

Color Palette menggunakan Python cv2 dan NumPy

Fauziah¹⁾, Yoan Elviralita²⁾, Widya Wisanty³⁾, Ismindari^{4,*}

¹ Politeknik Bosowa, Makassar and 90245, Indonesia

fauziah.uchie@politeknikbosowa.ac.id,

² Politeknik Bosowa, Makassar and 90245, Indonesia

yoan.elviralita@politeknikbosowa.ac.id,

³ Universitas Sawerigading, Jl. Kanda No. 127, Makassar and 90213, Indonesia

wwisanty@gmail.com

⁴ Politeknik Bosowa, Makassar and 90245, Indonesia

**ismindari@politeknikbosowa.ac.id*

Abstrak

Python cv2 dan numPy merupakan perpustakaan fungsi pemrograman yang ditujukan untuk visi komputer waktu nyata. Cv2 sebagai library untuk pengolahan citra objek secara umum baik objek manusia maupun objek kotak, numPy sebagai library untuk komputasi dan perhitungan sehingga dengan mengkolaborasi library ini diperoleh proses deteksi objek citra dengan proses iterasi menggunakan numerical lebih cepat dan akurat. Penelitian ini menggunakan metode eksperimental dimana akan dibuat jendela yang berisi palet warna RGB dengan *trackbar* menggunakan kolaborasi fungsi dari perpustakaan cv2 dan numPy. Dengan memindahkan *trackbar*, nilai warna RGB akan berubah warna pada rentang 0 hingga 255 dengan nilai heksadesimal sesuai dengan standar ISO 20677.

Keywords: Palet, warna, openCV, numPy

I. PENDAHULUAN

Penelitian ini merupakan dasar dari suatu sistem perangkat lunak pendeteksi warna yang dapat digunakan untuk kepentingan suatu objek atau area visual. Perangkat lunak pendeteksi warna merupakan salah satu perangkat pengambil keputusan dalam menentukan suatu objek.

Penelitian tentang deteksi warna pada buah saat ini banyak dilakukan. Sekelompok peneliti melakukan pendekatan computer vision untuk memperbaiki gambar pohon yang terkena bayangan, lalu menghitung buah dengan metode K-Means. Peneliti lain juga melakukan mengembangkan metode ekstraksi fitur buah hijau dengan menggabungkan teknik model konversi warna, thresholding, histogram equalization, dan penyaringan spasial menggunakan operator Laplace dan Sobel dan Gaussian blur [1], Peneliti lain juga mendeteksi buah jeruk non matang dengan kesulitan membedakan warnanya dengan daun pohon jeruk sendiri. Namun dari penelitian-penelitian ini akurasi yang dimiliki dari sistem masih cukup rendah yakni $49.03 \pm 1,4\%$ dalam hal deteksi jeruk non matang dan 91.3% untuk kasus deteksi jeruk yang matang dengan pohon berbayang [1]. Sekelompok peneliti menunjukkan bahwa penggunaan RGB histogram dalam segmentasi citra menggunakan SVM untuk klasifikasi rambu lalu lintas dapat mempercepat proses segmentasi objek, dan peneliti lain menjelaskan mengenai penggunaan segmentasi RGB masih memerlukan bantuan metode lain untuk mendeteksi objek lebih akurat [2][3].

Penelitian ini menggunakan dua *library* yang ada pada pemrograman python dengan mengkolaborasi fungsi masing – masing *library* untuk mendeteksi citra dengan proses iterasi menggunakan numerical sehingga membantu dalam proses deteksi objek lebih cepat dan akurat.

II. KAJIAN LITERATUR

Python merupakan bahasa pemrograman interpretatif multiguna yang memiliki filosofi perancangan fokus pada tingkat keterbacaan kode. Python disebut sebagai bahasa yang menggabungkan kapabilitas, kemampuan, dengan sintaksis kode yang sangat jelas dan dilengkapi dengan fungsionalitas pustaka standar yang besar serta komprehensif. Python didukung oleh komunitas besar di seluruh dunia dan perkembangannya sangat pesat. Python mendukung pemrograman multi paradigma yakni pada pemrograman berorientasi objek, pemrograman imperatif, dan pemrograman fungsional. Salah satu fitur yang tersedia pada Python adalah sebagai bahasa pemrograman dinamis yang dilengkapi dengan manajemen memori otomatis. Seperti halnya pada bahasa pemrograman dinamis lainnya, Python umumnya digunakan untuk pemrograman dasar ataupun tingkat lanjut [4].

Pada penelitian ini kami fokus membahas library yang ada pada Python yaitu numPy (numerical python) dan cv2 (computer vision versi 2). Cv2 merupakan versi kedua dari OpenCV yang sebelumnya adalah cv. Pengguna menganggap OpenCV-Python lebih mudah dipelajari dari pada cv2, dan mesin pencari juga mendukung dan mempermudah menemukan paketnya. OpenCV (Open Source Computer Vision)-Python adalah perpustakaan ikatan Python yang dirancang untuk memecahkan masalah visi komputer. OpenCV-Python menggunakan numPy, yang merupakan perpustakaan dengan keandalan yang sangat optimal mirip dengan sintaks gaya MATLAB [5]. OpenCV (Open Source Computer Vision Library) adalah perpustakaan perangkat lunak visi komputer dan pembelajaran mesin sumber terbuka. OpenCV dibangun

untuk menyediakan infrastruktur umum untuk aplikasi visi komputer dan untuk mempercepat penggunaan persepsi mesin dalam produk komersial. Menjadi produk berlisensi Apache 2, OpenCV memudahkan bisnis untuk menggunakan dan memodifikasi kode. Perpustakaan ini memiliki lebih dari 2500 algoritma yang dioptimalkan, mencakup beberapa algoritma *machine learning* dan visi komputer klasik. Algoritma yang dimiliki perpustakaan ini dapat digunakan untuk mendeteksi dan mengenali wajah, mengidentifikasi objek, mengklasifikasikan tindakan manusia dalam video, melacak pergerakan kamera, melacak objek bergerak, mengekstrak model objek 3D, dan menyatukan gambar untuk menghasilkan resolusi tinggi. OpenCV memiliki lebih dari 47 ribu orang pengguna komunitas dan perkiraan jumlah unduhan melebihi 18 juta. Perpustakaan OpenCV digunakan secara luas di perusahaan, kelompok penelitian, dan badan pemerintah [6]. Semua properti anggota OpenCV yang sama dikonversi dan dihapus dari anggota NumPy yang sama. Hal ini mempermudah integrasi dengan perpustakaan NumPy lain seperti SciPy dan Matplotlib [5].

Generator yang mengikat OpenCV diberi nama cv2 (versi OpenCV yang sebelumnya dikenal sebagai cv). generator di Python berfungsi untuk menghasilkan nilai secara dinamis sehingga tidak perlu lagi menyimpan semuanya di dalam memori. Generator ini pada saat dipanggil akan menghasilkan nilai satu persatu, hal ini untuk mengurangi penggunaan memori dan sangat cocok digunakan untuk data yang besar dan tidak terbatas. Fungsi generator ini di dalam pemrograman menggunakan pernyataan "yield".

Ada beberapa metode untuk merepresentasikan data warna. Meskipun RGB paling banyak digunakan untuk pengambilan dan tampilan, RGB tidak selalu merupakan yang terbaik untuk pemrosesan gambar, karena representasinya tidak seragam. Artinya jika kita mengubah nilai RGB dengan jumlah yang tetap, perbedaan yang diamati bergantung pada nilai RGB aslinya [7].

Setiap warna yang ada pada citra/gambar adalah kombinasi dari warna RGB (Red, Green, Blue). Nilai pembacaan warna dari 0%-100% yaitu 0-255. Jika RGB diberikan nilai 100% pada setiap warna primernya misalnya RGB (255,255,255) maka warna yang akan dihasilkan adalah putih, sebaliknya, jika nilai RGB (0, 0, 0) maka warna yang dihasilkan adalah hitam. Dan Ketika dua warna diberikan nilai 100% satu warna dengan nilai 0%, misal RGB (255,0,255), (255,255,0), (0,255,255), maka terciptalah warna sekunder yang biasa disebut dengan CMY (Cyan, Magenta, Yellow) [8][9].

Pada tahun 1931 warna RGB ditetapkan oleh Commission International de l'Eclairage (CIE). CIE menstandarkan gelombang warna untuk R adalah sebesar 700nm, G sebesar 546.1 nm dan B sebesar 435.8 nm [10].

Standar Manajemen warna teknologi gambar — Ekstensi pada arsitektur, format profil, dan struktur data telah ditetapkan pada tahun 2019 yang berlaku sampai saat ini yaitu ISO 20677 [11].

III. METODE PENELITIAN

Metode penelitian ini adalah komparatif analitik dimana dua perpustakaan dibandingkan, dikaji dan dianalisa

untuk menghasilkan warna dengan nilai RGB-nya. Berikut adalah kode program untuk membuat warna RGB.

```
Python program to create RGB color
# palette with trackbars

# importing libraries
import cv2
import numpy as np
```

Pada bagian ini peneliti membuat jendela hitam dengan tiga saluran warna yang memiliki resolusi 512 x 512. Tiga buah trackbar dibuat yaitu 'B', 'G', 'R', B adalah *Blue*, G adalah *Green* dan R adalah *Red* menggunakan fungsi perpustakaan OpenCV yang telah ditentukan sebelumnya. Nilai saluran ditetapkan dari 0 hingga 255. Jendela hitam akan digabungkan dengan trackbar warna tersebut di atas.

```
# empty function called when
# any trackbar moves
def emptyFunction():
    pass

def main():
```

Untuk mendefinisikan fungsi, membuat nama fungsi dan blok kode yang berisi operasi yang ingin dilakukan, dapat dimulai dengan definisi yaitu kata kunci def yang diikuti dengan nama fungsi, tanda kurung, dan titik dua seperti pada kode program di atas [12].

```
# blackwindow having 3 color chanel
image = np.zeros((512, 512, 3), np.uint8)
windowName = "Open CV Color Palette"

# window name
cv2.namedWindow(windowName)

# there trackbars which have the name
# of trackbars min and max value
cv2.createTrackbar('Blue', windowName, 0, 255,
emptyFunction)
cv2.createTrackbar('Green', windowName, 0, 255,
emptyFunction)
cv2.createTrackbar('Red', windowName, 0, 255,
emptyFunction)
```

Instruksi uint8 menggunakan lebih sedikit memori dan beberapa operasi dihitung lebih cepat dibandingkan dengan tipe ganda [7]. Setelah instruksi penggabungan ketiga saluran warna dilakukan, kemudian dibuat gambar komposit dari RGB.

```
# Used to open the window
# till press the ESC key
while(True):
    cv2.imshow(windowName, image)
    if cv2.waitKey(1) == 27:
        break
```

```

# values of blue, green, red
blue = cv2.getTrackbarPos('Blue',
windowName)
green = cv2.getTrackbarPos('Green',
windowName)
red = cv2.getTrackbarPos('Red', windowName)
# merge all three color chanel and
# make the image composites image from rgb
image[:] = [blue, green, red]
print(blue, green, red)

cv2.destroyAllWindows()

# Calling main()
if __name__=="__main__":
main()

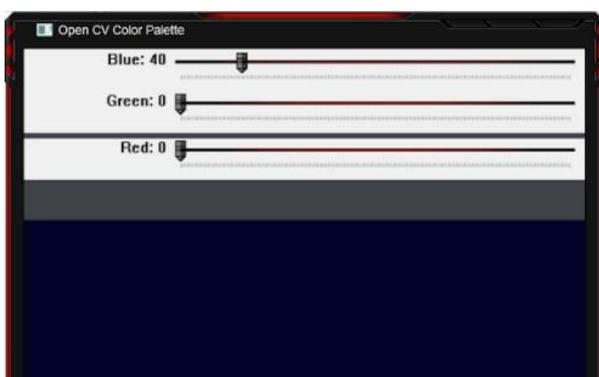
```

Buat jendela hitam dengan tiga saluran warna dengan resolusi 512 x 512. Kemudian buat tiga trackbar 'B' 'G' 'R' menggunakan fungsi pustaka OpenCV yang telah ditentukan sebelumnya. Atur nilai saluran dari 0 hingga 255. Menggabungkan jendela hitam dengan trackbar warna ini.

IV. HASIL DAN PEMBAHASAN

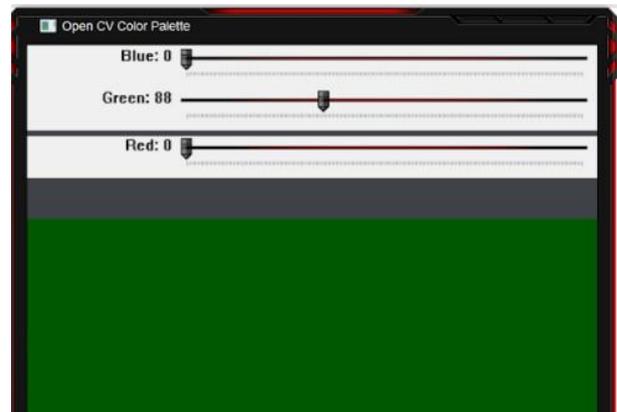
Fungsi `cv2.getTrackbarPos()`, argumen pertama adalah nama trackbar, argumen kedua adalah nama jendela yang dilampirkan, argumen ketiga adalah nilai default, argumen keempat adalah nilai maksimum dan argumen kelima adalah fungsi panggilan balik yang dijalankan setiap kali nilai trackbar berubah. Fungsi callback selalu memiliki argumen default yaitu posisi trackbar. Dalam penelitian ini, *function* tidak melakukan apa pun sehingga di teruskan saja. Aplikasi penting lainnya dari trackbar adalah fungsi tombol atau saklar yang digunakan. OpenCV, secara default, tidak memiliki fungsi tombol sehingga menggunakan trackbar untuk mendapatkan fungsi tersebut.

Trackbar di atur dari 0 sampai 255 sehingga pada saat posisi 0 warna yang tampil adalah hitam sedangkan jika posisi *trackbar* dinaikkan pada bagian warna biru/*blue* pada batas 40 seperti pada gambar 1 di bawah ini maka akan tampil warna biru navy dengan kode heksadesimal #000028.



Gambar 1. Pengaturan *color palette Blue* menggunakan *trackbar* dari 0 - 255

Trackbar pada *palette* dinaikkan pada bagian warna hijau/*green* pada batas 88 seperti pada gambar 2 di bawah ini maka akan tampil warna hijau dengan kode heksadesimal #005800.



Gambar 2. Pengaturan *color palette green* menggunakan *trackbar* 0-255

Kemudian pada *palette* gambar 3 *trackbar* dinaikkan pada bagian warna merah/*red* pada batas 75 maka akan tampil warna merah dengan kode heksadesimal #4B0000



Gambar 3. Pengaturan *color palette red* menggunakan *trackbar* 0 – 255.

Warna RGB telah di uji menggunakan skala *trackbar* 0 – 255 dan terbukti bahwa warna yang dihasilkan sesuai dengan standar ISO 20677 nilai R, G, B dan nilai heksadesimalnya.

V. KESIMPULAN

Kolaborasi dari kedua *library* python yaitu `cv2` dan `numpy` menjadi salah satu pilihan dari produk *open source* yang memudahkan peneliti lain untuk melakukan pengembangan untuk mempermudah proses deteksi objek dan waktu pemrosesan yang cepat. mendeteksi citra dengan proses iterasi menggunakan numerical sehingga membantu dalam proses deteksi objek lebih cepat dan akurat

REFERENSI

- [1] W. Maldonado and J. C. Barbosa, "Automatic green fruit counting in orange trees using digital images," *Comput. Electron. Agric.*, vol. 127, pp.572–581, 2016, doi: 10.1016/j.compag.2016.07.023.
- [2] V. Rajinikanth and M. S. Couceiro, "RGB histogram based color image segmentation using firefly algorithm," *Procedia Comput. Sci.*, vol. 46, pp. 1449–1457, 2015, doi: 10.1016/j.procs.2015.02.064.
- [3] S. B. Wali, M. A. Hannan, A. Hussain, and S. A. Samad, "An Automatic Traffic Sign Detection and Recognition System Based on Colour Segmentation, Shape Matching, and SVM," *Mathematical Problems in Engineering*, vol. 2015, hindawi.com, 2015. doi: 10.1155/2015/250461.
- [4] M. A. Qadar, A. Hidayatno, and Y. A. A. Soetrisno, "Aplikasi Pendeteksi Kotak Untuk Menentukan Ukuran Tubuh Menggunakan Algoritma Hsv," *Transient J. Ilm. Tek. Elektro*, vol. 9, no. 1, pp. 8–16, 2020, doi: 10.14710/transient.v9i1.8-16.
- [5] S. Parveen and J. Shah, "A motion detection system in python and opencv," *Proc. 3rd Int. Conf. Intell. Commun. Technol. Virtual Mob. Networks, ICICV 2021*, pp. 1378–1382, 2021, doi: 10.1109/ICICV50876.2021.9388404.
- [6] "ABOUT | OpenCV." <http://opencv.org/about.html>
- [7] A. Abdullah, W. Palash, A. Rahman, and S. M. A. Alim, "Digital Image Processing Analysis using Matlab," *Am. J. Eng. Res.*, vol. 5, no. 4, p. 5, 2016, [Online]. Available: www.ajer.org
- [8] M. Z. Andrekha and Y. Huda, "Deteksi Warna Manggis Menggunakan Pengolahan Citra dengan Opencv Python," *Voteteknika (Vocational Teknik Elektronika dan Informatika)*, vol. 9, no. 4, p. 27, 2021. doi: 10.24036/voteteknika.v9i4.114251.
- [9] A. F. Hastawan, R. Septiana, and Y. E. Windarto, "Perbaikan Hasil Segmentasi HSV Pada Citra Digital Menggunakan Metode Segmentasi RGB Grayscale," *Edu Komputika J.*, vol. 6, no. 1, pp. 32–37, 2019, oi: 10.15294/edukomputika.v6i1.23025.
- [10] R. S. Hagai, "Deteksi dan Perhitungan Jumlah Jeruk Matang dan Non Matang berdasarkan Citra Digital," *Deteksi dan Perhitungan Jumlah Jeruk Matang dan Non Matang berdasarkan Citra Digit.*, vol. 12, no. 12, pp. 1–10, 2019.
- [11] "INTERNATIONAL STANDARD Image technology colour management — Extensions to architecture , profile format and data iTeh STANDARD PREVIEW iTeh STANDARD PREVIEW," vol. 2019. 2019.
- [12] J. Cicolani, *An Introduction to OpenCV*. Springer, 2021. doi: 10.1007/978-1-4842-6891-9_9