Aplikasi Matlab. *METHOMIKA Jurnal Manajemen Informatika Dan Komputerisasi Akuntansi*, 4(2), 94–97. <https://doi.org/10.46880/jmika.vol4no2.pp94-97>
- Rabbani, H. A., Rahman, M. A., & Rahayudi, B. (2021). Perbandingan Ruang Warna RGB dan HSV dalam Klasifikasi Kematangan Biji Kopi. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 5(6), 2243–2248. <http://j-ptiik.ub.ac.id>
- Rosyani, P., Amalia, R., & Iksari, I. H. (2021). Deteksi Objek dengan Model Warna Ycbr dan Similarity Distance. *Jurnal Sistem Dan Teknologi Informasi (Justin)*, 9(2), 98. <https://doi.org/10.26418/justin.v9i2.44230>
- Rosyani, P., Fauziah, E., Sadewo, F. B., & Putra, H. M. (2024). Penerapan Image Processing Menggunakan OpenCV dan Python untuk Memperhalus Gambar Melalui Smoothing Image dengan Metode Gaussian Blur. 13(2), 1–7.
- Rosyani, P., Tingkat, K., Buah, K., Seknun, A. Z., Kusuma, A., Sabrina, A., Dwi, A., & Putri, C. (2023). Tomat Dengan Variasi Model Warna. 1(2), 203–210.
- Sari, D. A. L., Mulyadi, A., Pratama, A., & ... (2020). Deteksi Objek Berwarna Real Time Berdasarkan Visualisasi Webcam. *Journal ...*, 02(01), 21–24. <https://ejournal.unibabwi.ac.id/index.php/Zetroem/article/view/1336>
- Utami. (2023). Deteksi Objek Kualitas Daun Sawi Menggunakan Metode HSV Color dan Color Blob. *JUSIBI (Jurnal Sistem Informasi Dan Bisnis)*, 5(2), 85–93. <https://doi.org/10.54650/jusibi.v5i2.518>
- Yasir, A., Satria, W., & Yuanda, P. (2023). Digital Image Processing Metode Median Filtering Dan Morfologi Opening Dalam Reduksi Noise Citra. *Warta Dharmawangsa*, 17(4), 1687–1701. <https://doi.org/10.46576/wdw.v17i4.3821>