

# Pengenalan Jenis Bunga Anggrek Menggunakan Metode Color Local Binary Pattern dan Support Vector Machine

Debby Meliani Prayogo, Kartika Gunadi, Endang Setyati

Program Studi Teknik Informatika Fakultas Teknologi Industri Universitas Kristen Petra

Jl. Siwalankerto 121 – 131 Surabaya 60236

Telp. (031) – 2983455, Fax. (031) – 8417658

E-Mail: debbymlni98@gmail.com, kgunadi@petra.ac.id, endang@stts.edu

## ABSTRAK

Bunga Anggrek merupakan suku tumbuhan berbunga dengan jenis terbanyak. Salah satunya adalah bunga anggrek bulan yang merupakan salah satu dari 3 (tiga) bunga nasional yang ada di Indonesia. Bunga anggrek dapat ditemui di taman kota dan banyak tempat wisata karena keindahan dari bunganya. Namun, masyarakat pastinya mengalami kesusahan dalam mengenali jenis dari bunga anggrek tersebut. Oleh karena itu, dibuatlah program yang dapat membantu dalam mengenali jenis bunga anggrek yang terdapat disekitar. Pengenalan bunga anggrek sudah pernah dilakukan untuk mengenali tekstur yang dimiliki oleh bunga anggrek. Sedangkan, penelitian ini menggunakan 25 jenis bunga anggrek yang berasal dari Indonesia untuk dikenali.

Metode *You Only Look Once* (YOLO) untuk mendeteksi objek bunga dalam gambar. Sebelum dilakukan klasifikasi terhadap bunga, background dari gambar dapat dihilangkan menggunakan *Image Segmentation*. Metode *Color Local Binary Pattern* digunakan untuk mendapatkan tekstur dari gambar melalui beberapa *color space*, yaitu *grayscale*, RGB, HSI, YIQ, dan *oRGB*. *Support Vector Machine* kemudian digunakan untuk mengenali jenis dari bunga anggrek.

Hasil dari penelitian ini dapat mengenali jenis bunga anggrek pada gambar. Hasil pengujian dari dataset peneliti menunjukkan akurasi sebesar 30.7% untuk *color space grayscale*, 37% untuk *color space RGB*, 34.6% untuk *color space HSI*, 41% untuk *color space YIQ*, dan 40.2% untuk *color space oRGB* dalam mengenali jenis bunga anggrek.

**Kata Kunci:** *Color Local Binary Pattern, Support Vector Machine, Image Segmentation, You Only Look Once*

## ABSTRACT

*Orchid flowers are the flowering plants with the most types or species. One of them is the moon orchid flower which is one of the three national flowers in Indonesia. Orchid flowers can be found in city parks and many tourist attractions because of its beauty. However, people will certainly have difficulty in recognizing the type of orchid. Therefore, a program is made to help people in identifying the types of orchids that are around. Orchid flower recognition has already been researched to recognize the texture of its flower. However, this study uses 25 species of orchids that is from Indonesia to be recognized.*

*You Only Look Once (YOLO) method is used for detecting flower objects in the image. Before classifying the orchid species, the background image need to be removed using Image Segmentation. The Color Local Binary Pattern descriptor is used*

*to get the texture of the image through several colorspace, namely grayscale, RGB, HSI, YIQ, and oRGB. Support Vector Machine is then used to recognize the type of orchid.*

*The result of this program can recognize the species of orchids in the picture. From the test results using the researcher's dataset show an accuracy of 30.7% using color space grayscale, 37% using color space RGB, 34.6% using color space HSI, 41% using color space YIQ, and 40.2% using color space oRGB in recognizing the species of orchid.*

**Keywords:** *Color Local Binary Pattern, Support Vector Machine, Image Segmentation, You Only Look Once*

## 1. PENDAHULUAN

Bunga anggrek merupakan suku tumbuhan berbunga dengan jenis terbanyak, yang tersebar luas di daerah tropika basah hingga wilayah sirkompular. Banyak dari jenis bunga anggrek di daerah iklim tropik hidup sebagai benalu yang hidup dengan menumpang pada tumbuhan lain. Sedangkan anggrek di daerah iklim sedang biasanya hidup di tanah dan beradaptasi menjadi umbi terhadap musim dingin.

Karakteristik utama dari bunga anggrek yang membedakannya dari spesies bunga lain terdapat pada bunga. Di bagian bawah bunga non-anggrek yang tidak terspesifikasi adalah bata pendukung, yang disebut pedicel. Tepat diatas dan dibawah pangkal bunga itu sendiri terdapat lingkaran hijau seperti daun, yang disebut sepal. Kelopak dan sepal dari bunga anggrek biasanya memiliki warna yang serupa dan dalam set 3 (tiga). Namun, bentuk yang dimiliki oleh kelopak dan sepal biasanya sangat berbeda dan karenanya mempertahankan identifikasi mereka yang berbeda [1].

Anggrek bulan merupakan salah satu dari 3 (tiga) bunga nasional yang ada di Indonesia, selain itu masih ada banyak jenis bunga anggrek yang terdapat di Indonesia. Bunga anggrek yang ada di Indonesia saat ini jumlahnya kurang lebih 300 jenis. Jumlah tersebut berasal dari jenis bunga anggrek yang sudah dikembangkan dengan optimal namun masih ada banyak jenis bunga anggrek yang belum dikembangkan dengan optimal.

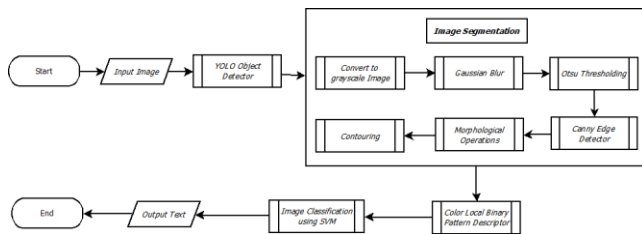
Saat ini, kebanyakan masyarakat di Indonesia mengenali dan menyukai bunga anggrek yang banyak terdapat di taman kota atau tempat wisata. Hal itu dikarenakan keindahan dari bunga anggrek yang bermacam-macam jenisnya. Dengan banyaknya jumlah jenis bunga anggrek yang terdapat di Indonesia, tentu saja masyarakat akan mengalami kesusahan dalam mengenali jenis dari bunga anggrek tersebut. Oleh karena itu, tenaga ahli dibutuhkan untuk mengenali jenis bunga anggrek. Tenaga ahli tersebut dapat

digantikan dengan sebuah program yang dapat mengenali jenis bunga anggrek.

Agar sistem dapat mengenali jenis bunga yang terdapat pada gambar, sistem harus mampu menentukan letak bunga dan mengenali jenis bunga yang terdapat pada citra. Penentuan letak bunga dapat menggunakan beberapa metode yaitu, object detection using Hog Features, region-based convolutional neural network (RCNN) dan You Only Look Once (YOLO). Sedangkan, pengenalan jenis bunga dapat dilakukan menggunakan visual descriptor, salah satunya adalah Color Local Binary Pattern (Color LBP). Color LBP merupakan adaptasi dari Local Binary Pattern yang merupakan texture descriptor dengan menambahkan warna dari citra yang digunakan [2]. Setelah itu, dilakukan klasifikasi menggunakan Support Vector Machine (SVM).

Hingga saat ini, sudah dilakukan banyak riset mengenai pengenalan jenis bunga. Salah satunya menggunakan metode linear binary SVM yang dilatih untuk mengenali karakter visual dari bunga [3]. Penelitian ini menggunakan beberapa jenis bunga anggrek untuk dikenali melalui tekstur yang dimiliki oleh bunga tersebut. Hasil yang didapatkan memiliki nilai akurasi yang tinggi dalam mengenali tekstur dari bunga anggrek.

Tujuan dari penelitian kali ini adalah untuk membuat program yang dapat mengenali jenis bunga anggrek yang terdapat di Indonesia. Diharapkan melalui program ini dapat membantu masyarakat Indonesia dalam mengenali jenis bunga anggrek yang ada. Dikarenakan bunga anggrek tidak hanya digunakan sebagai tanaman hias saja, namun dapat digunakan sebagai obat dan dapat dikonsumsi.



Gambar 1. Arsitektur Sistem

## 2. DASAR TEORI

### 2.1 Bunga Anggrek

Anggrek (family Orchidaceae), jumlahnya hampir 1.000 genera dan lebih dari 22.000 spesies yang menarik tersebar diseluruh dunia, terutama di daerah tropis basah. Banyak dari jenis bunga anggrek di daerah iklim tropik hidup sebagai benalu yang hidup dengan menumpang pada tumbuhan lain. Sedangkan anggrek di daerah iklim sedang biasanya hidup di tanah dan beradaptasi menjadi umbi terhadap musim dingin.

Karakteristik utama dari bunga anggrek yang membedakannya dari spesies bunga lain terdapat pada bunga. Di bagian bawah bunga non-anggrek yang tidak terspesifikasi adalah bata pendukung, yang disebut pedicel. Tepat diatas dan dibawah pangkal bunga itu sendiri terdapat lingkaran hijau seperti daun, yang disebut sepal. Kelopak dan sepal dari bunga anggrek biasanya memiliki warna yang serupa dan dalam set 3 (tiga). Namun, bentuk yang dimiliki oleh kelopak dan sepal biasanya sangat berbeda dan karenanya mempertahankan identifikasi mereka yang berbeda [1].

### 2.2 YOLO

YOLO merupakan sebuah pendekatan baru pada deteksi objek dengan mbingkai objek sebagai masalah regresi untuk kotak pembatas yang terpisah secara spasial dan probabilitas terkait. YOLO memiliki kecepatan 45 frames per second dan merupakan single network yang dapat dioptimalkan langsung pada performa deteksi. YOLO dapat memprediksi beberapa bounding box dan probabilitas kelas untuk box tersebut [4].

YOLOv3 memiliki empat langkah dalam melakukan object detection [5] :

- Bounding Box Prediction
- Class Prediction
- Prediction Across Scales
- Feature Extractor

### 2.3 Image Segmentation

Dalam Computer Vision, segmentasi citra adalah proses mempartisi citra digital menjadi beberapa segmen. Image segmentation biasanya digunakan untuk menemukan objek dan batas dalam sebuah gambar [6].

#### 2.3.1 Gaussian Blur

Gaussian Blur atau biasa disebut juga dengan Gaussian Smoothing adalah hasil dari mengaburkan suatu gambar menggunakan fungsi Gaussian. Gaussian Blur biasa digunakan pada software grafik untuk mengurangi noise dan detail. Gaussian Smoothing biasa digunakan pada tahap pre-processing dalam algoritma computer vision untuk meningkatkan struktur gambar pada skala yang berbeda [7].

#### 2.3.2 Otsu Thresholding

Metode Otsu merupakan metode nonparametric dan unsupervised dari automatic threshold selection untuk segmentasi gambar. Threshold yang optimal akan dipilih oleh kriteria diskriminan untuk memaksimalkan pemisahan hasil kelas pada gray level. Prosedurnya sangat sederhana dengan hanya memanfaatkan nol dan momen kumulatif orde pertama dari gray-level histogram [8]. Hasil dari metode Otsu mengembalikan threshold tunggal yang memisahkan piksel menjadi 2 kelas, foreground dan background.

#### 2.3.3 Canny Edge Detector

Canny Edge Detector merupakan algoritma pendeteksi tepi yang popular. Proses deteksi tepi berfungsi untuk menyederhanakan analisis gambar dengan mengurangi jumlah data yang akan diproses secara drastik, sementara pada saat yang sama menjaga informasi structural yang berguna tentang batas-batas objek. Persamaan yang dimiliki dari berbagai pendeteksi tepi adalah persyaratan yang menghasilkan masalah deteksi tepi abstrak [9].

Metode Canny Edge Detector memiliki empat proses yang dilakukan, yaitu [10]:

- Noise Reduction menggunakan Gaussian Smoothing
- Finding the intensity gradient of the image
- Non-maximum Suppression
- Hysteresis Thresholding

#### 2.3.4 Morphological Operations

Morfologi adalah operasi pemrosesan gambar yang memproses gambar berdasarkan bentuk. Dalam operasi morfologi, nilai setiap

piksel dalam gambar output didasarkan pada perbandingan piksel yang sesuai dengan gambar input dengan tetangganya [10].

Operasi morfologi awalnya dikembangkan untuk gambar biner, lalu diperluas untuk gambar *grayscale*. Teknik morfologi menyelidiki gambar dengan bentuk atau *template* kecil yang disebut elemen *structure*. Element *structure* ini ditempatkan di semua lokasi yang mungkin dalam gambar kemudian dibandingkan dengan lingkungan piksel yang sesuai. Operasi morfologi yang paling dasar adalah *dilation* dan *erosion*. Terdapat dua cara penggunaan dari operasi morfologi ini, yaitu *closing* dan *opening*. *Closing* merupakan hasil *erosion* dari *dilation*, sedangkan *opening* adalah hasil *dilation* dari *erosion* [11].

## 2.4 Contours

*Contour* merupakan sebuah kurva yang menghubungkan semua titik kontinu (sepanjang batas), memiliki warna atau intensitas yang sama. *Contour* merupakan alat yang berguna untuk analisis bentuk dan deteksi dan pengenalan objek. Kegunaan dari *contour* adalah untuk memodifikasi gambar [10].

## 2.5 Circularity

*Circularity* merupakan salah satu *shape descriptor* dasar yang dapat digunakan dalam melakukan analisa gambar. *Circularity* digunakan untuk melihat seberapa bulat sebuah gambar dengan melakukan perhitungan relasi dari luas area dan keliling dari gambar [12].

## 2.6 Statistical Region Merging

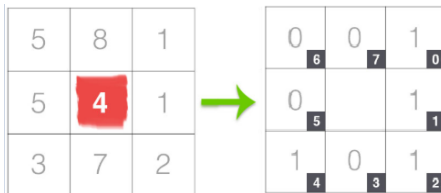
*Statistical Region Merging* (SRM) merupakan algoritma yang digunakan dalam melakukan segmentasi pada gambar. Dalam *image processing* algoritma ini digunakan untuk melakukan *grouping* pada pixel yang berdekatan atau tetangga berdasarkan bayangan warna yang berada pada *threshold* tertentu. *Statistical Region Merging* mampu mengatasi *noise corruption*, mengangani oklusi dan melakukan segmentasi *scale-sensitive segmentation* [13].

## 2.7 Color Space

*Color Space* adalah model warna (atau sistem warna), adalah model matematika abstrak yang hanya menggambarkan kisaran warna sebagai angka *tupple*, biasanya dibagi menjadi 3 atau 4 komponen warna, contohnya RGB. Setiap warna dalam suatu sistem mewakili satu titik. *Color Space* memiliki bermacam-macam jenis, seperti *grayscale*, RGB, HSI, YIQ, dan oRGB [7].

## 2.8 Local Binary Pattern

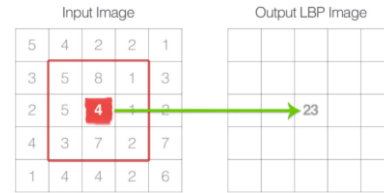
*Local Binary Pattern* merupakan operator tekstur yang sederhana namun efisien dengan memberikan label pada setiap *pixel* dari gambar dan mengubahnya menjadi angka biner. Terdapat banyak *research* dilakukan menggunakan metode LBP dalam melakukan pengenalan objek atau teks melalui fitur tertentu [14].



Gambar 2. Hasil Perhitungan *Local Binary Pattern*

Gambar 2 adalah hasil perhitungan menggunakan *Local Binary Pattern*. Dengan cara merubah nilai *pixel* yang lebih besar atau

sama dengan dari piksel tengah menjadi 1, jika tidak menjadi 0 [14]. Gambar 3 adalah penempatan lokasi hasil perhitungan dari Gambar 4 dengan nilai biner yang telah diubah menjadi desimal.



Gambar 3. Penempatan lokasi hasil perhitungan

## 2.9 Support Vector Machine

*Support Vector Machine* (SVM) merupakan model pembelajaran yang diawasi dengan algoritma yang menganalisis data yang digunakan untuk klasifikasi dan analisis regresi. Model SVM adalah representasi dari contoh-contoh kategori sebagai titik-titik dalam ruang, dipetakan sehingga kategori dipisahkan oleh celah yang jelas yang selebar mungkin. Klasifikasi model SVM dibantu oleh beberapa kernel yang berguna untuk mengubah *data input* dan mengubahnya sesuai format yang dibutuhkan. Kernel yang dapat digunakan oleh SVM antara lain, *linear*, *non-linear*, *polynomial*, *radial basis function* (RBF), dan *sigmoid* [15].

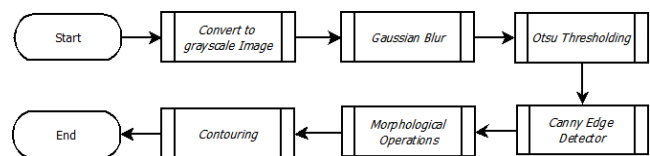
## 3. DESAIN SISTEM

### 3.1 YOLO Object Detector

*YOLO Object Detector* digunakan untuk mendeteksi keberadaan bunga pada gambar. Pada proses ini, objek yang dideteksi adalah bunga angrek dan bunga selain angrek. Dikarenakan YOLO tidak dapat mengenali bunga dengan sendirinya, maka diperlukan *training* pada *dataset* terlebih dahulu. *Training* untuk YOLO ini membutuhkan data gambar berupa JPG dan juga memberi label pada setiap objek gambar yang terdapat pada *dataset*. Hasil dari *training* ini berupa *weights* yang kemudian digunakan untuk mendeteksi bunga yang terdapat pada gambar saat *testing*.

### 3.2 Proses Image Segmentation

*Image Segmentation* sangat diperlukan pada proses pengenalan objek. Pada penelitian kali ini dilakukan percobaan melakukan segmentasi dengan dua cara, yaitu Proses Segmentasi Tepi dan Proses Segmentasi *Statistical Region Merging*. Proses Segmentasi Tepi melingkupi konversi gambar menjadi *grayscale image*, pemberian *Gaussian Blur*, *Otsu Thresholding*, *Canny Edge Detector* dan *Morphological Operations*. Gambar 4 adalah *flowchart* bagaimana proses segmentasi tepi terhadap bunga dilakukan.



Gambar 4. Flowchart dari proses *Image Segmentation*

### 3.3 Color Local Binary Pattern

Hasil gambar bunga yang didapat setelah dilakukan *image segmentation*, harus terlebih dahulu di deskripsikan sebelum dilakukan klasifikasi pada gambar tersebut. Deskripsi gambar dilakukan menggunakan metode *Color Local Binary Pattern* untuk

mendapatkan tekstur dari bunga tersebut. *Color Local Binary Pattern* kemudian diaplikasikan kepada 3 *channel* warna yang terdapat pada gambar berwarna.

### 3.4 Image Classification menggunakan Support Vector Machine

Klasifikasi digunakan untuk mengetahui jenis dari bunga anggrek yang terdapat pada gambar. Metode yang digunakan merupakan *Linear Support Vector Machine*. Sebelum itu diperlukan *training* menggunakan *dataset training* bunga anggrek yang sama pada saat melakukan *training* untuk YOLO. Setelah dilakukan ekstraksi *features* dari penerapan *Color Local Binary Pattern*, dan pemberian label untuk setiap gambar, maka *training* sudah dapat dilakukan. *Training* dilakukan agar program dapat mengetahui perbedaan *texture* yang terdapat pada banyak kelas berbeda. *Training* dilakukan pada beberapa jenis *color space* yang berbeda-beda, yaitu *grayscale*, RGB, HSI, YIQ, dan oRGB. Saat melakukan *testing* atau validasi pada gambar *input*, perlu dilakukan hal yang sama seperti pada saat *training*.

## 4. HASIL EKSPERIMEN

### 4.1 Pengujian Terhadap Hasil Training

#### 4.1.1 Training YOLO

Model pertama, *training* YOLO dilakukan dengan *batch* sebesar 64 dan *subdivisions* sebesar 16, dengan *learning rate* sebesar 0.001. Dilakukan penggantian jumlah filter menjadi 21 pada 1 buah *convolutional layer* diatas masing-masing *layer* YOLO. Jumlah data yang digunakan sebanyak 625 gambar bunga anggrek dan 390 gambar bunga lainnya. Tabel 1 adalah perbandingan mAP dan IoU pada *training* YOLO model pertama.

**Tabel 1. Perbandingan mAP dan IoU pada training YOLO model pertama**

YOLO last weights	mAP (%)	IoU (%)
yolo-obj_last-13.weights	89.57	74.39
yolo-obj_last-14.weights	88.24	70.24
yolo-obj_last-15.weights	87.86	69.59
yolo-obj_last-16.weights	87.71	69.72
yolo-obj_last-17.weights	91.88	72.73
yolo-obj_last-18.weights	90.69	71.92
yolo-obj_last-19.weights	87.74	76.17

Model kedua, *training* YOLO dilakukan dengan *batch* sebesar 64 dan *subdivisions* sebesar 16, dengan *learning rate* sebesar 0.001. Dilakukan penggantian jumlah filter menjadi 21 pada 3 buah *convolutional layer* diatas masing-masing *layer* YOLO. Jumlah data yang digunakan sebanyak 625 gambar bunga anggrek dan 390 gambar bunga lainnya. Tabel 2 adalah perbandingan mAP dan IoU pada *training* YOLO model kedua.

**Tabel 2. Perbandingan mAP dan IoU pada training YOLO model kedua**

YOLO last weights	mAP (%)	IoU (%)
yolo-obj_last-14.weights	89.29	66.51
yolo-obj_last-15.weights	86.38	74.61
yolo-obj_last-16.weights	90.26	69.90
yolo-obj_last-17.weights	90.84	73.61
yolo-obj_last-18.weights	87.23	63.15
yolo-obj_last-19.weights	89.12	75.37
yolo-obj_last-20.weights	89.35	72.45

#### 4.1.2 Training Support Vector Machine

Model SVM yang digunakan memiliki C sebesar 100.0 dan *random state* sebesar 42. Hasil yang didapatkan dibagi menjadi 5 dengan *colorspace* yang berbeda-beda, yaitu *training* pada *Grayscale*, RGB, HSI, YIQ, dan oRGB.

##### 4.1.2.1 Training pada Grayscale

Pengujian didapatkan dari hasil deskripsi *Local Binary Pattern* yang dilakukan pada *channel grayscale*. Tabel 3 adalah perbandingan akurasi yang didapatkan terhadap *color space grayscale*.

**Tabel 3. Perbandingan hasil akurasi terhadap color space Grayscale**

YOLO weights	Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)	
	NP	R	NP	R	NP	R	NP	R	NP	R
	8	1	8	2	8	3	8	4	8	5
yolo-obj_last-13.weights	25.2		21.3		19.7		29.1		28.3	
yolo-obj_last-17.weights	16.9		17		22.3		24.6		24.6	
yolo-obj_last-18.weights	19.3		19.3		23		23.7		24.4	
yolo-obj_last-19.weights	18.8		18.8		23.9		23.9		22.2	
n_yolo-obj_last-19.weights	6.7		8.3		6.7		10		9.2	

##### 4.1.2.2 Training pada RGB

Pengujian didapatkan dari hasil deskripsi *Local Binary Pattern* yang dilakukan pada ketiga *channel* RGB. Tabel 4 adalah perbandingan akurasi yang didapatkan terhadap *color space* RGB.

**Tabel 4. Perbandingan hasil akurasi terhadap *color space* RGB**

YOLO weights	Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)	
	NP	R	NP	R	NP	R	NP	R	NP	R
	8	1	8	2	8	3	8	4	8	5
yolo-obj_last-13.weights	34.6		24.4		36.2		37		25.2	
yolo-obj_last-17.weights	30.8		25.4		36.1		28.5		30	
yolo-obj_last-18.weights	14.8		21.5		31.9		21.5		26	
yolo-obj_last-19.weights	18.8		29		33.3		33.3		32.5	
n_yolo-obj_last-19.weights	9.2		13.3		15		18.3		18.3	

**4.1.2.3 Training pada HSI**

Pengujian didapatkan dari hasil deskripsi *Local Binary Pattern* yang dilakukan pada ketiga *channel* HSI. Tabel 5 adalah perbandingan akurasi yang didapatkan terhadap *color space* HSI.

**Tabel 5. Perbandingan hasil akurasi terhadap *color space* HSI**

YOLO weights	Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)	
	NP	R	NP	R	NP	R	NP	R	NP	R
	8	1	8	2	8	3	8	4	8	5
yolo-obj_last-13.weights	22.8		18.9		29.1		23.6		29.9	
yolo-obj_last-17.weights	23.1		23.8		23.1		29.2		27.7	
yolo-obj_last-18.weights	25.2		23.7		21.5		27.4		26.7	
yolo-obj_last-19.weights	25.6		16.2		20.5		25.6		18	
n_yolo-obj_last-19.weights	10.8		13.3		18.3		15		13.3	

**4.1.2.4 Training pada YIQ**

Pengujian didapatkan dari hasil deskripsi *Local Binary Pattern* yang dilakukan pada ketiga *channel* YIQ. Tabel 6 adalah perbandingan akurasi yang didapatkan terhadap *color space* YIQ.

**Tabel 6. Perbandingan hasil akurasi terhadap *color space* YIQ**

YOLO weights	Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)	
	NP	R	NP	R	NP	R	NP	R	NP	R
	8	1	8	2	8	3	8	4	8	5
yolo-obj_last-13.weights	26		33.1		33.9		36.2		25.2	
yolo-obj_last-17.weights	31.5		32.3		33.1		33.1		40	
yolo-obj_last-18.weights	26		35.5		28.1		33.3		24.4	
yolo-obj_last-19.weights	30.8		36.8		31.6		41		36.8	
n_yolo-obj_last-19.weights	15.8		19.2		20		18.3		15.8	

**4.1.2.5 Training pada oRGB**

Pengujian didapatkan dari hasil deskripsi *Local Binary Pattern* yang dilakukan pada ketiga *channel* oRGB. Tabel 7 adalah perbandingan akurasi yang didapatkan terhadap *color space* oRGB.

**Tabel 7. Perbandingan hasil akurasi terhadap *color space* oRGB 1**

YOLO weights	Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)		Akurasi (%)	
	NP	R	NP	R	NP	R	NP	R	NP	R
	8	1	8	2	8	3	8	4	8	5
yolo-obj_last-13.weights	34.6		31.5		33.9		30		26.7	
yolo-obj_last-17.weights	32.3		25.4		36.9		27.7		32.3	
yolo-obj_last-18.weights	28.8		33.3		32.6		28.8		26.7	
yolo-obj_last-19.weights	28.2		36.7		32.5		32.5		29.1	
n_yolo-obj_last-19.weights	17.5		21.7		18.3		21.7		17.5	

## 4.2 Pengujian Terhadap Rangkaian Program

Gambar 5 adalah *input* gambar yang akan digunakan untuk melakukan pengenalan jenis dari bunga anggrek.



Gambar 5. Input gambar

### 4.2.1 YOLO

Metode YOLO digunakan untuk melakukan pemotongan pada gambar input untuk mengetahui posisi bunga yang terdapat pada gambar dan membedakan bunga anggrek dari bunga lain. Ketika posisi telah didapatkan maka akan mempermudah proses deteksi jenis bunga anggrek. Gambar 6 adalah hasil deteksi bunga anggrek yang terdapat pada gambar menggunakan metode YOLO.



Gambar 6. Hasil pemotongan dari Metode YOLO

### 4.2.2 Image Segmentation

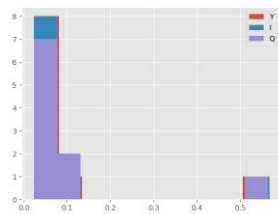
Gambar 7 adalah hasil segmentasi bunga menggunakan segmentasi tepi.



Gambar 7. Hasil segmentasi bunga 1

### 4.2.3 Local Binary Pattern

Gambar 8 adalah histogram yang didapatkan dari *Local Binary Pattern* dengan menggunakan segmentasi tepi.



Gambar 8. Hasil histogram dari *Local Binary Pattern* 1

### 4.2.4 Image Classification Menggunakan Support Vector Machine

Gambar 9 adalah hasil akhir untuk prediksi jenis bunga yang terdapat pada gambar.



Gambar 9. Hasil prediksi jenis bunga

## 4.3 Perhitungan Akurasi untuk Masing-Masing Jenis Bunga

Pengujian dilakukan menggunakan YOLO model pertama *yolo-obj\_last-19.weights* dan deskriptor LBP dengan *numPoints* sebesar 8 dan radius sebesar 4. Parameter tersebut dipilih karena memiliki nilai akurasi tertinggi. Tabel 13 adalah perbandingan akurasi terhadap beberapa *color space* pada masing-masing jenis bunga.

Tabel 13. Perbandingan hasil akurasi terhadap beberapa *color space*

Linear SVM	Grayscale (%)	RGB (%)	HSI (%)	YIQ (%)	oRGB (%)
Amesiella minor	100	100	100	100	100
Amesiella monticola	100	100	100	100	100
Ascocentrum aurantiacum	100	100	100	100	100
Calanthe hennisii	80	80	80	80	80
Calanthe rubens	66.7	66.7	66.7	66.7	66.7
Dendrobium auriculatum	40	40	40	40	40
Dendrobium balzerianum	50	50	50	50	50
Dendrobium dearei	60	60	60	60	60
Dendrobium goldschmidtianum	100	100	100	100	100
Dendrobium heterocarpum	50	50	33.3	33.3	33
Dendrobium macrophyllum	100	100	100	100	100
Dendrobium miyasakii	100	100	100	100	100
Dendrobium schuetzei	100	100	100	100	100
Dendrobium yeageri	100	100	100	100	100
Flickingeria scopa	50	50	50	50	50
Kingidium philippinense	80	80	80	80	80

**Tabel 13. Perbandingan hasil akurasi terhadap beberapa *color space* (lanjutan)**

Linear SVM	Grayscale (%)	RGB (%)	HSI (%)	YIQ (%)	oRGB (%)
Phalaenopsis amabilis	80	100	100	100	80
Phalaenopsis aphrodite	83.3	83.3	83.3	83.3	83.3
Phalaenopsis bastianii	83.3	83.3	83.3	83.3	83.3
Phalaenopsis fasciata	80	100	80	100	100
Phalaenopsis intermedia	83.3	83.3	100	100	100
Phalaenopsis mariae	40	40	40	40	40
Phalaenopsis pulchra	100	100	100	100	100
Phalaenopsis stuartiana	66.7	66.7	66.7	66.7	66.7
Phalaenopsis sumatrana	100	100	100	100	100

## 5. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Berdasarkan perancangan sistem, maka dapat disimpulkan bahwa :

- *Dataset* bunga yang dikumpulkan tidak seluruhnya *capture* bentuk kelopak dari bunga anggrek karena *dataset* yang didapatkan berasal dari *internet searching*.
- Metode *You Only Look Once* (YOLO) digunakan untuk mengetahui posisi bunga dan melakukan klasifikasi untuk bunga anggrek dengan bunga lainnya dengan mAP sebesar 90.69% dan IoU sebesar 71.92%.
- *Image Segmentation* dapat digunakan untuk memisahkan *background* dari objek bunga.
- *Local Binary Pattern* dapat digunakan untuk mendapatkan tekstur dari bunga dalam bentuk *histogram*.
- Klasifikasi untuk menentukan jenis bunga anggrek dilakukan oleh *Support Vector Machine* dengan kernel *linear*.
- Akurasi untuk *color space Grayscale* didapatkan 30.7% sebagai akurasi tertingginya dan didapatkan menggunakan segmentasi tepi yang ditambahkan *circularity* dari gambar.
- Akurasi tertinggi untuk *color space RGB* adalah 37% dan didapatkan menggunakan segmentasi tepi saja.
- Akurasi untuk *color space HSI* sebesar 34.6% yang didapatkan menggunakan segmentasi tepi dan ditambahkan dengan *circularity* dari gambar merupakan akurasi tertinggi.
- Akurasi tertinggi yang dimiliki oleh *color space YIQ* adalah 41% yang didapatkan menggunakan segmentasi tepi saja.
- Akurasi tertinggi dari *color space oRGB* adalah 40.2% dan didapatkan menggunakan segmentasi tepi dan *circularity* dari gambar.
- Jenis *color space* yang paling cocok digunakan dalam melakukan deskripsi menggunakan metode *Color Local*

*Binary Pattern* adalah *color space YIQ* dengan akurasi tertinggi sebesar 41%.

## 5.2 Saran

Berdasarkan hasil pengujian yang dilakukan pada sistem, maka disarankan:

- Improvisasi terhadap segmentasi bunga untuk memisahkan *background*.
- Penambahan jumlah *dataset training* dan menstabilkan jumlah bunga dan keutuhan kelopak bunga yang terdapat pada sebuah gambar.

## 6. DAFTAR PUSTAKA

- [1] C. Dodson, "Orchid," *Encyclopaedia Britannica, Inc.*, 2019. [Online]. Available: <https://www.britannica.com/plant/orchid>. [Accessed: 25-Jun-2019].
- [2] C. Singh, E. Walia, and K. P. Kaur, "Color texture description with novel local binary patterns for effective image retrieval," *Pattern Recognit.*, vol. 76, pp. 50–68, 2018.
- [3] S. Puttemans and T. Goedeme, "Visual detection and species classification of orchid flowers," *Proc. 14th IAPR Int. Conf. Mach. Vis. Appl. MVA 2015*, pp. 505–509, 2015.
- [4] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, vol. 2016-Decem, pp. 779–788, 2016.
- [5] J. Redmon and A. Farhadi, "YOLO v.3: An Incremental Improvement," pp. 1–6, 2018.
- [6] L. Shapiro and G. Stockman, *Computer Vision*, Illustrate. Prentice Hall, 2001.
- [7] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 4th, Illustr ed. Pearson, 2018.
- [8] N. Otsu, "A Threshold Selection Method from Gray-Level Histogram," *IEEE Trans. Syst. Man. Cybern.*, vol. 9, no. 1, pp. 62–66, 1979.
- [9] J. Canny, "A Computational Approach to Edge Detection," *Image Partitioning Perceptuan Organ.*, pp. 184–203, 1987.
- [10] G. Bradski, "The OpenCV Library," *Dr. Dobb's J. Softw. Tools*, 2000.
- [11] R. C. Gonzales and Ri. E. Woods, *Digital Image Processing*, 2nd ed. .
- [12] Joviša Žunić, "Shape Descriptors for Image Analysis," *Zb. Rad.*, pp. 5–38, 2012.
- [13] R. Nock and F. Nielsen, "Statistical region merging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 11, pp. 1452–1458, 2004.
- [14] A. Rosebrock, "Local Binary Patterns with Python & OpenCV," *pyimagesearch*, 2015. [Online]. Available: <https://www.pyimagesearch.com/2015/12/07/local-binary-patterns-with-python-opencv/>.
- [15] F. Pedregosa *et al.*, "Scikit-learn: Machine Learning in {P}ython," *J. Mach. Learn. Res.*, vol. 12, 2011.