

# APLIKASI PENGENAL WARNA UNTUK PENDERITA BUTA WARNA MENGGUNAKAN PENGOLAHAN MODEL WARNA HSV

## (APPLICATION OF COLOR RECOGNITION FOR COLOR BLIND PATIENTS USING HSV COLOR MODEL PROCESSING)

A.Nur Mufidah Nayif<sup>1</sup>, Rita Magdalena<sup>2</sup>, Eko Susatio<sup>3</sup>

<sup>1,2,3</sup>Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

<sup>1</sup>fidahnayif@telkomuniversity.ac.id, <sup>2</sup>ritamagdalenam@telkomuniversity.ac.id, <sup>3</sup>maharusdi@gmail.com

### Abstrak

Buta warna adalah ketidakmampuan seseorang untuk membedakan beberapa warna yang dapat dibedakan oleh orang lain. Mereka akan sulit membedakan warna tertentu (buta warna sebagian) atau bahkan seluruh warna (buta warna total). Maka dari itu, Tugas Akhir ini bertujuan untuk menghadirkan solusi dimana para penderita buta warna bisa mengetahui warna secara pasti dengan menggunakan aplikasi pengenalan warna yang dapat diakses dengan mudah bagi seluruh pengguna *smartphone*.

Dalam penelitian ini dibuat sebuah sistem yang dapat membedakan enam warna solid yaitu merah, hijau, biru, jingga, kuning, dan ungu dengan metode *thresholding*. Cara kerja sistem adalah dengan mendigitasi objek warna lalu melakukan klasifikasi warna berdasarkan rentang nilai warna HSV yang telah dimasukkan ke sistem. Setelah itu, dilakukan proses *thresholding* sehingga objek warna dapat dikenali. Algoritma tersebut diaplikasikan ke dalam sistem Android dan menggunakan model warna HSV.

Keluaran dari aplikasi ini berupa nama dari warna objek yang berhasil dikenali. Hasil pengenalan warna dalam bentuk teks dan ditampilkan pada layar *smartphone* Android. Dari penelitian ini diperoleh hasil pengenalan warna terbaik yaitu pada malam hari di dalam ruangan dengan intensitas cahaya 34 lumen. Performansi yang diperoleh sebesar 93,1% menggunakan kertas HVS dan 83,57% menggunakan kertas *glossy*

**Kata Kunci:** buta warna, pengenalan warna, HSV, android, *thresholding*

### Abstract

*Color blindness is the inability of a person to distinguish several colors that can be distinguished by others. They will be difficult to distinguish certain colors (partial color blindness) or even whole color (total color blindness). Therefore, this Final Project aims to present a solution where people with color blindness can find out colors for sure by using a color recognition application that can be easily accessed for all smartphone users. In this research, a system that can distinguish six solid colors, namely red, green, blue, orange, yellow, and purple, uses the thresholding method. The way the system works is to digitize color objects and then color classification based on the range of HSV color values that have been entered into the system. After that, a thresholding process is carried out so that color objects can be recognized. The algorithm is applied to the Android system and uses the HSV color model.*

*The output of this application is the name of the color of the object that was identified. The color recognition results are in text form and are displayed on the screen of an Android smartphone. From this study the best color recognition results are obtained at night in the room with 34 lumens of light intensity. The performance obtained was 93.1% using HVS paper and 83.57% using glossy paper.*

**Keywords:** *color blindness, color recognition, HSV, android, thresholding.*

### 1. Pendahuluan

Buta warna atau *color blindness* adalah ketidakmampuan seseorang untuk membedakan beberapa warna yang dapat dibedakan oleh orang lain [1]. Hal ini sebagian besar disebabkan oleh faktor genetik [1]. Penderita buta warna akan sulit membedakan warna tertentu (buta warna sebagian) atau bahkan seluruh warna (buta warna total). Penderitanya didominasi oleh kaum laki-laki yang secara umum persentasenya di Indonesia berdasarkan keluhan penderita adalah 0,7% [1]. Orang dengan kelainan buta warna terkadang mengalami kesulitan untuk membedakan berbagai warna di lingkungannya, namun ia masih bisa melakukan kegiatan normal seperti orang-orang pada umumnya.

Di samping itu, perkembangan teknologi dan ilmu pengetahuan yang semakin canggih mendorong penulis untuk menghadirkan solusi dimana para penderita bisa mengetahui warna secara pasti dengan menggunakan aplikasi pengenalan warna yang dapat diakses dengan mudah bagi seluruh pengguna *smartphone*.

Meninjau dari Penelitian yang dilakukan oleh Vera Nita.M , Robert, dan Gasim mengenai penerapan model warna HSV pada aplikasi deteksi warna[2]. Selain itu Penelitian oleh R. D. Kusumanto, Alan Novi. T, dan

Wahyu Setyo yang membahas tentang pengenalan warna menggunakan model HSV dan *software visual studio* 2008 yang dilengkapi dengan program penunjang EmguCV untuk menentukan jenis warna secara *real time* [3].

Pada Tugas Akhir ini akan dibuat aplikasi pengenal warna dengan metode klasifikasi warna menggunakan model warna *Hue, Saturation* dan *Value* (HSV). Pada penelitian-penelitian sebelumnya masukannya hanya satu warna, sementara pada penelitian ini diberikan masukan beberapa warna sekaligus untuk dikenali.

Berdasarkan penelitian sebelumnya, pada Tugas Akhir ini penulis akan tetap memanfaatkan model warna HSV. Pengembangan yang dilakukan adalah membuat aplikasi pengenal warna berbasis Android yang dapat mengenali enam kombinasi warna solid pada suatu objek secara *real time*. Aplikasi ini juga dirancang dengan mempertimbangkan warna yang sulit dikenali oleh penderita buta warna, sehingga dalam perancangan aplikasi ini ada beberapa warna yang tidak digunakan. Warna-warna yang digunakan pada penelitian ini adalah merah, hijau, biru, jingga, kuning, dan ungu.

## 2. Dasar Teori

### 2.1 Buta Warna

Buta warna adalah ketidakmampuan seseorang untuk membedakan beberapa atau bahkan seluruh warna yang dapat dibedakan oleh orang lain. Kelainan mata ini umumnya dianggap lebih banyak terdapat pada laki-laki dibanding perempuan dengan perbandingan 20:1 (Kartika, 2014: 269). Buta warna merupakan penyakit seumur hidup, namun penderitanya dapat berlatih untuk menyesuaikan kondisi ini dalam kehidupan sehari-hari [1].

Buta warna terdiri dari tiga jenis yaitu buta warna merah-hijau, buta warna biru-kuning, dan buta warna total (akromatopsia) [4]. Buta warna merah-hijau dan biru-kuning biasa disebut buta warna parsial.

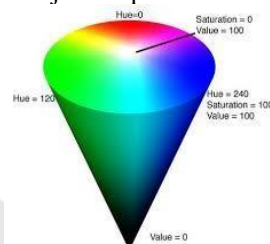
### 2.2 Pengolahan Citra

Pengolahan citra (*Image Processing*) adalah suatu bentuk pengolahan sinyal dengan masukan berupa gambar dan ditransformasikan menjadi gambar lain sebagai keluarannya menggunakan teknik tertentu [5]. Fungsi utama *Image Processing* adalah meningkatkan kualitas gambar agar terlihat lebih jelas. Hal ini ditegaskan oleh Munir (2014) bahwa masukan dari proses pengolahan citra adalah citra dan keluarannya juga citra yang berkualitas lebih baik dari pada citra masukannya. Selain itu, *Image Processing* juga digunakan untuk memproses data yang diperoleh dalam persepsi mesin [5].

Citra sendiri merupakan gambar dua dimensi yang dihasilkan dari gambar analog dua dimensi yang secara kontinu menjadi gambar diskrit melalui proses *sampling*. Gambar analog dibagi menjadi N baris dan M kolom sehingga menjadi gambar diskrit. Persilangan antara baris dan kolom tertentu disebut dengan piksel[5].

### 2.3 HSV (*Hue, Saturation Value*)

Model warna HSV adalah salah satu model pengolahan warna gambar yang mendefinisikan warna dalam terminologi *Hue, Saturation*, dan *Value* seperti ditunjukkan pada Gambar 1.



Gambar 1. Model Warna HSV [5].

*Hue* menyatakan tingkat kemurnian warna dengan jangkauan nilai 0 – 360 derajat. *Hue* digunakan untuk membedakan berbagai warna serta menentukan tingkat kemerahan, kehijauan, dsb, dari cahaya. *Saturation* (kadang disebut *chroma*) menyatakan tingkat kejenuhan warna dan mengindikasikan seberapa banyak warna putih yang diberikan pada warna. *Saturation* memiliki nilai 0 – 100%. *Value* merupakan tingkat kecerahan warna yang memiliki nilai 0 – 100%. Tabel 2 menyediakan komposisi nilai warna masing-masing komponen *Hue, Saturation*, dan *Value*.

Tabel 2. Nilai Warna HSV [2].

No	Nama Warna	Hue (°)	Saturation (%)	Value (%)
1.	Putih	0	0	0
2.	Hitam	0	0	100
3.	Merah	0	100	100
4.	Kuning	60	100	100
5.	Hijau	120	100	100
6.	Biru	240	100	100
7.	Ungu	300	100	100

Suatu warna dengan nilai *value* 100% akan terlihat sangat cerah. Sebaliknya, suatu warna dengan nilai *value* 0 akan tampak sangat gelap. Ruang warna HSV berbasis pada *cylindrical coordinates*. Ruang warna HSV memiliki rentang warna 0 sampai dengan 1 yang dapat diperoleh dari transformasi RGB dengan menggunakan rumus transformasi nonlinier seperti ditunjukkan pada persamaan (1) sampai (4) [6]

$$H = \begin{cases} \theta & \text{jika } b \leq g \\ 360^\circ & \text{jika } b > g \end{cases} \quad (1)$$

dengan,

$$\theta = \cos^{-1} \frac{1/2(r-g) + (r-b)}{[(r-g)^2 + (r-b)(g-b)]^{1/2}} \quad (2)$$

$$S = \frac{v - \min(r, g, b)}{v} \quad (3)$$

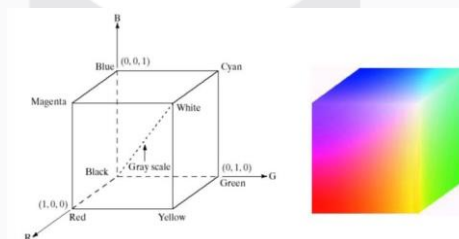
$$v = \max(r, g, b) \quad (4)$$

H = Nilai komponen *Hue*  
V = Nilai komponen *Value*  
S = Nilai komponen *Saturation*  
r = Nilai *red*  
g = Nilai *green*  
b = Nilai *blue*

Model warna HSV biasa digunakan pada proses pengolahan citra dan penelitian-penelitian yang berkaitan dengan pengenalan warna dan objek. Hal ini dikarenakan HSV adalah model warna yang lebih akurat dalam mendeskripsikan sensasi warna oleh mata manusia dibandingkan model warna *Red, Green, Blue* (RGB) [6].

#### 2.4 RGB (*Red, Green, Blue*)

RGB merupakan jenis ruang warna yang mendefinisikan warna berdasarkan tiga kanal warna yaitu merah, hijau, dan biru [7]. Model gambar RGB seperti ditunjukkan pada Gambar 2.2.









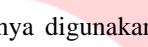


Gambar 2. Model warna RGB.

Ketiga warna tersebut dinamakan warna pokok karena campuran dari ketiga warna itu dapat menghasilkan warna lain [5]. Hal ini ditegaskan oleh Young (1802) yang menyatakan bahwa pencampuran warna-warna pokok C1, C2, dan C3 dengan komposisi tertentu dapat menghasilkan sembarang warna

Bila citra warna didigitasi, maka tiga buah *filter* digunakan untuk mengekstraksi intensitas warna merah, hijau, dan biru, dan bila ketiganya dikombinasikan kita memperoleh persepsi warna. Tabel 3 menunjukkan komposisi warna RGB hasil dari pencampuran warna-warna pokok.

Tabel 3. Komposisi Warna RGB.

Nama	Contoh	Red	Green	Blue
Merah		255	0	0
Hijau		0	255	0
Biru		0	0	255
Kuning		255	255	0
Magenta		255	0	255
Cyan		0	255	255
Putih		255	255	255
Hitam		0	0	0
Abu-abu		128	128	128

Ruang warna ini pada umumnya digunakan untuk menampilkan citra atau gambar dalam perangkat elektronik seperti televisi, komputer, kamera digital, dan pemindai [8]. Pada perhitungan dalam program-program komputer, model warna RGB direpresentasikan dengan nilai masing-masing komponennya yaitu antara 0 hingga 255, sehingga setiap komponen terdiri dari 256 tingkat. Apabila dikombinasikan maka ada  $256 \times 256 \times 256$  atau 16.777.216 kombinasi warna RGB yang dapat dibentuk [5].

## 2.5 Segmentasi Warna

Segmentasi warna merupakan proses memecah suatu citra digital menjadi banyak bagian (segmen) daerah yang tidak saling bertabrakan (*nonoverlapping*) dan daerah hasil segmentasi tersebut merupakan kelompok piksel yang bertetangga atau berhubungan (Castleman, 1996). Proses segmentasi warna dilakukan dengan pendekatan daerah yang bekerja melalui tahapan menganalisis nilai warna dari tiap piksel pada citra lalu membagi citra tersebut sesuai fitur yang diinginkan [9].

Pada penelitian ini, penulis menggunakan segmentasi warna dengan cara mengkonversi ruang warna citra yang semula RGB menjadi HSV dengan cara pendekatan batas. Penentuan nilai toleransi pada setiap dimensi warna HSV dilakukan untuk membentuk segmen sesuai dengan warna yang diinginkan. Tahap selanjutnya adalah proses perhitungan nilai ambang batas (*threshold*) berdasarkan nilai toleransi yang telah ditetapkan. Hasil dari proses *threshold* akan membentuk segmen area dengan warna sesuai toleransi yang diinginkan [9,11].

## 2.6 OpenCV

OpenCV (*Open Source Computer Vision Library*) adalah sebuah *library* yang berisi berbagai fungsi pemrograman untuk teknologi *computer vision* secara *realtime*. *Library* ini ditulis dalam bahasa pemrograman C atau C++ dan dapat beroperasi pada sistem operasi Linux, Windows, iOS *pack*, dan Android.

Salah satu tujuan OpenCV adalah untuk menyediakan infrastruktur *computer vision* yang mudah digunakan sehingga membantu pemakainya membangun aplikasi *vision* yang cukup canggih dengan cepat. OpenCV digunakan secara luas di perusahaan, kelompok peneliti, dan badan pemerintahan.

*Library* openCV memiliki 2500 algoritma yang dioptimalkan, yang mencakup serangkaian komprehensif *computer vision* dan algoritma mesin serta pembelajaran mesin yang canggih. Algoritma ini dapat digunakan untuk mendeteksi dan mengenali wajah, mengidentifikasi objek, mengklasifikasikan tindakan manusia dalam video, dll.

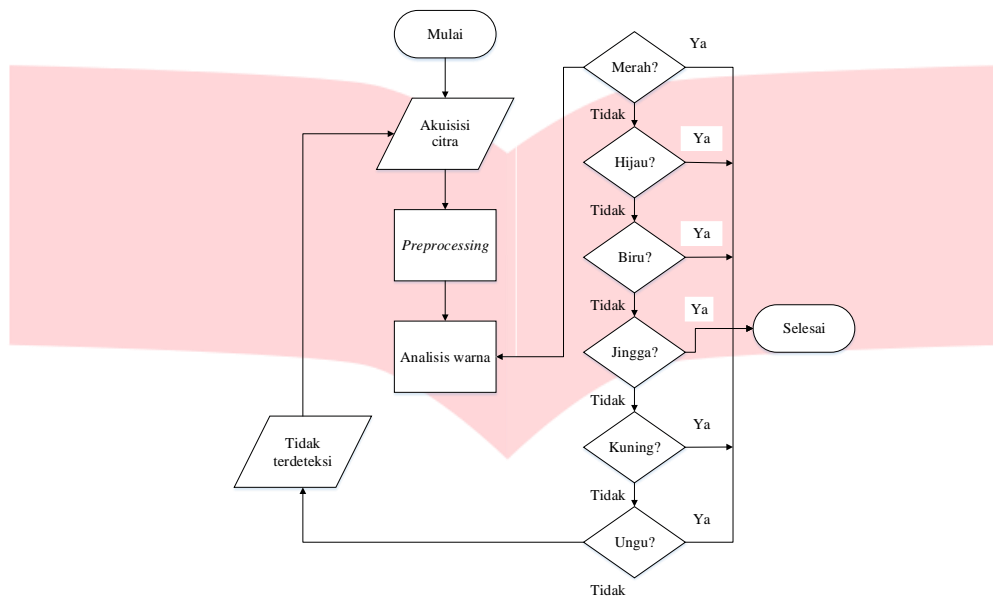
## 2.7 Computer Vision

*Computer Vision* adalah suatu proses transformasi data yang berasal dari kamera berupa video maupun foto ke dalam sebuah presentasi yang baru. Data yang dimasukkan ke dalam kegiatan transformasi tersebut memungkinkan untuk memiliki beberapa informasi yang terkontekstual seperti halnya sebuah foto atau gambar yang di dalamnya terdapat berbagai objek. Dengan demikian akan didapatkan keputusan-keputusan yang akan diambil pada gambar, misalnya berupa “adakah telapak tangan seseorang pada gambar tersebut?” atau “siapa sajakah orang yang terdapat pada foto tersebut?”. Adapun perubahan ke dalam presentasi yang baru seperti perubahan gambar menjadi *grayscale* atau juga pemotongan objek pada gambar.

### 3. Pembahasan

#### 3.1 Diagram Alir Kerja Sistem

Penulis akan membahas lebih detail mengenai cara kerja sistem pengenalan warna untuk penderita buta warna yang ditunjukkan pada Gambar 3.



Gambar 3. Diagram Alir Kerja Sistem.

Berdasarkan diagram alir pada Gambar 3, cara kerja sistem pengenalan warna yang dirancang oleh penulis sebagai berikut:

1. Pengguna membuka aplikasi pengenalan warna yang telah mereka *install* di *smartphone*.
2. Proses pengambilan citra objek dilakukan dengan cara mengarahkan kamera *smartphone* pada objek deteksi, misalnya kertas berwarna, baju, dsb.
3. Aplikasi secara otomatis akan mengambil area untuk mendeteksi warna pada objek.
4. Penentuan nilai ambang batas (*threshold*)

Penentuan batas nilai pada warna atau biasa disebut proses *thresholding* bertujuan untuk menentukan warna yang terdeteksi dengan melihat batas atas tertinggi dan terendah dari nilai komponen warna. Tahapan penentuan batas nilai sebagai berikut:

Langkah 1 : Menentukan titik *sample* lalu mengambil nilai piksel

Langkah 2 : Mengumpulkan nilai piksel dari setiap kondisi

Langkah 3 : Mengambil nilai tertinggi dan terendah dari pengumpulan nilai tersebut.

Dari data yang didapatkan, penulis memperoleh nilai maksimum dan minimum dari setiap kondisi untuk warna merah, hijau, biru, kuning, dan ungu, seperti yang ditunjukkan pada Tabel 4.

Tabel 4. Batas Nilai Warna HSV Aplikasi.

No	Nama Warna	Batasan Nilai		
		Hue (°)	Saturation (%)	Value (%)
1	Merah	0<H<4 && 169<H<181	26<S<100	20<V<100
2	Ungu	124<H<146	39<S<100	39<V<100
3	Biru	99<H<121	39<S<100	39<V<100
4	Hijau	49<H<96	19<S<100	30<V<100
5	Kuning	19<H<31	39<S<100	39<V<100
6	Jingga	9<H<19	15<S<100	75<V<100

5. Warna objek berhasil dikenali dari hasil *thresholding* adalah warna-warna yang memiliki nilai HSV yang terdapat dalam rentang warna pada Tabel 4. Selain dari warna tersebut, maka tidak dikenali oleh sistem. Untuk lebih jelasnya, penulis menyisipkan contoh hasil *thresholding* sistem yang ditunjukkan pada Gambar 4.



Dari enam percobaan yang telah dilakukan, penulis merangkum hasil pengukuran tingkat akurasi aplikasi dalam mengenali warna yang ditunjukkan pada Tabel 5.

**Tabel 5.** Rangkuman Hasil Percobaan

Warna	Kertas HVS			Kertas Glossy			Rata-rata
	Siang outdoor	Siang indoor	Malam indoor	Siang outdoor	Siang indoor	Malam indoor	
Semua	100%	91,67%	91,67%	83,33%	83,33%	91,67%	90,27
Merah	93,33%	100%	93,33%	100%	100%	100%	97,78%
Hijau	86,67%	100%	100%	73,33%	93,33%	86,67%	90%
Biru	86,67%	93,33%	100%	93,33%	93,33%	86,67%	92,22%
Jingga	100%	80%	100%	46,67%	33,33%	46,67%	67,78%
Kuning	100%	93,33%	93,33%	100%	100%	93,33%	96,67%
Ungu	60%	40%	73,33%	86,67%	60%	80%	66,67%
Rata-rata	89,52%	85,48%	93,1%	83,33%	80,47%	83,57%	

Berdasarkan data pada Tabel 5, dapat disimpulkan bahwa dari enam kondisi percobaan, warna merah memiliki rata-rata tingkat akurasi tertinggi yaitu 97,78%. Sementara itu, warna ungu pada posisi terendah yaitu 64,45%. Hal ini disebabkan oleh kemampuan aplikasi yang hanya dapat mendeteksi warna ungu yang tidak mengandung unsur campuran warna lain. Sementara pada bahan uji, warna ungu yang digunakan oleh penulis terdapat warna ungu-biru dan ungu-merah (magenta). Selain ungu, warna jingga juga memiliki akurasi buruk saat diuji pada objek kertas *glossy*. Hal ini karena, kertas jenis *glossy* memiliki permukaan yang mengkilap dan menghasilkan warna cetakan yang lebih cerah serta intensitas cahaya yang tinggi menyebabkan warna jingga hampir menyerupai warna merah. Oleh karena itu, warna jingga cerah sulit dideteksi karena menyerupai perpaduan warna merah dan jingga.

Percobaan 3, yaitu dengan kertas HVS dengan kondisi malam hari di dalam ruangan (indeks cahaya 34 lumen) merupakan percobaan dengan hasil terbaik, yaitu 93,1%. Kondisi ruangan dengan intensitas cahaya tidak terlalu besar membuat komposisi HSV pada warna bahan uji tidak berubah, sehingga aplikasi dapat mengenali warna secara akurat. Sedangkan percobaan 5 merupakan percobaan dengan hasil tingkat akurasi terendah, yaitu 80,47%.

#### 4. Kesimpulan

Dari percobaan yang dilakukan dapat disimpulkan bahwa aplikasi pengenalan warna untuk penderita buta warna menggunakan ruang warna HSV berhasil menjawab tujuan penelitian, karena :

1. Aplikasi ini dirancang untuk membantu para penderita buta warna agar dapat membedakan warna dengan benar. Aplikasi dapat mengenali enam warna solid sekaligus dalam satu waktu yaitu merah, hijau, biru, jingga, kuning, dan ungu dengan tingkat akurasi 83,33% hingga 100%
2. Kondisi terbaik untuk menggunakan aplikasi pengenalan warna yaitu pada malam hari dengan intensitas cahaya 34 lumen di dalam ruangan dengan tingkat akurasi sebesar 93,1% untuk jenis HVS dan 83,57% untuk jenis *glossy*.
3. Selain intensitas cahaya, tingkat akurasi aplikasi juga dipengaruhi oleh jenis objek deteksi. Dari hasil percobaan, kertas HVS lebih baik dalam mengenali warna dibanding kertas *glossy*. Tingkat akurasi pada kondisi intensitas cahaya 4850 Lumen, 250 Lumen, dan 34 Lumen masing-masing adalah 89,52%, 85,48%, dan 93,1%.

**DAFTAR PUSTAKA**

- [1] Purwoko M, "Prevalensi Buta Warna Pada Mahasiswa Universitas Muhammadiyah Palembang," *Jurnal Kedokteran Brawijaya*, Malang, vol.30, No.2, pp.159-162, December 2017.
- [2] Dewi V N, Robert, and Gasim, "Penerapan Model Warna HSV pada Aplikasi Deteksi Warna," *IJCCS*, vol.10, No.10, pp. 1-5, July 2016.
- [3] Kusumanto R D, Tompunu A N, and Pambudi W S, "Klasifikasi Warna Menggunakan Pengolahan Model Warna HSV," *Jurnal Ilmiah Elite Elektro Politeknik Negeri Sriwijaya*, Palembang, vol. 2, No.2, pp. 83-87, September 2011.
- [4] Kuntjoro K, Yenni, and Y. Halim, "Patofisiologi dan Diagnosis Buta Warna", *Cermin Dunia Kedokteran*. Jakarta, vol.41, No.4, pp.268-271, 2014.
- [5] Darmaprawira S, *Warna: Teori dan Kreatifitas Penggunaannya*, 2nd ed. Bandung: ITB, 2002.
- [6] Putranto B B, Hapsari W, and Wijana K, "Segmentasi Warna Citra Dengan Deteksi Warna HSV Untuk Mendeteksi Objek," *Jurnal Informatika*. Yogyakarta, vol.6, No.2, November 2018.
- [7] Munir R. *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Bandung : ITB, 2004.
- [8] Prayitno Y P, Harianto, and Wibowo M C, "Rancang Bangun Aplikasi Pendeteksi Bentuk dan Warna Benda Pada Mobile Robot Berbasis Webcam," *JCONES*. Surabaya, vol.1, No.2, 2012.
- [9] Castleman K. R. , *Digital Image Processing*. New Jersey : Prentice Hall, (1996).
- [10] Ellanda A, Aulia S, and Hariyani Y S, "Perancangan Aplikasi Pembaca Warna Untuk Penderita Buta Warna Berbasis Android", *JETT*. Bandung, vol.1, No.1, pp.59-66, 2014.
- [11] C.P. Riesmala, A.Rizal, L.Novamizanti, "Pengenalan Motif Batik Dengan Analisis Struktur dan Warna Pada Citra Digital", *Skripsi Sarjana IT Telkom*. Bandung, 2012.