

DETEKSI OBJEK DENGAN METODE COLOR FILTERING HSV DAN BLOB DETECTION PADA ROBOT VERTICAL TAKE OFF AND LANDING

(STUDI KASUS KONTES ROBOT TERBANG INDONESIA 2018)

Object Detection Using HSV Color Filtering And Blob Detection Methods On Robot Vertical Take Off And Landing

(Case Study : Indonesia Flying Robot Contest 2018)

Arifin Yusuf

Informatika, Universitas Teknokrat Indonesia, Jl.Z.A Pagar Alam No 9-11, Kedaton, Bandar Lampung

E-mail : arifinyusuf96@gmail.com

INTISARI

Pada Kontes Robot Terbang Indonesia 2018, divisi VTOL, setiap tim memiliki misi mengirimkan LOG (Logistik) dan MP (Muatan PICK) Survival Kits berupa objek berwarna orange dan berbentuk lingkaran ke Lokasi-lokasi yang telah ditentukan. Terbatasnya sumber tenaga yang mampu diangkat oleh wahana membuat wahana memiliki batas waktu yang tidak lama ketika berada di udara. Selain itu, kebutuhan pemrosesan citra secara realtime mengharuskan wahana dapat mendeteksi objek dengan cepat dan tepat. Masalah lain adalah kondisi lingkungan yang tidak bisa diprediksikan.

Salah satu metode yang populer digunakan untuk mendeteksi objek adalah *learning*. Metode *learning* akan menciptakan algoritma yang dapat mendeteksi objek berdasarkan sampel yang diberikan saat proses *learning* sehingga metode ini memiliki tingkat akurasi yang tinggi, namun memiliki beban kerja yang berat. Selain *learning*, *color filtering* juga dapat digunakan untuk mendeteksi objek, namun metode ini memiliki kelemahan karena hanya melakukan sekali filter sehingga belum mampu sepenuhnya mengeliminasi noise jika ada objek lain yang memiliki warna serupa dengan objek deteksi.

Karena itu digunakanlah *Blob detection* setelah dilakukan filter warna. *Blob detection* mendeteksi kumpulan pixel yang memiliki perbedaan warna (lebih terang atau lebih gelap). Dari hasil pengujian didapatkan bahwa metode ini mampu mendeteksi objek hingga jarak x meter. Selain itu, metode ini juga mampu mengeliminasi noise dari objek lain dengan warna serupa maupun noise getaran.

Kata kunci : Deteksi Objek, *Color Filtering* HSV, BLOB *Detection*. *Vertical Take Off And Landing*

ABSTRACT

At the Indonesia Flying Robot Contest 2018, the VTOL division, each team had a mission to send LOG (Logistics) and MP (Load PICK) Survival Kits in the form of orange and circular objects to designated locations. The limited power source that can be raised by the vehicle makes the vehicle has a time limit that is not long when in the air. In addition, the need for realtime image processing requires vehicles to detect objects quickly and precisely. Another problem is environmental conditions that cannot be predicted.

One popular method used to detect objects is learning. The learning method will create an algorithm that can detect objects based on the samples given during the learning process so that this method has a high degree of accuracy, but has a heavy workload. In addition to learning, color filtering can also be used to detect objects, but this method has the disadvantage of only doing one filter so that it has not been able to completely eliminate noise if there are other objects that have colors similar to the object of detection.

Therefore Blob detection is used after the color filter. Blob detection detects a collection of pixels that have different colors (lighter or darker). From the test results it was found that this method is able to detect objects up to a distance of x meters. In addition, this method is also able to eliminate noise from other objects of the same color or vibration noise.

Key word : Deteksi Objek, *Color Filtering* HSV, BLOB *Detection*. *Vertical Take Off And Landing*

1. Pendahuluan

Kontes Robot Terbang Indonesia (KRTI) merupakan perlombaan robot yang diadakan secara rutin setiap tahun oleh Kemenristek Dikti. KRTI sendiri terbagi dalam 4 divisi yaitu Fixed Wing Mapping, Racing Plane, Vertical Take Off Landing, dan Technology Development.

Pada KRTI 2018, divisi VTOL memiliki tema “Pick and Drop Survival Kits”. Dengan misi wahana mengirimkan LOG (Logistik) dan MP (Muatan PICK) Survival Kits ke Lokasi-lokasi yang telah ditentukan. Logistik yang dipindahkan berupa objek berwarna orange dan berbentuk lingkaran. Dengan keterbatasan sumber tenaga yang mampu diangkat oleh wahana membuat wahana memiliki batas waktu yang tidak lama ketika berada di udara. Selain itu, kebutuhan pemrosesan citra secara realtime mengharuskan wahana dapat mendeteksi objek dengan cepat dan tepat. Masalah lain adalah kondisi lingkungan yang tidak bisa diprediksi. Meskipun diatur sedemikian rupa, tetap menyisakan ruang yang dapat menyebabkan perbedaan warna yang diakibatkan oleh pengaruh pencahayaan.

Sehingga diperlukan cara yang dapat dengan cepat mampu mendeteksi objek serta mengeliminasi gangguan dari noise di sekitar objek ketika wahana sedang berada di udara. Noise dapat berupa objek lain yang memiliki warna atau bentuk serupa dengan logistik. Salah satu metode yang populer digunakan untuk mendeteksi objek adalah *learning*. Metode *learning* akan menciptakan algoritma yang dapat mendeteksi objek berdasarkan sampel yang diberikan saat proses *learning* (Lukman dan Baharuddin, 2012). Oleh karena itu, metode ini memiliki tingkat akurasi yang tinggi. Akan tetapi memiliki beban kerja yang berat. Pada penelitian sebelumnya hanya menggunakan metode *color filtering* untuk mendeteksi objek. Metode ini memiliki kelemahan karena hanya melakukan sekali filter sehingga belum mampu sepenuhnya mengeliminasi noise jika ada objek lain yang memiliki warna serupa dengan objek deteksi.

Blob detection adalah metode lain yang dapat digunakan untuk mendeteksi bentuk. Metode ini mendeteksi kumpulan pixel yang memiliki perbedaan warna (lebih terang atau lebih gelap) (Pamungkas *et al*, 2017). Untuk menggunakan metode ini terlebih dahulu harus dilakukan filter warna untuk mendeteksi warna orange dan mengeliminasi noise dari objek lain. Setelah dilakukan deteksi warna, selanjutnya kembali dilakukan filter menggunakan metode *blob detection*. Kombinasi antara metode *color filtering* dan *blob*

detection dinilai efektif karena memiliki beban kerja yang kecil sehingga wahana masih dapat mendeteksi objek secara *realtime* serta lebih akurat karena dilakukan dua kali filter.

2. Metode Penelitian

1.1. Tempat Penelitian

Ruang lingkup dari penelitian ini dilakukan di Universitas Teknokrat Indonesia, khususnya di ruang laboratorium robotic. Yang di jadikan objek penelitian adalah Drone Krakatau X-8 yang dijadikan sebagai objek penelitian. Robot Drone Krakatau X-8 tersebut digunakan untuk Kontes Robot Terbang Indonesia (KRTI) 2018, yang diselenggarakan di Universitas Teknokrat Indonesia, Lampung.

1.2. Analisis Sistem

Pada bab ini akan di bahas mengenai metode penelitian menggunakan metode eksperimen dan studi literatur. Sistem yang akan dirancang untuk penelitian ini merupakan suatu system pendeteksian objek berbasis pengolahan citra dengan metode berbasis HSV dan *BLOB detection*.

1.3. Alat dan Bahan Penelitian

Adapun alat dan bahan yang digunakan dalam penelitian ini terdiri dari perangkat keras komputer (*hardware*) dan Perangkat lunak (*software*), dengan bahan penelitian adalah rule (aturan) perlombaan pada KRTI 2018 devisi VTOL.

1.3.1. Perangkat Keras

Perangkat keras (*hardware*) yang digunakan pada penelitian ini, Laptop sebagai median Pemograman dan Analisa, Mini PC Intel NUC7I5BNH, Kamera Logitech c920 dan Pixhawk Cube 2.1.

- a. Laptop SAMSUNG 355V4C
- b. Mini PC Intel NUC7I5BNH
- c. Kamera Logitech c920
- d. Pixhawk Cube 2.1

1.3.2. Perangkat Lunak

Perangkat lunak (*software*) yang digunakan pada penelitian ini adalah sebagai berikut :

- 1) Mission Planner.
merupakan salah satu software yang dirancang untuk mempermudah user dalam perencanaan terbang otomatis dan biasanya digunakan untuk pemetaan. Perencanaan seperti ini termasuk pada tahap Pra-akuisisi data (persiapan sebelum penerbangan).
- 2) Sublime Text Editor.

Sublime Text Editor adalah teks editor untuk berbagai bahasa pemrograman termasuk pemrograman PHP. Sublime Text merupakan teks editor lintas platform dengan Phyon Application Interface (API).

- 3) Python
Python digunakan sebagai bahasa skrip dan dapat digunakan untuk berbagai keperluan pengembangan perangkat lunak dan dapat berjalan di berbagai platform sistem operasi.
- 4) Visual Studio
Microsoft Visual Studio dapat digunakan untuk mengembangkan aplikasi dalam native code (dalam bentuk bahasa mesin yang berjalan di atas Windows) dan sebagai sarana tool.
- 5) Open CV
Open CV digunakan untuk mengolah gambar dan video hingga kita mampu meng-ekstrak informasi didalamnya. Dan OpenCV dapat berjalan di berbagai bahasa pemrograman.
- 6) Linux Ubuntu 18.04, Elementary OS.
Di gunakan sebagai sistem operasi mini pc yang di letakan di drone.
- 7) Virtual Network Computing (VNC)
Digunakan sebagai remote-control untuk mengontrol komputer lain melalui koneksi network. Untuk mengontrol satu komputer ke komputer lainnya sehingga dapat mengelola sebuah desktop.

1.3.3. Bahan Penelitian

Yang menjadi bahan penelitian dalam penelitian ini adalah aturan perlombaan dalam KRTI 2018 devisi VTOL yang di keluarkan oleh panitia RISTEKDIKTI. Pada devisi VTOL peserta diwajibkan untuk dapat mengirimkan logistik dan mengambil muatan pick secara *outonomous*.

1.4. Metode Pengumpulan Data

Metode pengumpulan data pada penelitian ini yaitu menggunakan metode eksperimen dan studi literatur. Berikut adalah penjelasannya :

- 1) Eksperimen
Pengumpulan data yang dilakukan dengan cara melakukan percobaan terhadap suatu hal, yang dilanjutkan dengan melakukan pengamatan dan pencatatan terhadap yang berkaitan dengan penelitian. Hasil dari pengamatan adalah data kuantitatif yaitu berupa angka, dapat berupa waktu deteksi ataupun jarak maximum system dapat melakukan deteksi.
- 2) Studi Literatur
Yaitu data yang diperoleh melalui literatur, melakukan studi kepustakaan dalam mencari bahan bacaan dari internet, membaca jurnal, dan

membaca buku yang berkaitan sesuai dengan objek dan parameter yang sedang diteliti.

1.5. Tahapan Penelitian

Pada penelitian ini, penulis melakukan penelitian eksperimen dengan mengangkat studi kasus pada Kontes Robot Terbang Indonesia (KRTI) 2018, Divisi Robot VTOL yang diselenggarakan di Universitas Teknokrat Indonesia, Lampung.



Gambar 3.1 Tahapan penelitian

Dari gambar 3.1, berikut penjelasan dari masing-masing tahapan penelitian :

1.5.1. Analisa Kebutuhan

Pada tahapan analisa kebutuhan dalam penelitian ini, peneliti melakukan analisa kebutuhan dalam membangun sebuah mekanisme maupun elektronika yang dibutuhkan, mekanisme seperti desain drone, tataletak sensor kamera dan peletakan komponen elektronika, adapun perlengkapan lain yang dibutuhkan seperti objek yang akan diteliti dan lapangan kontes.

Pada tahapan ini peneliti menganalisa *rules* yang diberikan panitia penyelenggara kontes KRTI 2018 devisi VTOL.

1.5.1.1. Kontes Robot

Robot Terbang atau Pesawat Tanpa Awak (Unmanned Aerial Vehicle, UAV) atau Unmanned Aircraft System (UAS) adalah wahana terbang nir-awak yang dalam satu dasawarsa terakhir ini berkembang kian pesat di ranah riset unmanned system (sistem nir-awak) di dunia. (Rule KRTI 2018)

1.5.1.2. Rules KRTI 2018 Divisi VTOL

Tema divisi VTOL KRTI 2018 ini adalah “Pick and Drop Survival Kits”. Artinya wahana wajib mengirimkan logistik dan muatan pick ke lokasi - lokasi yang telah ditentukan. logistik dikirim ke lokasi drop di lokasi yellow, sedangkan muatan pick 1 dan muatan pick 2 dikirim ke lokasi drop option. Pengiriman dilakukan dengan cara menjatuhkannya dari ketinggian yang cukup rendah di tiap lokasi yang telah ditentukan.

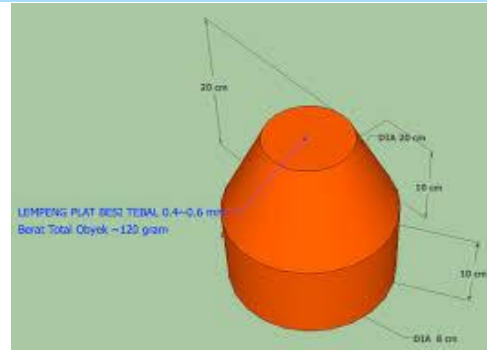
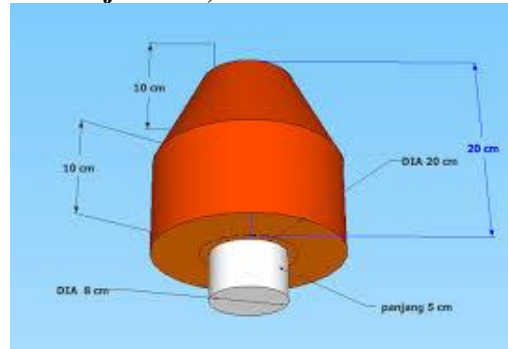
1.5.1.3. Lapangan Kontes VTOL



Gambar 3. 2 Lapangan Kontes VTOL

- H = HOME
 - DM dan DY = Drop di lokasi Yellow
 - MP = Muatan Pick
 - DO = Drop Option
 - P = Lokasi Awal MP
- Lokasi yang ditandai : H adalah HOME; P-1 dan P-2 adalah lokasi Wal MP-1 dan MP-2; DY-1, DM dan DY-2 adalah lokasi dropping Logistik LOG; DO-1, DO-2 dan DO-3 adalah lokasi dropping MP-1 atau MP-2

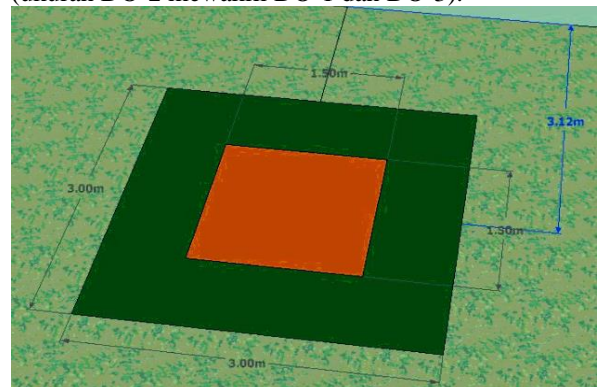
1.5.1.4. Objek LOG, MP-1/2 dan Lokasi



Gambar 3. 3 Detil Dimensi MP-1/2

1.5.1.5. Lokasi Dropping Muatan MP

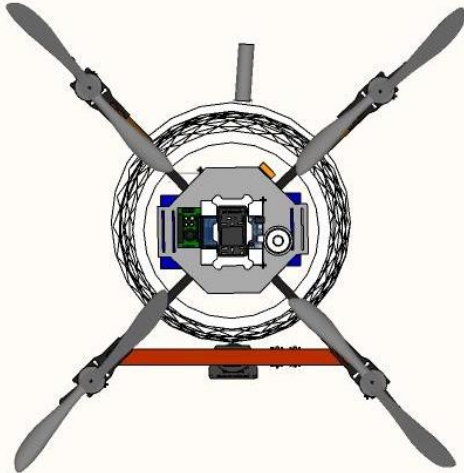
Lokasi dropping muatan MP-1/2 adalah landasan bujur-sangkar berwarna dasar HIJAU dan ORANGE seperti dalam Gambar 9 berikut ini (ukuran DO-2 mewakili DO-1 dan DO-3).



Gambar 3. 4 Lokasi Dropping Muatan MP

1.5.2. Desain dan Membuat Kerangka Drone

Pada tahapan ini peneliti melakukan desain kerangka drone yang bertujuan untuk menentukan ukuran, bentuk, berat dan bahan yang akan digunakan. Untuk membuat wahana drone dan menentukan tataletak komponen elektronika. Lalu setelah itu membuat kerangkanya, berikut merupakan desain kerangka wahana drone yang dapat di lihat pada gambar 3.3 dan gambar 3.4 berikut :



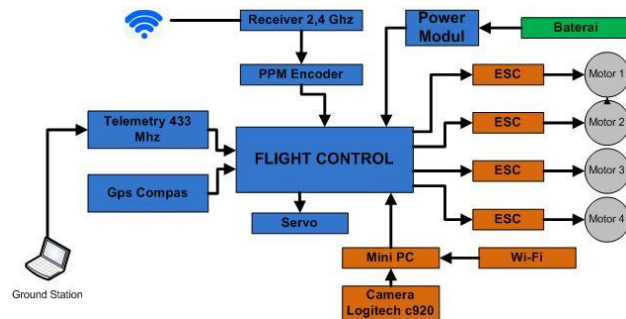
Gambar 3. 5 Tampak Atas



Gambar 3. 6 Tampak Depan

1.5.3. Desain Arsitektur System

Diagram blok ini menggambarkan secara garis besar alur pengontrolan drone dan kamera deteksi. Dibuat sebuah perancangan diagram blok keseluruhan sisitem yang digunakan.



Gambar 3. 7 Desain Arsitektur System

1.5.4. Pemasangan Rangkaian Elektronika

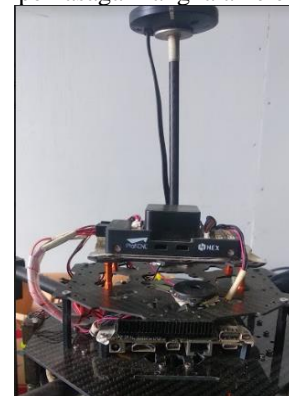
Tada tahapan ini peneliti melakukan perakitan dan pemasangan semua komponen

elektronika pada kerangka muatan roket berdasarkan desain yang telah dibuat sebelumnya. Pada tahapan ini semua modul-modul ditetapkan sesuai dengan desain yang sudah dibuat dan mengatur letak rangkaian elektronik agar tidak mengganggu komponen lainnya. Berikut gambar modul utama yang dipakai :



Gambar 3. 8 Koneksi Modul Utama

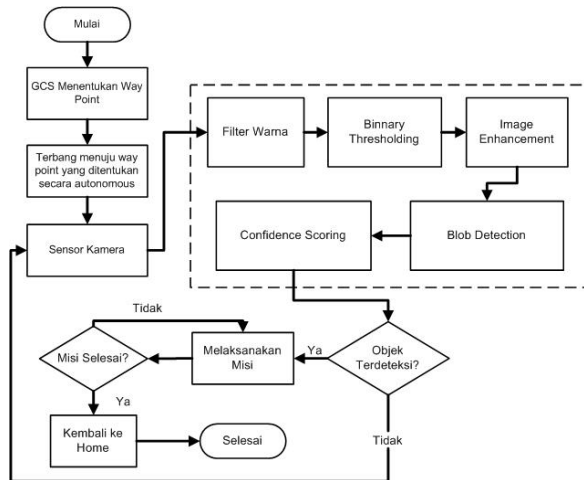
Berikut merupakan gambar setelah dilakukan perakitan dan pemasangan rangkaian elektronika :



Gambar 3. 9 Drone VOTOL Tampak Depan

1.6. Flowchart Strategi Deteksi Objek

Padatahapan ini peneliti melukukan implementansi program sekaligus merancang strategi deteksi yang akan digunakan pada drone. Untuk menemukan objek dan secara bersamaan mengeliminasi noise dari objek yang memiliki kemiripan warna ada beberapa langkah yang perlu dilakukan. Langkah – langkah tersebut akan dijelaskan pada *flowchart* di bawah ini.



Gambar 3. 10 Flowchart Strategi Deteksi Objek

Pada gambar 3.10 dapat dijelaskan bahwa start awal robot berada pada posisi Home, kemudian memasukan titik kordinat pada GCS untuk menentukan titik waypoint, setelah menentukan waypoint robot akan bergerak secara autonomus bergerak ke titik kordinat waypoint yang diberikan, setelah mencapai titik kordinat yang ditentukan camera akan aktif untuk melakukan proses pendeksian, di dalam proses pendeteksian terdapat beberapa proses yang akan di lakukan yaitu filter warna, binary thresholding, images enhancement, blob detection dan confidence scoring. Untuk menemukan objek yang akan dideteksi, setelah objek di temukan drone akan melakukan misi mengambil objek kemudian membawa ke waypoint berikutnya. Kemudian misi selesai dan kembali ke titik home.

1.7. Sistem Deteksi

1.7.1. Filter Warna

Ruang warna ini dipilih karena dapat memisahkan antara warna dan cahaya. Seperti kondisi lingkungan dalam permainan, meskipun diatur sedemikian rupa, tetap menyisakan ruang yang dapat menyebabkan perbedaan warna yang di akibatkan oleh penaruh pencahayaan. Oleh karenanya dibutuhkan filter warna yang tidak sensitif terhadap perbedaan intensitas cahaya. Untuk itu digunakan domain warna HSV (Hue Saturation Value). Proses ini akan mengeliminasi warna yang tidak diinginkan sehingga nantinya hanya menyisakan warna yang diinginkan (warna orange). Filter warna dipilih arena objek yang dideteksi memiliki fitur berupa warna orange. Sehingga metode ini lebih cocok dibandingkan ekstraksi fitur lain seperti *local binary pattern* (LBP) yang mendeteksi fitur berdasarkan pola dari objek .

Pada pengolahan warna gambar, ada bermacam-macam model warna. Warna RGB (*red green blue*) merupakan model yang banyak digunakan, salah satunya adalah camera. Selain model

RGB terdapat model HSV dimana model ini terdapat 3 komponen yaitu *hue*, *saturation*, dan *value* (Neneng and Fernando, 2017). *Hue* adalah suatu ukuran panjang gelombang yng terdapat pada warna dominan yang diterima oleh penglihatan sedangkan *Saturation* adalah ukuran banyaknya cahaya putih yang percampur pada *hue*.

Untuk mentransformasi dari RGB ke HSV. Diasumsikan koordinat-koordinat R, G, B [0,1] adalah berurutan merah, hujau, biru dalam ruang warna RGB dengan *max* adalah nilai maksimum dari nilai *red*, *green*, *blue* dan *min* adalah nilai minimum dari nilai *red*, *green*, *blue*. Unruk memperoleh sudut *hue* [0.360] yang tepat untuk ruang warna HSV, menggunakan rumus seperti berikut:

$$H = \begin{cases} 60^\circ \times \left(\frac{(G - B)}{\max - \min} \text{mod } 6 \right), & \text{jika } MAX = R \\ 60^\circ \times \left(\frac{(B - R)}{\max - \min} + 2 \right), & \text{jika } MAX = G \\ 60^\circ \times \left(\frac{(R - G)}{\max - \min} + 4 \right), & \text{jika } MAX = B \\ 0, & \text{jika } MAX = MIN \end{cases}$$

Nilai-nilai untuk *S* dan *V* pada HSV di definisikan sebagai Berikut :

$$S (\text{saturation}) = \begin{cases} 0, & \text{jika } \max = \min \\ \frac{\max - \min}{V} & \text{otherwise} \end{cases}$$

$$V (\text{value}) = MAX$$

Rumus di atas menghasilkan nilai value dan saturatio dalam jangkauan RGB [0,1]. Kalikan dahulu dengan 255 untuk memperoleh nilai dengan jangkauan RGB [0,255]. Misalnya ingin ditrasformasikan RGB (65, 27, 234) kedalam bentuk HSV, maka langkahnnya adalah sebagai berikut :

Setiap nilai RGB (65, 27, 234) diubah dalam jangkauan [0,1] dengan membagi setiap nilai dengan 255 :

$$\text{menjadi } \left(\frac{65}{255}, \frac{27}{255}, \frac{234}{255} \right) = (0.255, 0.106, 0.918)$$

RGB(0.255, 0.106, 0.918) ini yang akan ditransformasikan ke bentuk HSV

$$\begin{aligned} \max &= \text{nilai B (blue)} = 0.918, \\ \min &= \text{nilai G (green)} = 0.106, \\ \max - \min &= 0.918 - 0.106 = 0.812. \end{aligned}$$

$$\begin{aligned} H (\text{Hue}) &= 60^\circ \times \left(\frac{R - G}{\max - \min} + 4 \right), & \text{karena } \max = B (\text{blue}) \\ &= 60^\circ \times \left(\frac{0.255 - 0.106}{0.812} \right) = 251^\circ \end{aligned}$$

$$\begin{aligned} V (\text{value}) &= \max \\ &= 0.918 \end{aligned}$$

$$S (\text{saturation}) = \frac{\max - \min}{V}, \text{ karena } \max < \min$$

$$= \frac{0.812}{0.918} = 0.885$$

Sehingga nilai RGB (65, 27, 234) ditrasformasikan menjadi HSV (251^o, 0.885, 0.918) dengan jangkauan RGB [0.1].

```
def rgb_to_hsv(r, g, b):
    r, g, b = r / 255.0, g / 255.0, b / 255.0

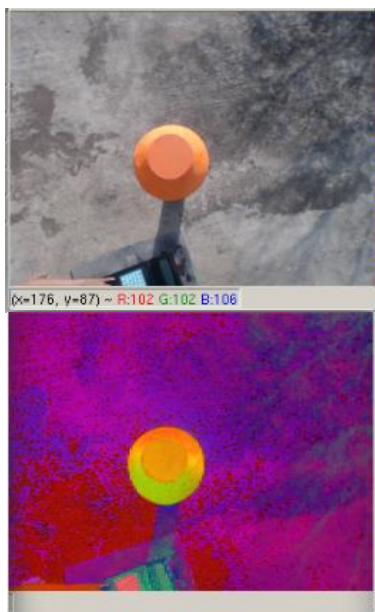
    # h, s, v = hue, saturation, value
    cmax = max(r, g, b) # mencari nilai maximum dari r, g dan b
    cmin = min(r, g, b) # mencari nilai minimum dari r, g dan b
    diff = cmax - cmin # mengurangi nilai maximum dan minimum

    if cmax == cmin:
        h = 0
    elif cmax == r:
        h = (60 * ((g - b) / diff) + 360) % 360
    elif cmax == g:
        h = (60 * ((b - r) / diff) + 120) % 360
    elif cmax == b:
        h = (60 * ((r - g) / diff) + 240) % 360
    if cmax == 0:
        s = 0
    else:
        s = (diff / cmax) * 100

    v = cmax * 100
    return h, s, v
```

Gambar 3. 11 Source Code filter warna

Untuk merubah RGB ke HSV di lakukan filter warna untuk memilah warna yang nantinya akan digunakan sebagai warna yang akan di pilih. Untuk mencari Hue= cmax=h=0 merupakan nilai terkecil sedangkan cmax=r untuk mencari nilai max dari red, dan cmax=g untunk mencari nilai max dari nilai green, cmax=b untuk mencari nilai max dari blue. Untuk nilai saturation adalah cmax=0 s=0 makan nilai s = nilai max dan min / nilai dari s kemudian dikaliakn dengan 100. Untuk nilai value yaitu cmax dikalikan 100.



Gambar 3. 12 Hasil Filter Warna RGB to HSV

1.7.2. Binary Thresholding

Untuk setiap warna yang dibutuhkan, dilakukan binary thresholding dimana jika pixel pada citra berada pada rentang yang ditentukan. Proses ini bertujuan untuk memisahkan warna pada rentang nilai hue antara 5 -15 (warna orange) dengan warna di luar rentang tersebut (selain orange). Sehingga nantinya warna orange akan dirubah menjadi warna putih, dan warna selain orange akan dirubah menjadi warna hitam.

$$map(i,j) \begin{cases} idc, & \text{if } L_thresh \geq img(i,j) \leq H_thresh \\ map(i,j), & \text{otherwise} \end{cases}$$

Dimana :

map(i,j) : map biner untuk semua warna

idc : Id warna yang dideteksi

L_thresh: Batas bawah nilai pixel

H_thresh: Batas atas nilai pixel

img(i,j) : nilai pixel pada citra input

```
for i in range(rows):
    for j in range(cols):
        color = img[i,j]

        b = 0.144 * img.item(i,j,0)
        g = 0.587 * img.item(i,j,1)
        r = 0.299 * img.item(i,j,2)
        gray = (b + g + r)
        if gray > 255:
            gray = 255
        else:
            gray = gray

        img[i,j] = gray

        if gray > 128:
            img[i,j] = 255
        else:
            img[i,j] = 0
```

Gambar 3. 13 Source Code Binary Thresholding



Gambar 3. 14 Hasil Proses Binnary Thresholding

1.7.3. Image Enhancement

Meskipun sudah dilakukan filter warna dan proses binary thresholding, citra masih belum sepenuhnya terlihat sempurna. Perlu dilakukan proses image enhancement untuk memperbaiki kualitas citra. Proses image enhancement yang digunakan terdiri dari 2 proses yaitu erori dan dilasi. Erosi adalah proses penghapusan titik-titik batas objek menjadi bagian dari latar sedangkan dilasi adalah proses penggabungan titik-titik latar menjadi bagian dari objek, berdasarkan structuring element yang digunakan.



Gambar 3. 15 Bentuk Image Enhancement

$$g(x,y) = f(x,y) \theta SE$$

Dimana :

$g(x,y)$ = pixel hasil Erosi

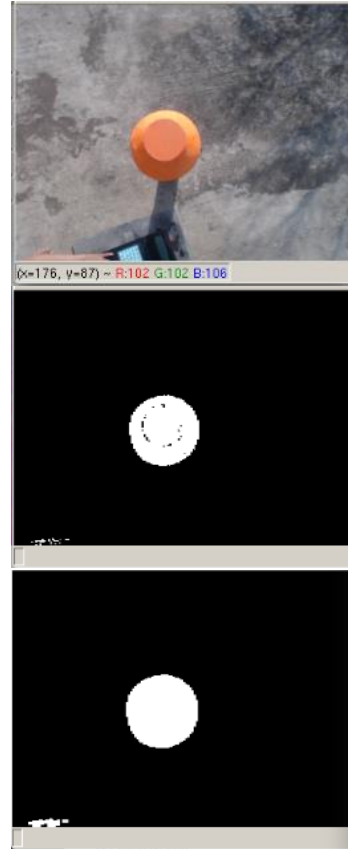
$f(x,y)$ = pixel citra asli

SE = pixel tetangga

```

1 ea = cag'p'f'f'm'f'e' s'ud'(f'f's'w'e' f'f's'w'e' w's'ak= f'w'd'p'u)
2 f'w'd'p'u = cag'q't't'e'r'e(w's'ak'k'e'k'u'e't' f'r'e'i'e'r't'o'n'a = f)
3 f'w'd'p'u = cag'g'o'd'e(w's'ak'k'e'k'u'e't' f'r'e'i'e'r't'o'n'a = f)
4 k'e'k'u'e't = u'b'o'u'e'a((3'3)'u'b'p't'u'e'r'e)
5 w's'ak = cag'f'w'b'y'u'd'e(p'a'l'(p'r'o'm'g'o'm'g'o'm)'(p'h't'a'r'g'h't'a'r'g'h't'a'r))
    
```

Gambar 3. 16 Source Code Image Enhancement



Gambar 3. 17 Hasil Image Enhancement

1.7.4. Blob Detection

Citra threshold yang sudah diperbaiki pada proses sebelumnya, dilakukan deteksi blob untuk menentukan mana objek dan noise. Proses ini akan memilih objek yang bentuknya paling menyerupai lingkaran. Jika terdapat lebih dari satu objek, maka akan dipilih objek dengan ukuran paling besar. Output yang dihasilkan dari proses deteksi ini adalah nilai kordinat xy dan ukuran objek.

Berikut hasil dari proses blob detection ditunjukkan pada gambar Berikut :

```

mask = cv2.inRange(hsv, (hLow,sLow,vLow), (hHigh,sHigh,vHigh))
kernel = np.ones((3,3), np.uint8)
imgEnh = cv2.erode(mask, kernel, iterations = 1)
imgEnh = cv2.dilate(mask, kernel, iterations = 1)
res = cv2.bitwise_and(frame, frame, mask= imgEnh)
    
```



```
#start blob detection
params = cv2.SimpleBlobDetector_Params()

# Change thresholds
params.minThreshold = 10;
params.maxThreshold = 200;

# Filter by Area.
params.filterByArea = False
params.minArea = 500

# Filter by Circularity
params.filterByCircularity = True
params.minCircularity = 0.75

# Filter by Convexity
params.filterByConvexity = True
params.minConvexity = 0.87

# Filter by Inertia
params.filterByInertia = True
params.minInertiaRatio = 0.01

imgInv = cv2.bitwise_not(imgEnh)
detector = cv2.SimpleBlobDetector_create(params)
keypoints = detector.detect(imgInv)
```

Gambar 3. 18 Source Code Blob Detection



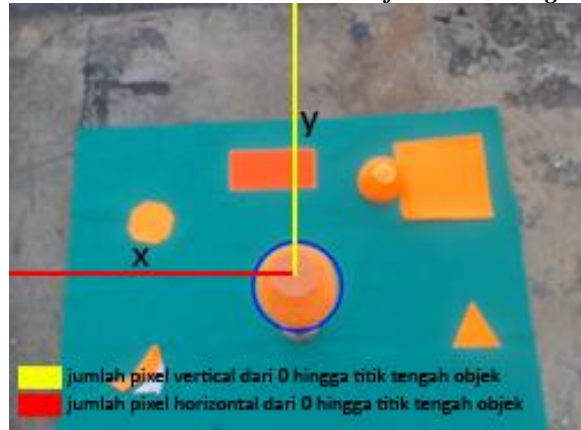
Gambar 3. 19 Hasil Blob Detection

1.7.5. Confidence Scoring

Kandidat yang layak dinyatakan sebagai objek adalah objek yang lolos dalam proses filter warna (berwarna orange) serta berukuran cenderung lingkaran. Untuk menghindari deteksi objek lebih dari satu, proses ini akan memilih objek dengan ukuran terbesar yang dianggap berada paling dekat dengan robot. Selain itu juga akan dilakukan perhitungan ukuran objek dengan rentang nilai konstanta yang sudah ditentukan. Nilai konstanta ini didapat dari hasil sampling ukuran objek dengan ketinggian yang bervariasi. Output dari proses ini adalah nilai variable x dan y yang merupakan kordinat dari titik tengah objek hasil deteksi. Kordinat ini yang akan menjadi acuan wahana dalam menentukan posisi dari mp yang akan di-pick.

```
radius = int(sqrt((x - cx)**2 + (y - cy)**2))
if radius < minRadius:
    continue
if radius > maxRadius:
    continue
if not (minArea <= area <= maxArea):
    continue
if not (minCircularity <= circularity <= maxCircularity):
    continue
if not (minConvexity <= convexity <= maxConvexity):
    continue
if not (minInertiaRatio <= inertiaRatio <= maxInertiaRatio):
    continue
keypoints.append((cx, cy, radius))
```

Gambar 3. 20 Source Code Confidence Scoring



Gambar 3. 21 Hasil Dari Confidence Scoring

1.8. Tahapan Pengujian

1. Pengujian dilakukan secara outdoor.
2. Kondisi lingkungan dibuat menyerupai arena kontes namun hanya berfokus pada area deteksi.
3. Pengujian dilakukan berdasarkan ketinggian yang berbeda.
4. Pengujian berdasarkan ketinggian untuk menguji tingkat akurasi algoritma.

3. Hasil dan pembahasan

1.1. Analisis dan Masalah KRTI 2019

Berdasarkan penjelasan pada bab sebelumnya, dapat disimpulkan bahwa masalah pada KRTI 2019 sebagai berikut :

- 1) Pencahayaan

Karena KRTI merupakan kontes yang bersifat outdoor, sehingga kondisi lingkungan kontes tidak bisa diprediksi. Kondisi lingkungan meskipun sudah diatur sedemikian rupa, tetap saja dapat menyisakan ruang yang dapat menyebabkan perbedaan warna yang diakibatkan oleh pengaruh pencahayaan.
- 2) Noise

Arena yang bersifat outdoor dan wahana cenderung bergerak, mengakibatkan noise pencahayaan, juga noise getaran di karnekan hawana bergerak menimbulkan fibrasi.
- 3) Waktu

Terbatasnya waktu yang diberikan oleh panitia untuk setiap kesempatan membuat setiap tim harus menyusun algoritma yang dapat dengan

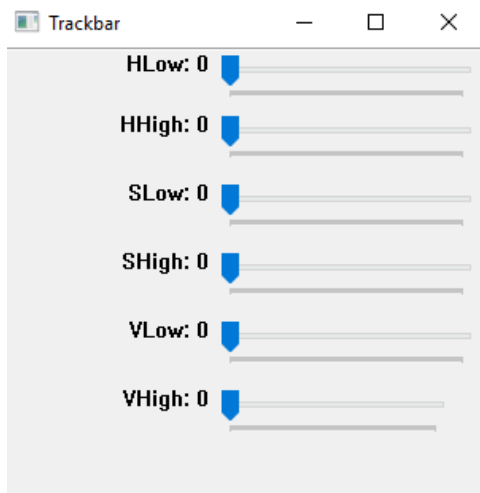
cepat melakukan deteksi. Di samping itu, terbatasnya hardware yang dapat dipasang juga tentunya menjadi masalah sendiri karena hardware yang terbatas dapat mempengaruhi kecepatan algoritma.

Pada bab ini dijelaskan tentang pengujian terhadap strategi pendeteksian objek MP yang telah di buat untuk mengetahui apakah strategi tersebut telah berjalan sesuai dengan yang di diharapkan. Pengujian dilakukan secara bertahap dan berkesinambungan, hasil dari tiap tahapan pengujian kemudian di analisis sehingga dapat diketahui kelebihan maupun kekurangan dari stratgi deteksi objek MP yang di gunakan. Analisis yang dilakukan adalah pengujian yang dilakukan terhadap strategi yang telah dibuat. Pengujian yang di lakukan meliputi pengujian warna yang dideteksi, pengujian lokasi objek MP, pengujian eliminasi noise getar, pengujian deteksi objek MP berdasarkan dengan perubahan ketinggian.

1.2. Pelaksanaan Pengujian

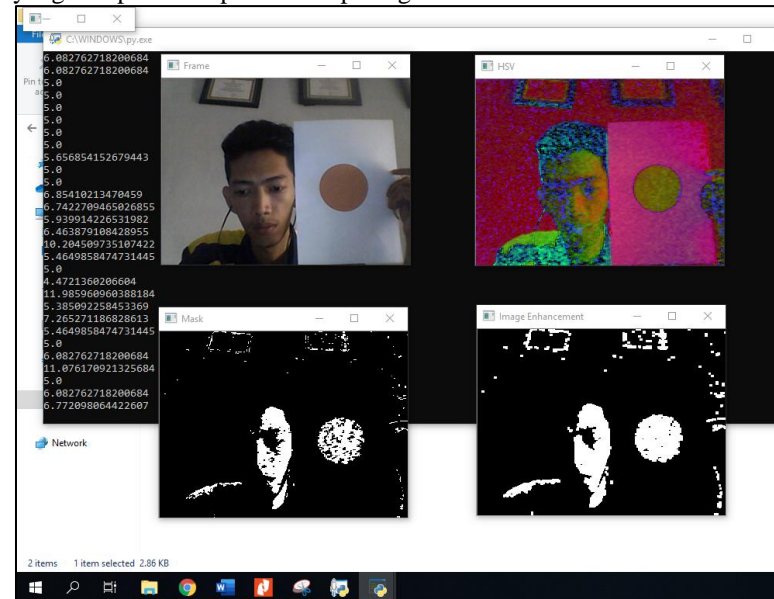
Pengujian dilakukan pada cuaca terik karena mencari rentan noise cahaya. Eliminasi noise getar dilakukan oleh system deteksi. Dalam keadaan bergerak ataupun diam noise getar bisa dieliminasi langsung oleh system deteksi. Jadi tidak dilakukan pencegahan getar karna system deteksi sudah mampu mengeliminasi noise getar tersebut. Sama dengan halnya noise getar, noise pencahayaan dapat di eliminasi oleh system deteksi. Pada pengujian yang dilakukan oleh penulis berfokus pada hasil dari deteksi objek MP dan blob deteksi.

Untuk mendapatkan hasil dan analisa maka perlu dilakukan beberapa pengujian. Pengujian yang dilakukan yaitu dengan meletakkan objek MP di lapangan dengan warna orange. Sample yang diambil adalah deteksi warna orange dari rentan 3 m sampai 10 m. untuk filter objeknya peneliti, peneliti membuat sebuah control *trackbar* dengan menggukan pemrograman C. control tersebut terdiri dari 6 buah. *Trackbar* yaitu HLow (Low Hue), HHigh (High Hue), Slow (Low Saturation), SHigh (High Saturation), VLow (Low Value), VHigh (High Value). *Trackbar* nilai *hue saturation* dan *value* terbagi menjadi dua yaitu *High* dan *Low* untuk membuat rentang nilai. *Trackbar* HHigh sampai VLow bernilai 0 – 255. Berikut tampilan control *trackbar* untuk filter objek.



Gambar 4. 1 Trackbar Untuk Filter Objek

Pada system ini, objek yang telah difilter dan terdeteksi akan berwarna putih sedangkan warna lain disekitarnya akan berwarna hitam. Warna hitam dan putih adalah hasil dari proses binerisasi. Saat pertama merunningkan program, tampilan trackbar yang muncul akan sama seperti gambar 4.1 dengan tampilan masih di posisi Low bernilai 0 dan High bernilai 255. Setelah dilakukan filter terhadap warna orange, hasil yang didapatkan dapat di lihat pada gambar :



Gambar 4. 2 Hasil Dari Trackbar

1.3. Scenario Pengujian

Pengujian terakhir adalah pengujian sisitem secara keseluruhan dari awal hingga akhir, dimana pengujian dilakukan dengan menjalankan system secara keseluruhan. Kemudian meletakkan abjek MP pada arena point kedua kemudian drone diterbangkan secara aonomous sesuai waypoint yang sudah ditentukan oleh GCS. Drone memasuki waypoint kedua dimana objek berada dan melakukan preoses

deteksi, setelah proses deteksi berhasil drone terbang ke waypoint berikutnya untuk misi dropping. Sebelum drone melakukan dropping MP drone diminta untuk mendeteksi lokasi dropping untuk menentukan titik tengah, setelah titik tengah sudah didaapat drone menjatuhkan MP ke lokasi dropping. Kemudian drone melakukan misi berikutnya sesuai arah waypoint. Brikut gambar waypoint GCS:



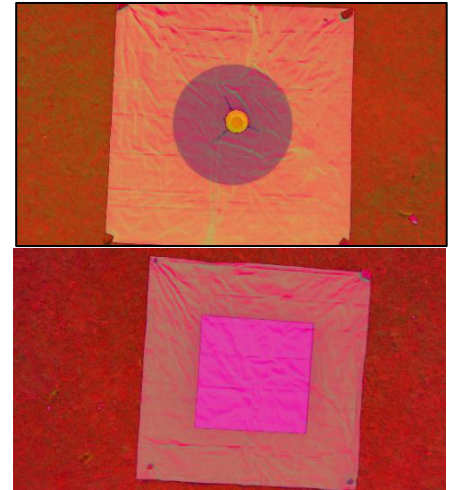
Gambar 4. 3 Titik Waypoint Misi

1.4. Pengujian Tiap Algoritma

Untuk mempermudah dalam pengambilan data-data, maka perlu dilakukan pengujian. Pengujian ini akan di lakukan uji treading koneversi warna HSV dan BLOB yang di rancang.

4.4.1 Filter warna / Konversi warna HSV

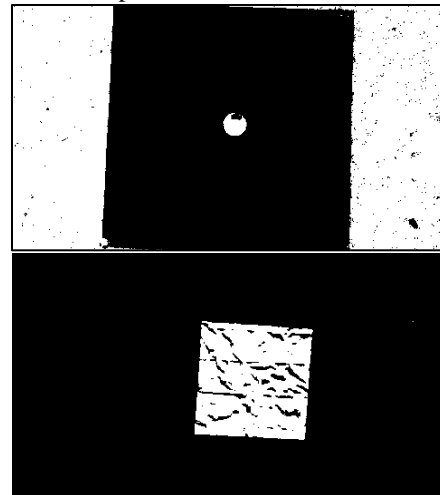
Kontes Robot Terbang Indonesia selalu dilakukan di luar ruangan. Dengan demikian, pencahayaan tidak bisa diprediksi. Untuk mengatasi masalah ini, digunakan ruang warna HSV agar dapat memisahkan antara warna dan pencahayaan, berikut hasil dari HSV:



Gambar 4. 4 Konversi warna HSV

4.4.2 Binary Thresholding / Color Filtering

Untuk setiap warna yang dibutuhkan, dilakukan binary thresholding dimana jika pixel pada citra berada pada rentang yang ditentukan adalah 10 - 200, maka kondisi tersebut dicatat dengan nilai 1 pada citra hasil. Untuk setiap cell dalam pixel warna hanya diperbolehkan untuk diisi oleh satu warna saja. Proses thresholding ini dilakukan dalam satu kali jalan (single pass) untuk setiap citra kamera.

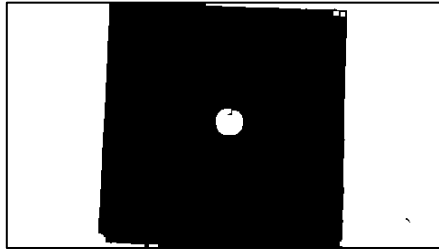


Gambar 4. 5 Color Filtering

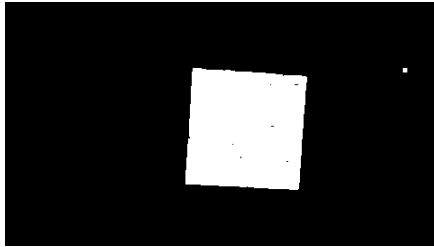
4.4.3 Image Enhancement

Pada proses image Enhancement adalah memperbaiki kualitas citra yang masih belum sempurna pada proses sebelumnya. image enhancement dilakukan dengan memodifikasi citra

masukannya agar sesuai dengan karakteristik yang



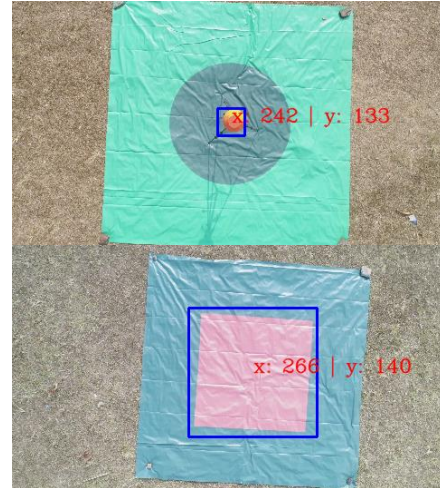
diharapkan.



Gambar 4. 6 Image Enhancement

4.4.4 Blob Detection

Proses ini merupakan proses mendeteksi objek yang berbentuk cenderung lingkaran berdasarkan hasil pada proses sebelumnya. ditujukan untuk mendeteksi daerah dalam gambar digital yang berbeda, seperti kecerahan atau warna, dibandingkan dengan daerah sekitarnya. semua poin dalam gumpalan dapat dianggap dalam beberapa hal mirip satu sama lain.





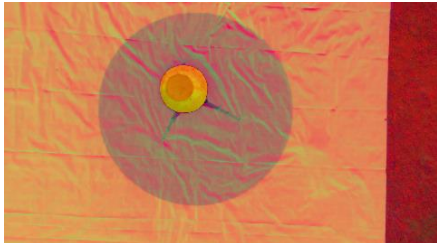
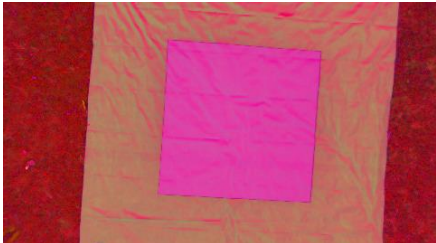

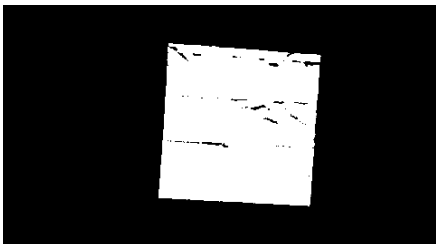
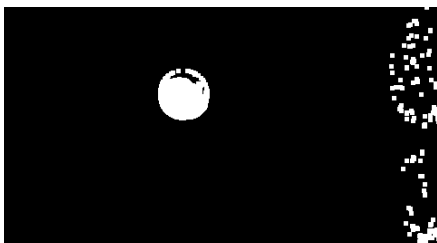

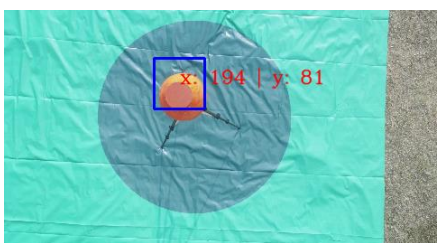
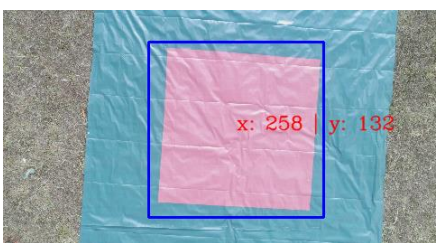
Gambar 4. 7 Blob Detection

1.5. Pengujian Deteksi Objek

Pengujian deteksi objek dilakukan secara manual untuk melihat apakah proses deteksi berjalan sesuai yang diharapkan, pengujian dilakukan berdasarkan ketinggian dari 3 meter hingga 10 meter keatas untuk melihat apakah objek masih bisa dideteksi oleh system. Berikut hasil pengujian dari beberapa ketinggian



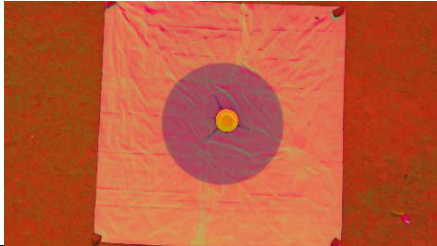
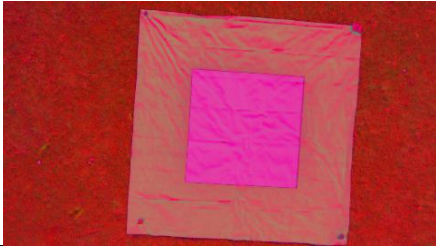
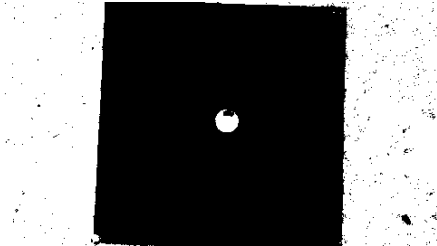
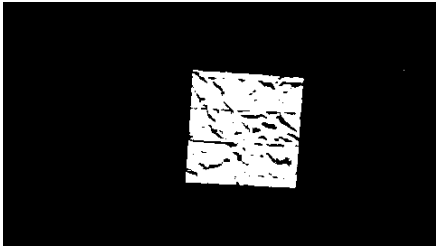
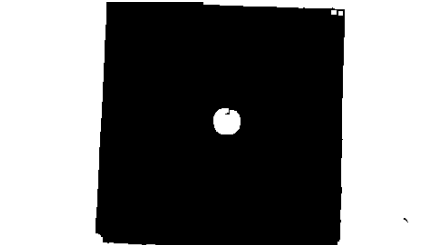
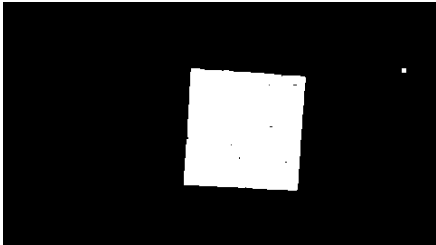
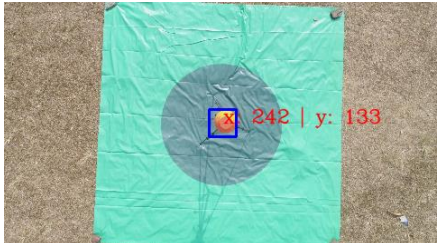
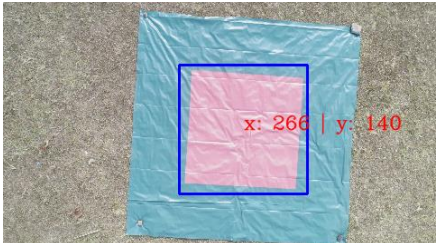
1.5.1. Pengujian ketinggian 3 Meter

Tabel 4. 1 Pengujian Ketinggian 3 Meter

No	Pengujian	Hasil Gambar MP	Hasil Gambar Dropping
1	Citra Asli		
2	Hasil HSV		
3	Hasil Binary Thersholding		
4	Hasil Image Enhancement		
5	Hasil BLOB Detektion		




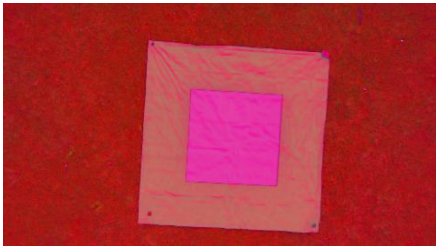
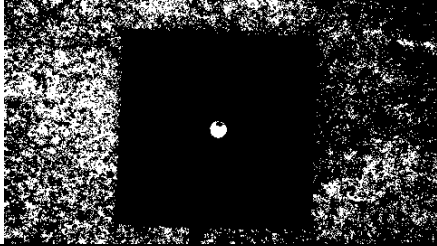
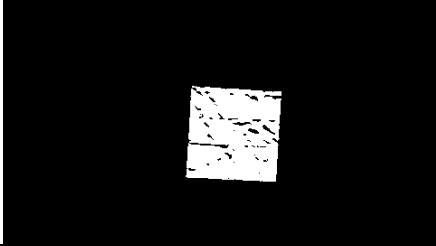
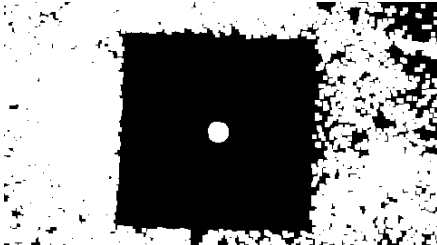
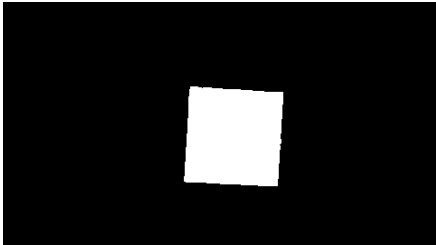

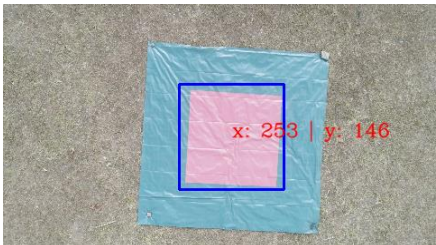
1.5.2. Pengujian Ketinggian 4 Meter

Tabel 4. 2 Pengujian Ketinggian 4 Meter

No	Pengujian	Hasil Gambar MP	Hasil Gambar Dropping
1	Citra Asli		
2	Hasil HSV		
3	Hasil Binary Thersholding		
4	Hasil Image Enhancement		
5	Hasil BLOB Detektion		

1.5.3. Pengujian Ketinggian 5 Meter

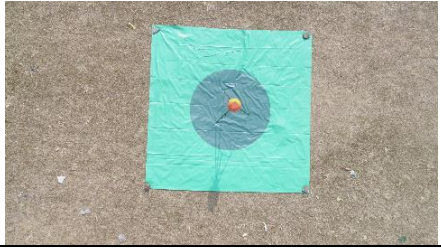


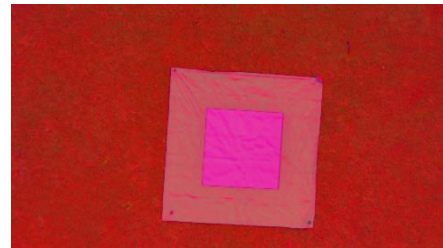
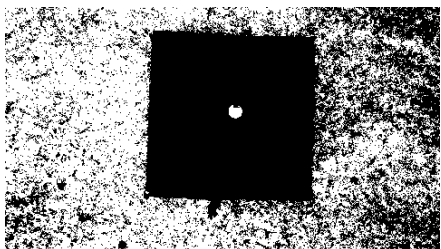

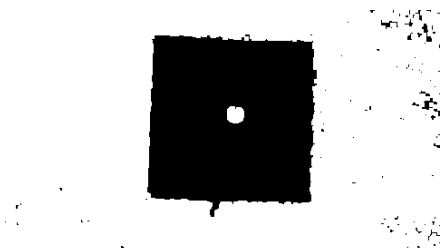
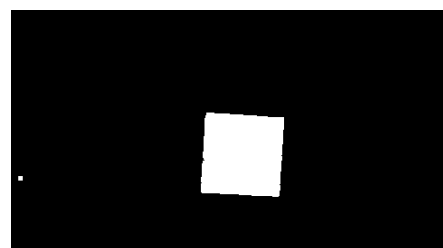
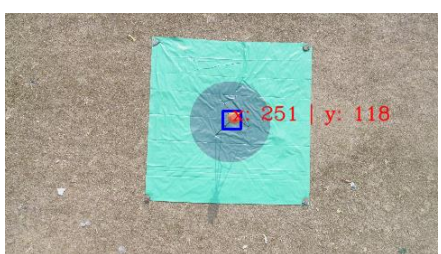
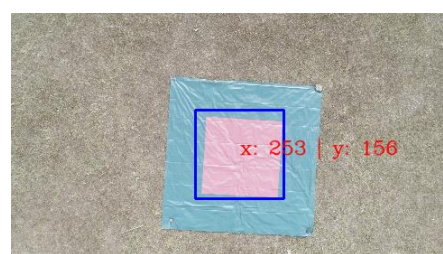
Tabel 4. 3 Pengujian Ketinggian 5 Meter

No	Pengujian	Hasil Gambar MP	Hasil Gambar Dropping
1	Citra Asli		
2	Hasil HSV		
3	Hasil Binary Thersholding		
4	Hasil Image Enhancement		
5	Hasil BLOB Detektion		

1.5.4. Pengujian Ketinggian 6 Meter




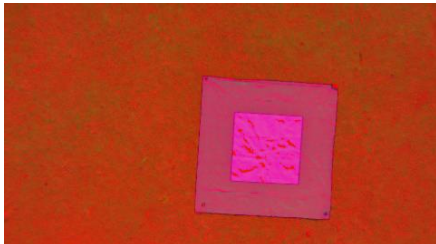
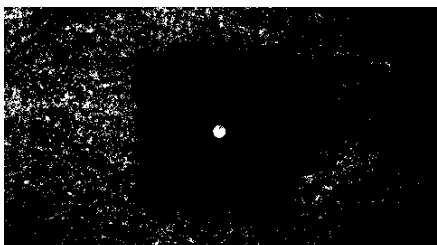

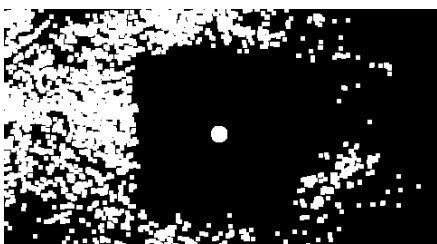
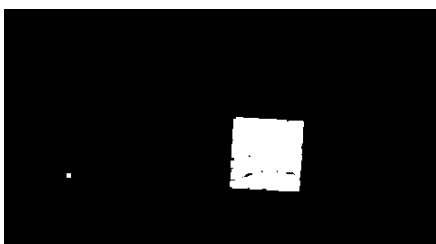
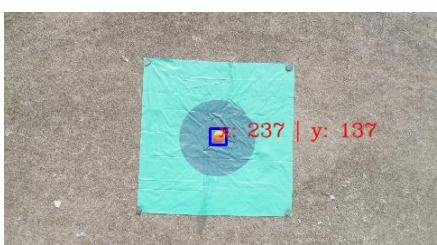
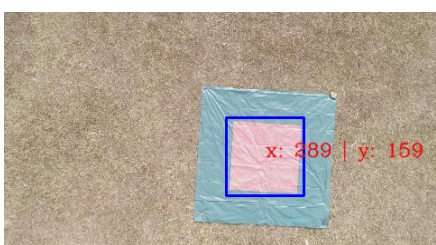
Tabel 4. 4 Pengujian Ketinggian 6 Meter

No	Pengujian	Hasil Gambar MP	Hasil Gambar Dropping
----	-----------	-----------------	-----------------------

1	Citra Asli		
2	Hasil HSV		
3	Hasil Binary Thersholding		
4	Hasil Image Enhancement		
5	Hasil BLOB Detektion		




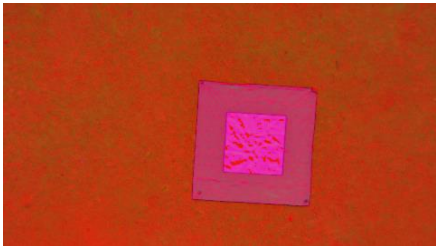
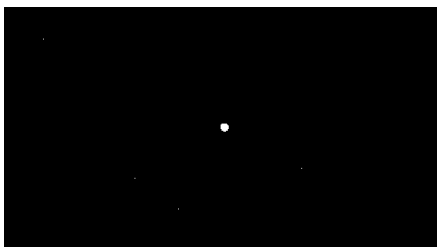

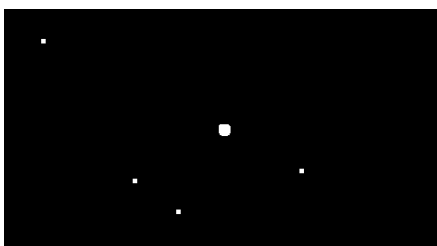
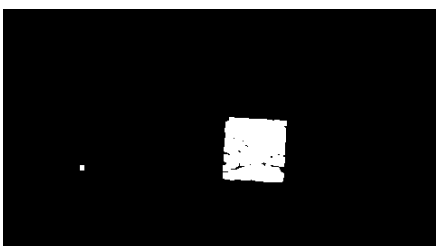
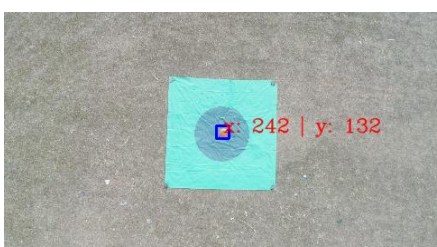
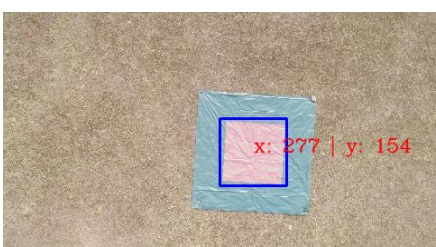
1.5.5. Pengujian Ketinggian 7 Meter

Tabel 4. 5 Pengujian Ketinggian 7 Meter

No	Pengujian	Hasil Gambar MP	Hasil Gambar Dropping
1	Citra Asli		
2	Hasil HSV		
3	Hasil Binary Thersholding		
4	Hasil Image Enhancement		
5	Hasil BLOB Detektion		



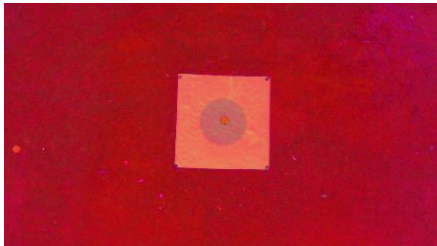
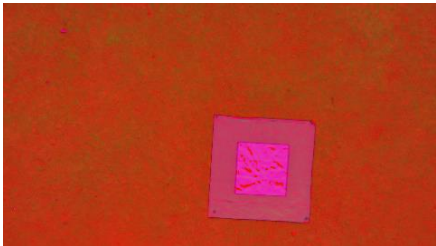
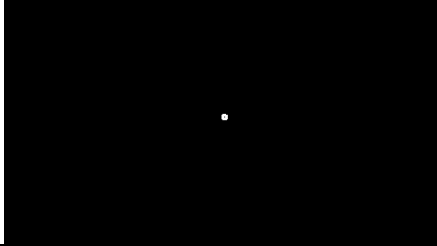
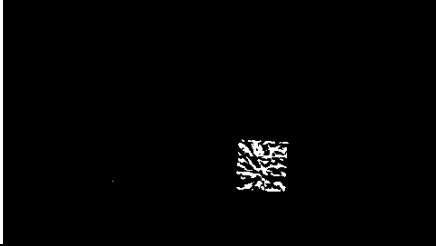
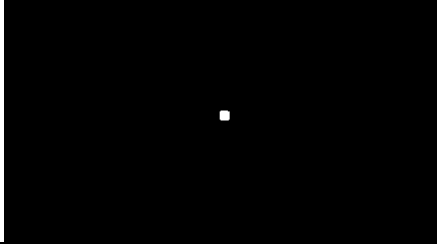
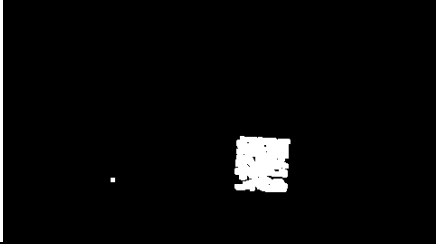
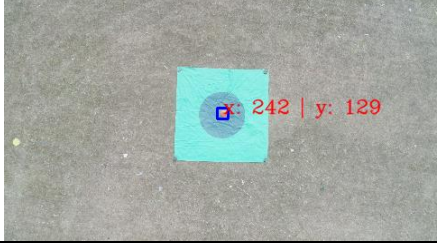
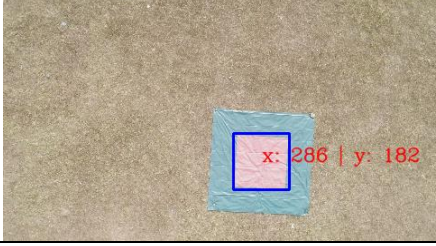
1.5.6. Pengujian Ketinggian 8 Meter

Tabel 4. 6 Pengujian Ketinggian 8 Meter

No	Pengujian	Hasil Gambar MP	Hasil Gambar Dropping
1	Citra Asli		
2	Hasil HSV		
3	Hasil Binary Thersholding		
4	Hasil Image Enhancement		
5	Hasil BLOB Detektion		


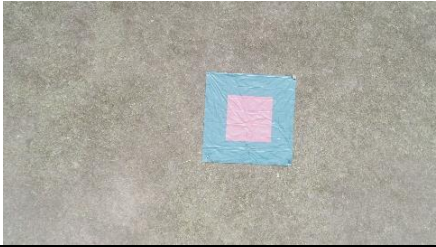
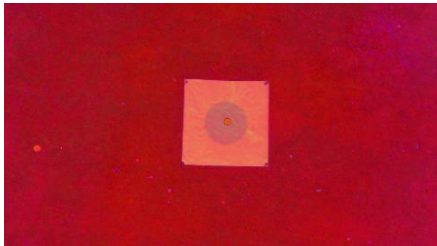
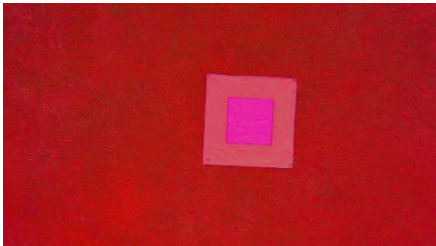
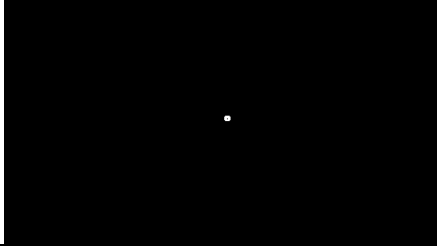

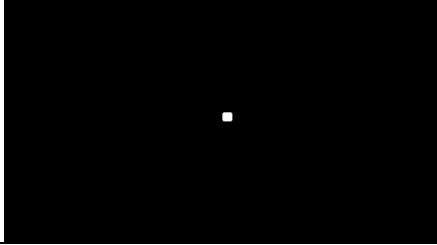
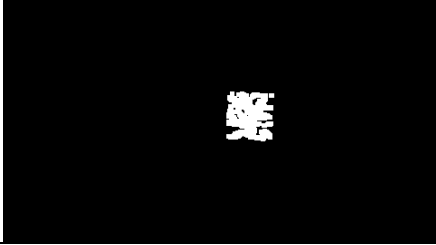
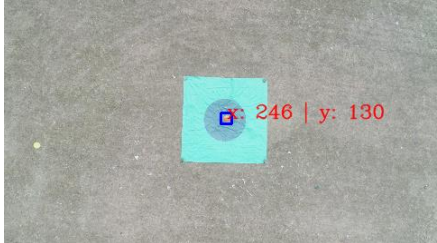
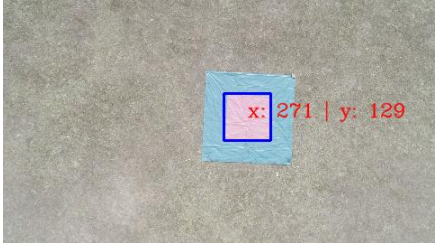
1.5.7. Pengujian Ketinggiain 9 Meter

Tabel 4. 7 Pengujian Ketinggian 9 Meter

No	Pengujian	Hasil Gambar MP	Hasil Gambar Dropping
1	Citra Asli		
2	Hasil HSV		
3	Hasil Binary Thersholding		
4	Hasil Image Enhancement		
5	Hasil BLOB Detektion		

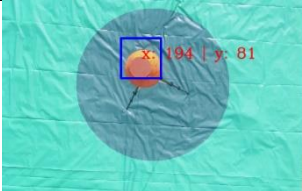
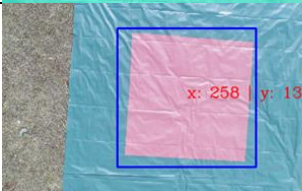

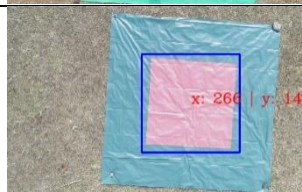

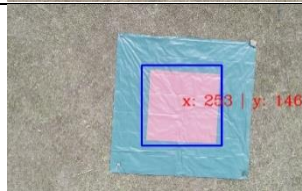


1.5.8. Pengujian Ketinggiain 10 Meter

Tabel 4. 8 Pengujian ketinggian 10 Meter


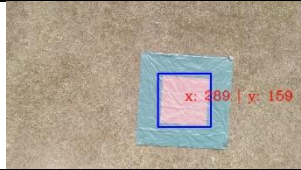
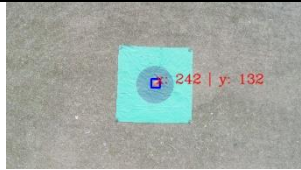
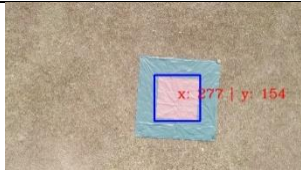

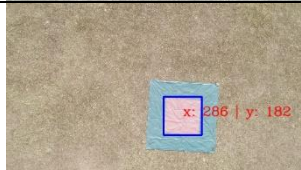
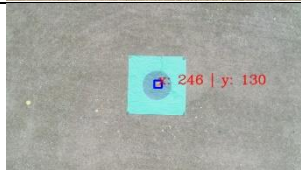
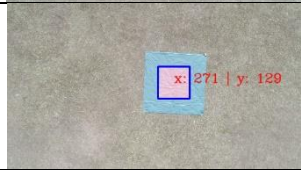


No	Pengujian	Hasil Gambar MP	Hasil Gambar Dropping
1	Citra Asli		
2	Hasil HSV		
3	Hasil Binary Thersholding		
4	Hasil Image Enhancement		
5	Hasil BLOB Detektion		

1.6. Rangkuman Penelitian

Tabel 4. 9 Hasil Pengujian Sistem Deteksi Warna Objek

No	Hasil BLOB	Jarak (m)	Objek	Deteksi	Status
1		3	MP	Terdeteksi	Berhasil
			Dropping	Terdeteksi	Berhasil
2		4	MP	Terdeteksi	Berhasil
			Dropping	Terdeteksi	Berhasil
3		5	MP	Terdeteksi	Berhasil
			Dropping	Terdeteksi	Berhasil
4		6	MP	Terdeteksi	Berhasil
			Dropping	Terdeteksi	Berhasil

No	Hasil BLOB	Jarak (m)	Objek	Deteksi	Status
----	------------	-----------	-------	---------	--------

5		7	MP	Terdeteksi	Berhasil
			Dropping	Terdeteksi	Berhasil
6		8	MP	Terdeteksi	Berhasil
			Dropping	Terdeteksi	Berhasil
7		9	MP	Terdeteksi	Berhasil
			Dropping	Terdeteksi	Berhasil
8		10	MP	Terdeteksi	Berhasil
			Dropping	Terdeteksi	Berhasil
9		11	MP	Tidak Terdeteksi	Tidak Berhasil
			Dropping	Terdeteksi	Berhasil

1.7. Pembahasan Pengujian Deteksi

Pengujian dilakukan pada siang hari, pada tahapan pengujian ini noise getar tidak berpengaruh dalam proses pendeteksian objek dan noise cahaya masih jelas tertangkap oleh proses deteksi, dan diperbaiki dengan adanya *Image Enhancement*. dengan demikian masalah pada noise getar dan pancahaya telah di minimalisir oleh system deteksi.

Untuk meminimalisir getaran pada saat pengambilan gambar, camera di berikan damper untuk meredam getaran pada saat pengambilan data gambar. Damper yang digunakan untuk meredam getaran camera dapat dilihat pada gambar 4. 8 Damper.



Gambar 4. 8 Damper

Berdasarkan pengujian dari ketinggian menunjukkan hasil yang cukup baik. Algoritma masih mampu mendeteksi objek yang dituju dengan baik meskipun berada pada kondisi pencahayaan yang berbeda. Pencahayaan ketika pengujian pada siang hari rentan waktu jam 10.00 - 14.00 WIB cuaca terik membuat proses color filtering rentang terhadap noise. Dapat dilihat pada gambar 4.5 Color Filtering (33), bagian yang terkena sinar matahari memantulkan cahaya putih (mengkilat) pada saat proses color filtering dijalankan menghasilkan noise ruang kosong yang dihasilkan, untuk meminimalisir digunakan *image enhancement*. Dan menghasilkan gambar seperti pada gambar 4.6 *image enhancement* (34). Pengujian pada ketinggian 3-10 meter objek MP dan lokasi dropping terdapat perubahan kontras warna

diakibatkan oleh cahaya matahari dan masih terdeteksi sebagai objek, karena dapat mengeliminasi noise.

4. Kesimpulan dan saran

1.1. Kesimpulan

Berdasarkan uraian yang telah dijelaskan pada BAB I sampai BAB IV, dapat ditarik kesimpulan sebagai berikut :

1. Penelitian ini sudah berhasil mengimplementasikan HSV dan implementasi algoritma *Blob Detection* dengan tambahan algoritma perbaikan citra atau *image enhancement*.
2. Algoritma lain yang digunakan adalah *color filtering* untuk proses filter warna dan mengeliminasi semua warna lain selain warna objek yang ingin dideteksi (orange). Selanjutnya adalah proses *binary thresholding* untuk merubah citra menjadi dua warna yaitu hitam dan putih dimana setiap warna pada rentang orange akan dipetakan menjadi warna putih, selain itu akan dipetakan menjadi warna hitam. Proses *erosi* dan *dilasi* untuk memperbaiki citra hasil *binary thresholding*. Terakhir adalah proses *confidence scoring* untuk memastikan hanya ada satu objek yang dideteksi dengan cara melakukan ranking berdasarkan ukuran semua objek yang dideteksi kemudian hanya mengambil satu objek dengan ukuran paling besar.
3. Penelitian ini efektif digunakan untuk KRTI 2018. Pada pengujian pertama diberikan noise dari objek lain dengan warna serupa, algoritma masih mampu memilih satu objek dengan benar. Berdasarkan hasil pengujian algoritma kedua masih mampu mendeteksi objek hingga jarak 10 meter. Terakhir diberikan noise berupa getaran, dan algoritma juga masih mampu mendeteksi objek.

1.2. Saran

1. Proses confidence scoring pada penelitian ini hanya berdasarkan ukuran objek. Jika terdeteksi lebih dari satu objek, maka hanya akan dipilih objek dengan ukuran terbesar serta memiliki ukuran pada rentang 0.75 - 0.87. Untuk penelitian selanjutnya disarankan menambahkan parameter lain selain ukuran.
2. Penelitian ini hanya mampu mendeteksi objek dengan jarak maksimal 10 meter. Untuk penelitian selanjutnya diharapkan dapat memperjauh jangkauan deteksi.

DAFTAR PUSTAKA

- Deswal, M. and Sharma, N. (2014) 'A Simplified Review on Fast HSV Image Color and Texture Detection and Image Conversion Algorithm', 3(5), pp. 1216–1222.
- Dhiemas, R. Y. . (2011) 'Pencarian Posisi Robot: Studi Kasus Pencarian Sumber Kebocoran Gas', *Fakultas Ilmu Komputer, Universitas Indonesia*.
- Kusumanto, R. D. *et al.* (2011) 'Klasifikasi Warna Menggunakan Pengolahan Model Warna HSV', 2(2), pp. 83–87.
- Lintz, J. J. and Simonett (1976) 'Remote Sensing and Environment', *Addison Wesley Publishing Company London*.
- Nana, A., Ichwan, M. and Santika Putra, I. M. (2012) 'Segmentasi citra untuk deteksi objek warna pada aplikasi pengambilan bentuk citra rectangle', *Itenas*, pp. 1–10.
- Pamungkas, E. M., Sumbodo, B. A. A. and Candradewi, I. (2017) 'Sistem Pendeteksi dan Pelacakan Bola dengan Metode Hough Circle Transform, Blob Detection, dan Camshift Menggunakan AR.Drone', *IJEIS (Indonesian Journal of Electronics and Instrumentation Systems)*, 7(1), p. 1. doi: 10.22146/ijeis.15405.
- Prayitno, Y. . (2012) 'Rancang Bangun Aplikasi Pendeteksi Bentuk Dan Warna Benda Pada Mobile Robot Berbasis Webcam', *Jurnal stikom.edu*, 1.
- Stokman, H. (2006) 'Color Feature Detection THEO', *Color Image Processing: Emerging Applications*, 20065568, pp. 203–225. doi: 10.1201/9781420009781.ch9.
- Sudarisman, D. *et al.* (2015) 'Strategi Deteksi Bola pada Tim Krakatau', *Isrsc 2015*.
- A. Mulyanto, R. I. Borman, P. Prasetyawan, W. Jatmiko and P. Mursanto, "Real-Time Human Detection and Tracking Using Two Sequential Frames for Advanced Driver Assistance System," 2019 3rd International Conference on Informatics and Computational Sciences (ICICoS), Semarang, Indonesia, 2019, pp. 1-5, doi: 10.1109/ICICoS48119.2019.8982396.
- Mulyanto, Agus & Indra, Rohmat & Prasetyawan, Purwono & Sumarudin, Ahmad. (2020). Implementation 2D Lidar and Camera for detection object and distance based on RoS. JOIV : International Journal on Informatics Visualization. 4. 10.30630/joiv.4.4.466.
- Mulyanto, Agus & Indra, Rohmat & Prasetyawan, Purwono & Jatmiko, Wisnu & Mursanto, Petrus & Sinaga, Aprian. (2020). Indonesian Traffic Sign Recognition For Advanced Driver Assistent (ADAS) Using YOLOv4. 520-524. 10.1109/ISRITI51436.2020.9315368.
- Neneng and Fernando, Y. (2017) 'Klasifikasi Jenis Daging Berdasarkan Analisis Citra Tekstur Gray Level Co-Occurrence Matrices (Glcm) Dan Warna', *Seminar Nasional Sains dan Teknologi 2017*, (November), pp. 1–7.
- Neneng, Neneng & Adi, Kusworo & Isnanto, Rizal. (2016). Support Vector Machine Untuk Klasifikasi Citra Jenis Daging Berdasarkan Tekstur Menggunakan Ekstraksi Ciri Gray Level Co-Occurrence Matrices (GLCM). JURNAL SISTEM INFORMASI BISNIS. 6. 1. 10.21456/vol6iss1pp1-10.