

# PENDETEKSIAN OBJEK MENGGUNAKAN ALGORITMA HSV BERBASIS RASPBERRY PI 3B

Taufik Rahmadani<sup>\*)</sup>, Wisnu Dyan Nugroho<sup>\*)</sup>, Atha' Dwira Perdana<sup>\*)</sup>, Aris Triwiyatno<sup>\*)</sup>

Departemen Teknik Elektro, Universitas Diponegoro  
Jl. Prof. Sudharto, SH, Kampus UNDIP Tembalang, Semarang 50275, Indonesia

<sup>\*)E-mail:</sup> rahmadani.taufik@gmail.com, wisnudyan@gmail.com, athadp@hotmail.com<sup>1)</sup>, aristriwiyatno@undip.ac.id<sup>2)</sup>

## Abstrak

Teknologi dengan menggunakan metode pengolahan citra digital (*digital image processing*) marak dikembangkan belakangan ini, hal ini disebabkan kemudahan dalam penggunaannya serta didukung dengan kemampuan komputer yang memadai untuk melakukan komputasi. Pada penelitian ini, kami membuat alat pendeteksi objek menggunakan algoritma HSV (*Hue, Saturation, Value*) dengan menggunakan Raspberry PI 3B sebagai pengolah data. Dari penelitian yang kami lakukan, didapatkan bahwasannya metode HSV dapat mendeteksi dan *tracking* objek dalam jarak berapapun, selama masih terlihat jelas oleh kamera, sehingga resolusi atau kualitas kamera berperan penting dalam pengolahan citra digital menggunakan metode HSV ini.

*Kata Kunci: pengolahan citra digital, algoritma HSV, Raspberry PI 3B, deteksi, tracking objek*

## Abstract

*Many technologies using digital image processing (digital image processing) developed lately, this is due to ease of use and supported with adequate computer skills to perform computing. In this research, we made an object detector using HSV algorithm using Raspberry PI 3B as a data processor. From our research, it was found that HSV method can detect and track objects in any distance, as long as it is still visible to the camera, so the resolution or quality of the camera plays an important role in digital image processing using this HSV method.*

*Keywords: digital image processing, HSV algorithm, Raspberry PI 3B, detection, object tracking*

## 1. Pendahuluan

Belakangan ini teknologi pendeteksian objek banyak dikembangkan, mengingat pentingnya akan kebutuhan hal tersebut dalam kehidupan untuk menganalisa dan mengamati lingkungan sekitar. Berbagai cara digunakan untuk melakukan pendeteksian, beberapa cara di antaranya adalah dengan sensor magnetik, gelombang, dan pemrosesan gambar bergerak. Belakangan, metode yang banyak diteliti adalah metode untuk pemrosesan citra bergerak. Hal ini disebabkan oleh meningkatnya kekuatan komputasi [1]-[3]

Pada penelitian ini, kami menggunakan metode pengolahan citra digital (*digital image processing*). Pengolahan citra digital sendiri merupakan teknik manipulasi citra secara digital yang khususnya menggunakan komputer menjadi citra lain yang sesuai dengan kebutuhan. Agar mudah diinterpretasi oleh manusia atau komputer, pengolahan citra harus dilakukan dengan berbagai macam metode untuk mencapai citra sesuai yang diinginkan. Operasi pengolahan citra digital umumnya dilakukan dengan tujuan memperbaiki kualitas suatu

gambar sehingga dapat dengan mudah diinterpretasikan oleh mata manusia dan untuk mengolah informasi yang ada pada suatu gambar untuk kebutuhan identifikasi objek secara otomatis [4]-[5]. Berikut proses pengolahan citra digital yang digunakan dalam penelitian ini :

### a. Thresholding

*Thresholding* merupakan konversi citra berwarna ke citra biner yang dilakukan dengan cara mengelompokkan nilai derajat keabuan setiap *pixel* kedalam 2 kelas, hitam dan putih. Pada citra hitam putih terdapat 256 level, artinya mempunyai skala "0" sampai "255" atau [0,255], dalam hal ini nilai intensitas 0 menyatakan hitam, dan nilai intensitas 255 menyatakan putih, dan nilai antara 0 sampai 255 menyatakan warna keabuan yang terletak antara hitam dan putih [4].

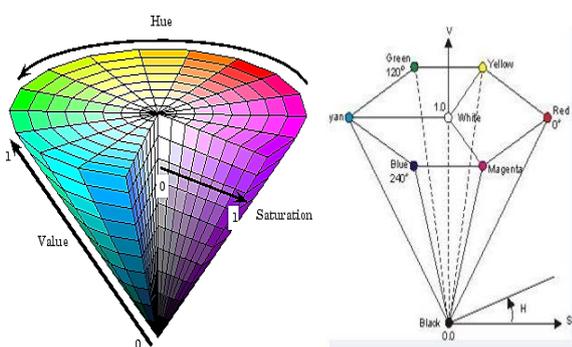
### b. Color Filtering

*Color Filtering* adalah suatu teknik pengolahan citra yang yang dipakai untuk memanipulasi suatu citra berdasarkan warna spesifik. Cara kerjanya adalah dengan membandingkan komponen warna setiap *pixel* citra dengan warna spesifik. Apabila warnanya sesuai dengan

warna spesifik komponen warna pixel tersebut dibiarkan saja. Namun, bila warnanya tidak sesuai dengan warna spesifik maka komponen warna pixel tersebut diubah menjadi warna background, biasanya menjadi warna hitam [4].

Warna yang digunakan dalam penapisan warna dapat direpresentasikan dalam berbagai ruang warna. Ada beberapa ruang warna yang dikenal, antara lain *RGB* (*Red, Green, Blue*), *HSV* (*Hue, Saturation, Value*), *YCbCr*, dsb. *HSV* merupakan ruang warna yang sangat cocok untuk mengidentifikasi warna-warna dasar, dimana warna dasar ini digunakan dalam penelitian sebagai warna identifikasi cahaya. Selain itu, *HSV* menoleransi terhadap perubahan intensitas cahaya. Inilah yang menjadi keunggulan *HSV* dibandingkan dengan ruang warna lainnya [5].

Ruang warna pada metode *HSV* memiliki komposisi perpaduan dari warna dasar *red, green, blue* seperti pada Gambar 1 dan diatur pada komponen *hue* yang direpresentasikan terhadap grafik roda melingkar 360°. Pada setiap komposisi warna dasar dari pengaturan *hue*, diatur pula tingkat kemurnian warna pada pengaturan *saturation*. Semakin membuat kontras warna, semakin rendah nilai *value* maka akan semakin tinggi *saturation*, maka semakin murni pemilihan warna dari komposisi *hue*. Jika nilai *saturation* semakin kecil maka warna akan memudar menuju putih. Pengaturan lainnya pada ruang warna *HSV* yaitu *value* yang gelap hingga berwarna hitam. Namun jika semakin besar nilai *value*, maka akan semakin terang dan menuju pada warna putih [5]. Ruang warna metode *HSV* ditunjukkan pada Gambar 1.



Gambar 1. Ruang warna metode HSV [5]

## 2. Metode

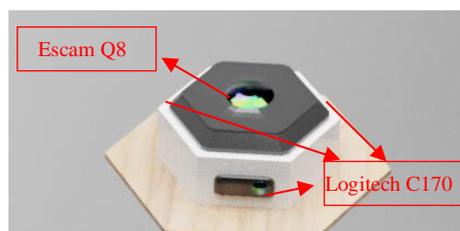
Perancangan sistem alat terdiri atas perancangan hardware dan software.

### 2.1. Perancangan Hardware

Dalam perancangan hardware kami menggunakan 3 buah kamera Logitech C170 dan satu buah kamera *panoramic* Escam Q8, yang masing- masing kamera memiliki

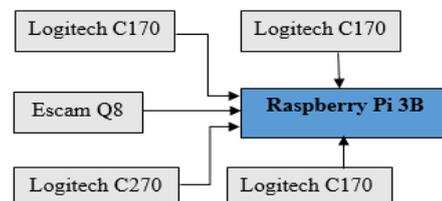
fungsinya tersendiri, serta satu pengolahan data kami menggunakan satu buah Raspberry PI 3B.

Tiga kamera Logitech C170 yang berada di bagaian sisi berfungsi untuk mendeteksi adanya suatu objek pada sekitar alat dengan algoritma *HSV*. Kamera *panoramic* Escam Q8 sendiri yang berada di atas alat digunakan untuk melihat keadaan lingkungan sekeliling, dikarenakan kamera tersebut dapat melihat/ memantau keadaan sekitar seluas 180°. Perancangan kamera seperti ditunjukkan pada Gambar 2.



Gambar 2. Posisi kamera

Pada penelitian ini digunakan satu buah Raspberry PI 3B, mengingat pada penelitian ini memerlukan proses komputasi yang sangat cepat, sehingga tidak memungkinkan apabila menggunakan *controller* / IC (*Integrated Circuit*) biasa. Hubungan antrara kamera dan Raspberry PI 3B ditunjukan pada Gambar 3.



Gambar 3. Hubungan kamera dan Raspberry Pi 3B

### 2.2. Perancangan Software

Pada perancangan software kami menggunakan Python 2.7 serta OpenCV. OpenCV adalah sebuah *API* (*Application Programming Interface*) *library* yang sangat familiar pada pengolahan citra menggunakan *computer vision*. *Computer vision* adalah salah satu cabang dari bidang ilmu pengolahan citra (*Image Processing*) yang memungkinkan komputer dapat melihat seperti manusia.

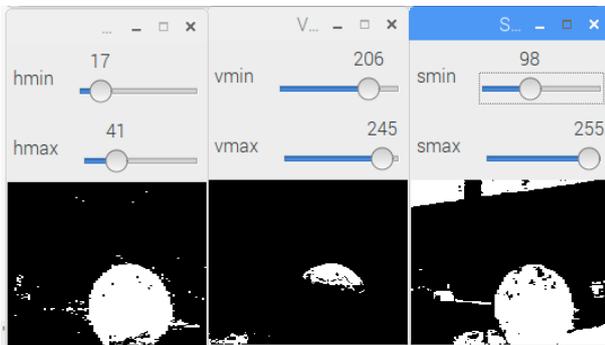
Dengan *vision* tersebut komputer dapat mengambil keputusan, melakukan aksi, dan mengenali terhadap suatu objek. Beberapa implementasi dari *computer vision* adalah pengenalan wajah, deteksi wajah, deteksi objek, deteksi jalan, dan lain-lain. OpenCV adalah *library Open Source* untuk *computer vision* untuk C/C++. OpenCV didisain Untuk aplikasi *real-time*, memiliki fungsi-fungsi akuisisi yang baik untuk mengolah citra atau video.

### 3. Hasil dan Analisis

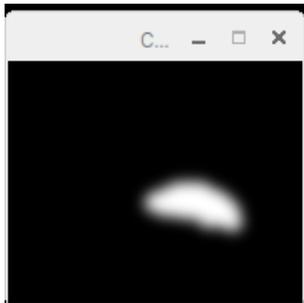
Hasil penelitian ini diperoleh melalui serangkaian pengujian yaitu pendeteksian dan *tracking*. Pada proses pendeteksian nilai H (*hue*), S (*saturation*), dan V (*value*) diatur terlebih dahulu sampai objek yang diinginkan terdeteksi. Sedangkan pada proses *tracking* jarak objek yang sudah dideteksi divariasikan sedemikian rupa.

#### 3.1. Pendeteksian

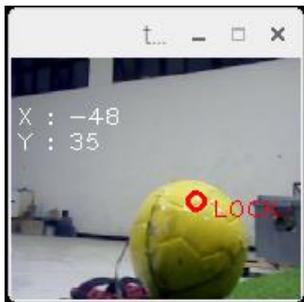
Dari penentuan nilai HSV, didapatkan nilai H sebesar  $H_{min} = 17$  dan  $H_{max} = 41$ , S sebesar  $S_{min} = 98$  dan  $S_{max} = 255$ , dan V sebesar  $V_{min} = 206$  dan  $V_{max} = 245$ , agar alat dapat mendeteksi objek yang diinginkan. Pada penelitian ini kami menggunakan bola berwarna kuning sebagai objek.



Gambar 4. Penentuan Nilai H, S, dan V



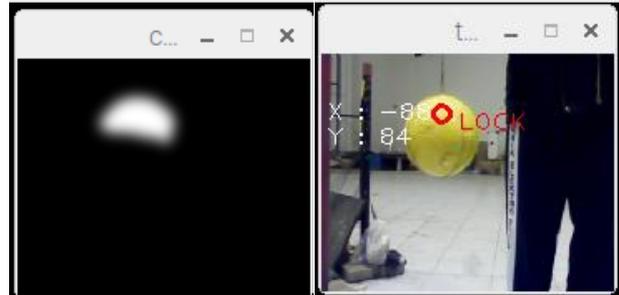
Gambar 5. Hasil Pemfilteran



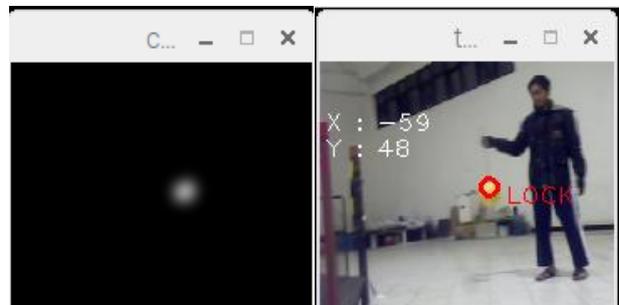
Gambar 6. Objek terdeteksi

#### 3.2. Tracking

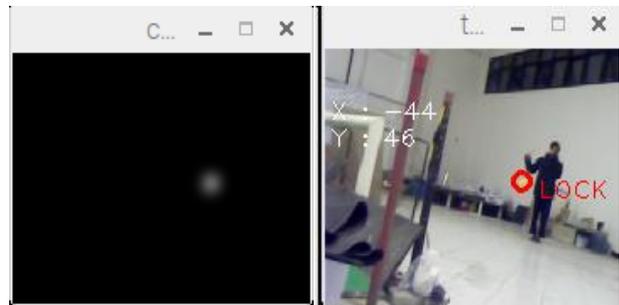
Pada proses ini jarak objek dari alat divariasikan, kami memvariasikan sebesar 1 m, 5 m, 10 m, dan 15 m.



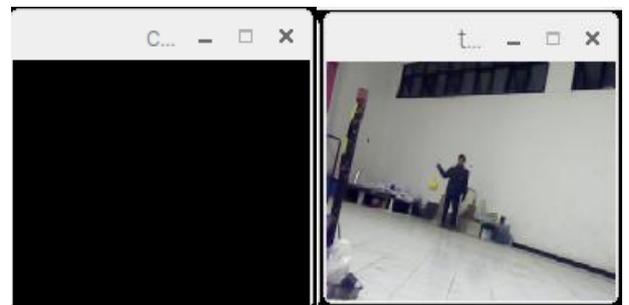
Gambar 7. Jarak objek 1 m



Gambar 8. Jarak objek 5 m



Gambar 9. Jarak objek 10 m



Gambar 10. Jarak objek 15 m

Tabel 1. Status Objek

No	Jarak	Tracking
1	1 m	Ya
2	5 m	Ya
3	10 m	Ya
4	15 m	Tidak

#### 4. Kesimpulan

Telah berhasil dirancang alat pendeteksian objek menggunakan algoritma *HSV* berbasis Raspberry Pi 3B, hasil yang didapat alat dapat mendeteksi dan *tracking* objek sampai jarak 10 m, sedangkan untuk jarak 15 m alat tidak mendeteksi objek, hal ini dipengaruhi resolusi kamera dan pencahayaan lingkungan di sekitar objek. Pada penelitian digunakan objek berwarna kuning dengan nilai *HSV* sebesar  $H$  sebesar  $H_{min} = 17$  dan  $H_{max} = 41$ ,  $S$  sebesar  $S_{min} = 98$  dan  $S_{max} = 255$ , dan  $V$  sebesar  $V_{min} = 206$  dan  $V_{max} = 244$ .

#### Referensi

- [1] N. Buch, S. A. Velastin, and J. Orwell, "A Review of Computer Vision Techniques for the Analysis of Urban Traffic," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, pp. 920–939, 2011.
- [2] V. Kastrinaki, M. Zervakis, and K. Kalaitzakis, "A survey of video processing techniques for traffic applications," *Image Vis. Comput.*, vol. 21, pp. 359–381, 2003.
- [3] B. D. Stewart, I. Reading, M. S. Thomson, T. D. Binnie, K. W. Dickinson, and C. I. Wan, "Adaptive lane finding in road traffic image analysis," in *Seventh International Conference on Road Traffic Monitoring and Control*, 1994.
- [4] D. Putra, *Pengolahan Citra Digital*. Yogyakarta: Andi, 2010.
- [5] V. Bharadwaj, "Colours : A Scientific Approach," in *Composition of Colours*, 2014, pp. 1–6.