

# SISTEM IDENTIFIKASI JENIS BURUNG DENGAN IMAGE CLASSIFICATION MENGGUNAKAN OPENCV

Karno Diantoro<sup>1</sup>, Budhi Adriasyah<sup>2</sup>

[abiluthfi@gmail.com](mailto:abiluthfi@gmail.com)  
[adrians23@gmail.com](mailto:adrians23@gmail.com)

## Abstrak

*Klasifikasi adalah usaha penggolongan makhluk hidup berdasarkan pada persamaan dan perbedaan yang nampak pada makhluk hidup. Klasifikasi atau penggolongan hewan dilakukan dengan cara mencari persamaan ciri di antara keanekaragaman hewan.*

*Proses identifikasi jenis burung yang berbasis komputer dapat dilakukan secara baik dan efektif sehingga dapat mempermudah pengenalan jenis burung. Metode penelitian yang dibuat dengan menggunakan metode deskriptif dan metode aktif dimana gambaran tentang penelitian masih dapat di hubungkan dengan metode image classification, selanjutnya untuk penerapan hubungan metode tersebut akan diambil tindakan untuk pemrograman menggunakan python untuk mengambil gambar atau data objek identifikasi jenis burung dan untuk hasil akhir atau menampilkan hasil menggunakan OpenCv.*

*Harapan kedepannya sistem ini dapat memberikan masukan kepada terhadap rekan-rekan yang ingin melakukan klasifikasi sebagai solusi dari permasalahan. sekarang sudah terkomputerisasi dan proses identifikasi jenis burung menggunakan OpenCv menjadi lebih efektif dan efisien karena identifikasi jenis burung dengan klasifikasi gambar data sudah dalam bentuk sistem.*

**Kata kunci:** *image classification, identifikasi, jenis burung, python, OpenCv*

## Pendahuluan

Untuk mempermudah dalam mengenali dan mempelajari makhluk hidup maka perlu mengenali, mengetahui dan mempelajari makhluk hidup disebut Sistem Klasifikasi (penggolongan/ pengelompokan). Dalam penelitian ini akan membahas secara lebih menghusus pada identifikasi jenis burung.

Penelitian perlu terus dikembangkan untuk memperoleh hasil yang diinginkan. Satu hal yang dapat ditambah atau diimplementasikan dalam pengembangan teknologi pengenalan jenis burung yaitu dengan menambah tingkat kecepatan dan akurasi dalam pendeteksiannya. Banyak dari sistem pendeteksian tersebut menggunakan metode pendeteksi objek. Metode *Image Classification* memiliki keakuratan yang cukup tinggi karena menggabungkan beberapa konsep (Fitur Haar, Citra Integral, *AdaBoost*, *Cascade Classifier*) menjadi sebuah metode utama untuk mendeteksi objek.

Pada penelitian ini akan dijelaskan mengenai cara kerjanya, serta mengaplikasikannya dalam sistem deteksi jenis burung yang sederhana dengan memanfaatkan librari *OpenCV* dan *Python*. Setelah sistem selesai dibuat, akan dibahas mengenai cara kerja sistem deteksi mulai dari akuisisi citra, pengolahan citra, pengenalan pola, dan analisis citra. Kemudian dilakukan pengujian mengenai karakter jenis burung yang dapat dideteksi.

## Pengolahan Citra Awal

Proses ketika citra non-digital diubah ke citra digital. Citra digital diperoleh dari hasil digitisasi citra analog. Digitisasi citra melibatkan dua proses, yaitu *sampling* dan kuantisasi. *Sampling* menunjukkan banyaknya *pixel*/blok untuk mendefinisikan suatu gambar. Kuantisasi

menunjukkan banyaknya derajat nilai pada setiap *pixel* (menunjukkan jumlah bit pada gambar digital, *black/white* dengan 2 bit, *grayscale* dengan 8 bit, *true color* dengan 24 bit). Pengolahan Citra dilakukan untuk memperbaiki kualitas citra agar mudah untuk diinterpretasi oleh manusia/komputer. Masukannya adalah citra dan keluarannya juga citra, tetapi dengan kualitas lebih baik daripada citra masukan. Operasi pengolahan citra yang berhubungan dalam deteksi wajah: *grayscale*, *neighborhood operation*, *thresholding*, *histogram equalization*, *resizing*. Proses analisis citra, dimulai dari pencitraan, sampai proses terakhir, sehingga didapat sebuah keputusan untuk maksud atau tujuan tertentu, misalnya, memandu robot, dan lain-lain. Pada penelitian ini lebih focus bagaimana menentukan jenis burung.



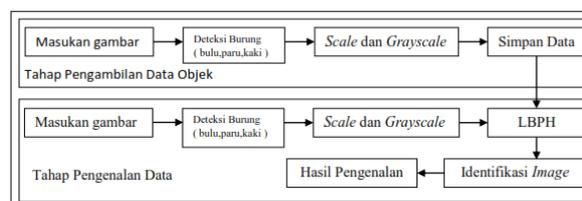
Gambar 1. Diagram Alir Pengolahan Citra

## Image Classification ( Klasifikasi Citra )

*Image Classification* adalah Klasifikasi citra adalah mengelompokkan objek berdasarkan *class* (kelas) tertentu. Sehingga kita dapat dengan mudah mengenali objek. Klasifikasi gambar mengacu pada tugas penggalian kelas informasi dari gambar *raster multiband*. Raster yang dihasilkan dari klasifikasi citra dapat digunakan untuk membuat peta tematik. Bergantung pada interaksi antara analis dan komputer selama klasifikasi, ada dua jenis klasifikasi: *supervised* dan *unsupervised*.

## Algoritma

Analisis algoritma di lakukan untuk dapat mengetahui alur proses dari algoritma yang digunakan dan dapat diterapkan kedalam system perangkat lunak. Dalam proses pengidentifikasian ini menggunakan *library* dalam *OpenCV*. Pada pendeteksian jenis burung pada perangkat lunak ini digunakan metode yaitu *viola-jones*. Serta dibantu oleh algoritma *Local Binari Pattern Histograms* (LBPH) untuk pelabelan gambar (*image recognition*). Berikut ini adalah tahap-tahap yang dilakukan untuk mendeteksi jenis burung,yaitu:



Gambar III.2. Blok Diagram System

## Metode Viola Jones

Prosedur deteksi wajah dengan metode *Viola-Jones* adalah dengan mengklasifikasikan gambar berdasarkan pada nilai fitur sederhana. Terdapat banyak alasan untuk menggunakan fitur daripada piksel secara langsung. Alasan yang paling umum adalah bahwa fitur dapat digunakan untuk mengkodekan pengetahuan domain ad-hoc yang sulit dalam pembelajaran terhadap data latih yang terbatas jumlahnya.

Alasan penting kedua untuk fitur adalah sistem fitur berbasis operasi jauh lebih cepat daripada sistem berbasis *pixel*. Klasifikasi gambar dilakukan berdasarkan nilai dari sebuah fitur.

Penggunaan fitur dilakukan karena pemrosesan fitur berlangsung lebih cepat dibandingkan pemrosesan citra per-piksel. Jika terdeteksi, akan dilakukan penggambaran garis persegi pada objek tersebut. Pendeteksian objek menggolongkan gambar berdasarkan pada nilai dari fitur sederhana. Operasi dasar dari suatu fitur jauh lebih cepat dibandingkan dengan pengolahan *pixel*. Sejumlah Fitur Haar mewakili wilayah persegi pada citra dan menjumlahkan semua piksel pada daerah tersebut.

*Viola Jones* mengklasifikasikan citra dari nilai fitur-fitur sederhana dan menggunakan tiga jenis fitur, yaitu fitur persegi, fitur tiga persegi, dan fitur empat persegi. Nilai dari fitur-fitur tersebut adalah selisih antara daerah hitam dan putih. Di dalam tiap sub-*window image*, jumlah total dari Fitur Haar sangat besar, jauh lebih besar jika dibandingkan dengan jumlah *pixel*. Untuk memastikan pengklasifikasian dapat dilakukan secara cepat, proses pembelajaran harus menghilangkan fitur-fitur mayoritas yang tersedia, dan memusatkan pada sekumpulan kecil fitur yang perlu. *AdaBoost* bertujuan untuk membentuk *template*.

Suatu metode klasifikasi yang menggunakan beberapa tingkatan dalam penyeleksian. Pada tiap tingkatan dilakukan penyeleksian menggunakan algoritma *AdaBoost* yang telah di-*training* dengan menggunakan Fitur Haar. Penyeleksian berguna untuk memisahkan antara sub-*window* yang mengandung positif objek (gambar yang terdeteksi memiliki objek yang diinginkan) dengan negatif objek (gambar yang terdeteksi tidak memiliki objek yang diinginkan).

## Algoritma Haar Cascade

Setelah *training* data telah dilakukan, selanjutnya memanggil algoritma *Haar Cascade* dalam program. Pendeteksian objek (muka, paruh, kaki) pada penelitian ini dikhususkan untuk pendeteksian wajah dengan posisi lurus ke samping terhadap kamera. Dan berikut ini proses-proses yang terjadi dalam metode *haar cascade*.

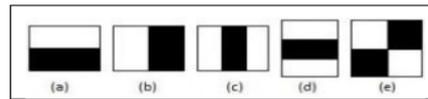
### Proses Menentukan Haar

Nilai *Haar feature* pada *openCV* yang dikenal dengan *Haar cascade*. Proses ini dilakukan untuk mendapat kan sampel dari gambar yang dimasukan dan akan dibandingkan dengan nilai yang ada pada *OpenCV*. Kemudian nilai *frame* akan digunakan untuk mendetek jenis objek gambar yang masuk jenis burung atau bukan. Keberadaan ada atau tidaknya fitur objek (warna muka, paruh, kaki) ditentukan dengan mengurangi nilai *pixel* di wilayah gelap dengan nilai *pixel* di wilayah terang. Jadi setiap gambar dirubah kedalam warna hitam dan putih. Jika nilai dari hasil perbedaannya diatas dari ambang batas selama masa pembelajaran citra maka fitur tersebut dapat dikatakan ada.

Fitur-fitur ini merupakan gambaran dari objek yang dikelompokkan berdasarkan sisi yang terang dan sisi yang gelap. Contohnya daerah mata memiliki sisi yang lebih gelap dari pada bagian yang lain.

Terdapat 3 jenis fitur berdasarkan jumlah persegi yang terdapat di dalamnya, seperti yang

dapat dilihat pada gambar di bawah ini:



Gambar III.3. Pallette Skala *Grayscale*

### Proses Menghitung Citra Integral

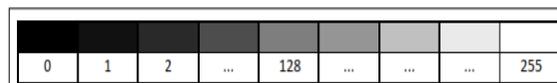
Untuk menghitung nilai dari setiap fitur *Haar* maka digunakan citra integral. Setiap fitur mempunyai nilai yang berbeda sehingga dibutuhkan cara untuk mendapatkan fitur yang memiliki nilai yang paling baik. Cara menghitung nilai dari fitur ini adalah mengurangi nilai piksel pada area hitam dengan piksel pada area putih. Untuk mempermudah proses penghitungan nilai fitur, metode ini menggunakan sebuah media berupa citra integral yaitu adalah sebuah citra yang nilai tiap pikselnya merupakan akumulasi dari nilai piksel atas dan kirinya

### Proses Cascade Classifier

Hasil deteksi dari *Haar* ini belum terlalu bagus karena hasil deteksinya masih sangat sedikit lebih baik dari asal tebak. Jika ingin mendapatkan hasil yang lebih akurat maka harus dilakukan proses *Haar* secara massal (banyak), semakin banyak proses *Haar* yang dilakukan maka akan semakin akurat hasil yang dicapai, namun waktu yang dibutuhkan untuk memproses gambar tersebut akan menjadi lebih lama.

*Grayscale* adalah warna-warna piksel yang berada dalam rentang gradasi warna hitam dan putih. Tahap ini adalah tahap setelah melalui proses *scaling* kemudian diubah menjadi gambar dua warna dengan proses *grayscale*.

Proses *grayscale* dilakukan oleh system perangkat lunak. Pada umumnya warna yang dipakai warna hitam sebagai warna minimal (0) dan warna putih (255) sebagai warna maksimalnya, sehingga warna antaranya adalah abu-abu, seperti ditunjukkan pada gambar berikut.



Gambar III.4. Pallette Skala *Grayscale*

Derajat keabuan sendiri sebenarnya memiliki beberapa nilai, tidak hanya skala 0 sampai 255. Hal ini tergantung pada nilai kedalaman *pixel* yang dimiliki oleh citra. Misalnya dalam gambar objek ( muka, paruh, kaki ) yang telah dimasukan mempunyai nilai warna *Red*, *Green* dan *Blue* seperti pada tabel berikut.

Tabel . Nilai Warna Gambar

x/y		0	1	2	3	4	5	6	7	8	9
0	R	210	215	209	220	220	207	203	192	208	209
	G	200	190	175	185	191	161	155	129	173	163
	B	175	136	104	121	135	84	79	52	115	88
1	R	212	221	228	222	216	214	207	209	193	193
	G	185	192	202	196	189	173	171	173	130	143
	B	132	136	145	145	142	107	113	123	61	72
9	R	212	221	228	222	216	214	207	209	193	193
	G	185	192	202	196	189	173	171	173	130	143
	B	132	136	145	145	142	107	113	123	61	72

## Tahap Scaling

Tahap *scaling* adalah tahap untuk teknik yang berguna untuk merubah ukuran gambar dalam hal ini memperbesar dan memperkecil gambar, biasanya ukuran untuk proses pendeteksian objek yang efektif menggunakan *pixel* 20x20. Misalkan gambar yang telah dimasukan mempunyai nilai *value*:

**Tabel . Nilai Data Gambar**

x/y	0	1	2	3	4	5	6	7	8	9
0	195	180	163	175	182	151	146	124	165	153
1	176	183	192	188	182	165	164	168	128	136
.										
.										
9	176	183	192	188	182	165	164	168	128	136

Dikarenakan gambar berukuran 10x10 maka gambar harus diskala sehingga menjadi 20x20. Karena rumus untuk penskalaan atau scaling adalah :

$$\begin{aligned}x' &= s_x \cdot x \\ y' &= s_y \cdot y\end{aligned}$$

(*s<sub>x</sub>* = ordo Horizontal, *s<sub>y</sub>* = ordo Vertikal)

## OpenCV dan Python

*Open Computer Vision (OpenCV)* sendiri merupakan *library open source* yang tujuannya dikhususkan untuk melakukan pengolahan citra. Maksudnya adalah agar komputer mempunyai kemampuan yang mirip dengan cara pengolahan visual pada manusia. *OpenCV* telah menyediakan banyak algoritma visi komputer dasar. *OpenCV* juga menyediakan modul pendeteksian objek yang menggunakan algoritma *Viola Jones*.

*Python* adalah salah satu bahasa pemrograman tingkat tinggi yang bersifat *interpreter, interactive, object-oriented*, dan dapat beroperasi hampir di semua platform: *Mac, Linux, dan Windows*. *Python* termasuk bahasa pemrograman yang mudah dipelajari karena sintaks yang jelas, dapat dikombinasikan dengan penggunaan modul- modul siap pakai, dan struktur data tingkat tinggi yang efisien (Kadir, 2005).

Distribusi *Python* dilengkapi dengan suatu fasilitas seperti *shell* di *Linux*. Lokasi penginstalan *Python* biasa terletak di “/usr/bin/Python”, dan bisa berbeda. Menjalankan *Python*, cukup dengan mengetikkan “*Python*”, tunggu sebentar lalu muncul tampilan “>>>”, berarti *Python* telah siap menerima perintah. Ada juga tanda “...” yang berarti baris berikutnya dalam suatu blok *prompt* '>>>'. *Text editor* digunakan untuk modus skrip.

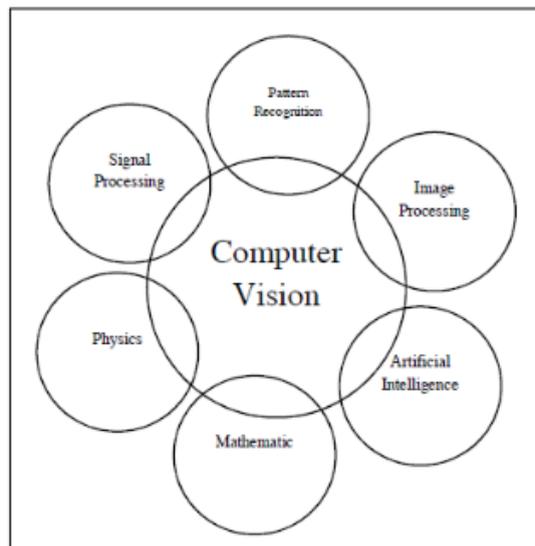
Untuk membangun penelitian ini digunakan *wxPython* yang merupakan *toolkit GUI* untuk bahasa pemrograman *Python*. *wxPython* memungkinkan *programmer Python* untuk membuat aplikasi dengan pondasi kuat, grafis antarmuka dengan pengguna yang sangat fungsional, sederhana, dan mudah. *wxPython* diimplementasikan sebagai modul ekstensi oleh *Python* (kode asli). *wxPython* membungkus *wxWidget* sebagai salah satu *GUI library* populer yang ditulis dalam bahasa *C++*. Selain itu, digunakan pula *Boa Constructor* yang merupakan *Integrated Development Environment (IDE)* untuk *Python* dan *wxPython GUI Builder* yang *cross-platform*. *Boa Constructor* mampu membuat, memanipulasi *frame* secara visual (tanpa skrip), dan ada banyak *object inspector* seperti: *browser* objek, hirarki warisan, *debugger* yang canggih, dan bantuan yang sudah terintegrasi.

## Computer Vision

*Computer vision* merupakan teknologi paling penting di masa yang akan datang dalam pengembangan robot yang interaktif. *Computer vision* merupakan bidang pengetahuan yang berfokus pada bidang sistem kecerdasan buatan dan berhubungan dengan akuisisi dan pemrosesan *images*.

Contoh penerapan *computer vision* pada dunia riset dan industri ialah:

1. Pengontrolan proses industri
2. Pendeteksi nomor plat kendaraan
3. *3D model building (photogrammetry)*
4. *Robot Vision, Humanoid Robot dan Robot Soccer*
5. *Surveillance* (monitor penyusup, analisis trafik jalan tol dan lainnya)
6. *Modelling objek* atau lingkungan
7. Interaksi manusia dan robot (*Human Robot Interaction*)

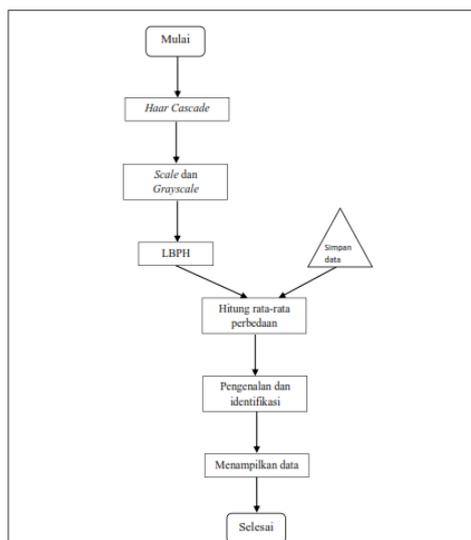


**Gambar Hubungan *Computer Vision* ang Lainnya**

## Metode Penelitian

Suatu prosedur penelitian dibutuhkan agar pekerjaan dapat dilaksanakan secara berurutan dan berkelanjutan tanpa harus mengganggu jenis pekerjaan lainnya. Persiapan meliputi segala sesuatu yang berhubungan dengan proses perancangan, yakni: mempelajari dan memahami cara *Viola Jones* bekerja, perancangan untuk algoritma dan *flowchart*, pembuatan sistem, kemudian menganalisis sistem.

Sistem yang akan dibuat mengacu diagram alir sistem setelah dijalankan untuk mengakuisisi citra, lalu dilakukan pengolahan citra: *grayscale*, *resizing*, *equalization*. Objek dicari menggunakan *Viola Jones*. Jika terdeteksi, akan dilakukan penggambaran garis persegi pada objek tersebut.



Gambar Diagram Alir Sistem Deteksi jenis burung

## Hasil dan Pembahasan

Untuk pendeteksian jenis burung dalam penelitian ini menggunakan gambar data yang memiliki ciri-ciri sebagai berikut:

**Tabel Karakteristik Burung**

No.	Jenis Burung	Ciri-Ciri	No.	Gambar	Jenis Burung	No.	Gambar	Jenis Burung
1		<b>Elang :</b> Warna muka bervariasi Warna paruh Kuning Melengkung Tajam Kaki berwarna putih Tinggi ukuran 40cm – 1m Berat tubuh ± 30 kg Makanan pemangsa daging	1		<b>Elang :</b> Tampak Depan	3		<b>Merak:</b> Tampak Samping Kiri
2		<b>Merak :</b> Warna muka bervariasi berjambul Warna paruh bervariasi Tinggi badan ± 50 cm Berat tubuh ± 6 - 7 kg Makanan pemangsa biji-bijian, aneka serangga, dedaunan	2		<b>Elang :</b> Tampak Samping Kiri	6		<b>Merak:</b> Tampak Samping Kanan
3		<b>Pelikan :</b> Warna muka bervariasi Warna paruh bervariasi Panjang lurus berkantong Tinggi badan ± 30 cm Berat tubuh ± 8 - 10 kg Makanan pemangsa ikan	3		<b>Elang :</b> Tampak Samping Kanan	7		<b>Pelikan:</b> Tampak Depan
			4		<b>Merak:</b> Tampak Depan	8		<b>Pelikan:</b> Tampak Samping Kiri
			5			9		<b>Pelikan:</b> Tampak Samping Kiri

Penyeleksian fitur akan melibatkan nilai fitur, nilai fitur tersebut dihitung dengan *Integral Image*. Jumlah sub-window pada suatu citra terlalu banyak, maka dilakukan penyeleksian sub-window oleh *Cascade Classifier (template)*. Sub-window yang lolos seluruh tahapan seleksi *Classifier* akan dideskripsikan. Menampilkan Data Gambar dengan Objek

```

Elang_dpn1 - C:\Python27\1TEST\Elang_dpn1
File Edit Format Run Options Windows Help
import cv2

image = cv2.imread('pemakandaging.png')
cv2.imshow('Burung Elang pemakan Daging', image)

cv2.waitKey(0)
cv2.destroyAllWindows()

```



Gambar Hasil Identifikasi Objek

```

Gray_PemakanDaging - C:\Python27\1TEST\Gray_PemakanDaging
File Edit Format Run Options Windows Help
import cv2
import numpy as np

img = cv2.imread('pemakandaging.png', cv2.IMREAD_GRAYSCALE)

cv2.imshow('Burung Elang', img)

cv2.waitKey(0)
cv2.destroyAllWindows()

```



Gambar Hasil Proses RGB ke Gray

```

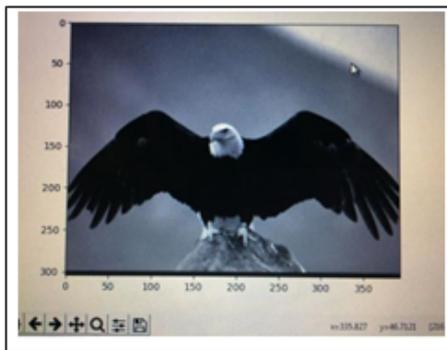
Scale_pemakanDaging - C:\Python27\1TEST\Scale_pemakanDaging
File Edit Format Run Options Windows Help
import cv2
import numpy as np
import matplotlib.pyplot as plt

img = cv2.imread('pemakandaging.png', cv2.IMREAD_GRAYSCALE)
#IMREAD_COLOR = 1
#IMREAD_UNCHANGED = -1

##cv2.imshow('Jenis Burung Pemangsa', img)
##cv2.waitKey(0)
##cv2.destroyAllWindows()

plt.imshow(img, cmap='gray', interpolation='bicubic')
plt.show()

```



Gambar Hasil Proses Grayscale

```

Edge_PemakanDaging - C:\Python27\1TEST\Edge_PemakanDaging
File Edit Format Run Options Windows Help
import cv2

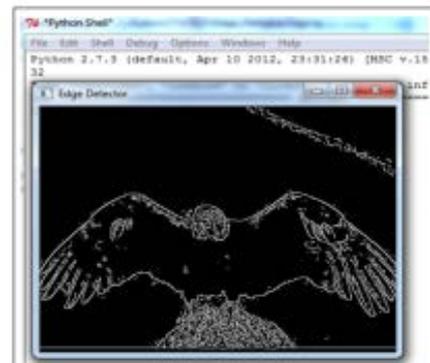
image = cv2.imread('pemakandaging.png')

edge = cv2.Canny(image, 10, 10)

cv2.imshow('Edge Detector', edge)

cv2.waitKey(0)
cv2.destroyAllWindows()

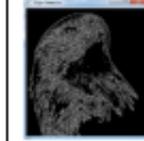
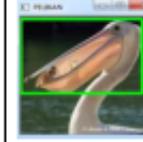
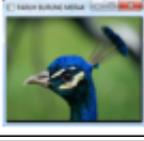
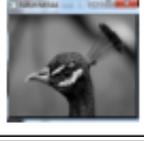
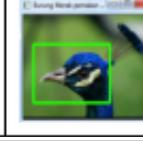
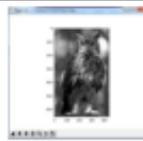
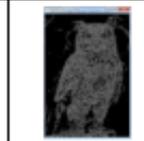
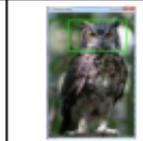
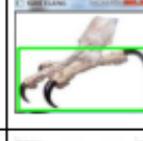
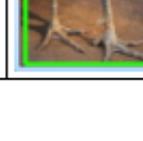
```



Gambar Hasil Identifikasi Edge Detection

## Hasil Identifikasi

**Tabel Tabel hasil dengan OpenCy**

No.	Original	Gray	Grayscale	Edge Detector	Haar Cascade	Keimpulan
1.						Hasil deteksi untuk muka + paruh pemakan daging.
2.						Hasil deteksi untuk muka + Paruh pemakan Ikan Laut.
3.						Hasil deteksi untuk muka + paruh pemakan Bijan.
No.	Original	Gray	Grayscale	Edge Detector	Haar Cascade	Keimpulan
5.						Hasil deteksi untuk kaki pemakan serangga malam.
6.						Hasil deteksi untuk kaki pemakan daging.
7.						Hasil deteksi untuk kaki pemakan ikan.
8.						Hasil deteksi untuk kaki pemakan biji-bijian.

## Kesimpulan

Berdasarkan dari penerapan dan percobaan yang dilakukan, maka dapatlah disimpulkan bahwa:

1. Untuk pendeteksian dalam *Image classification* jenis burung menggunakan *OpenCV* dan *Python* sangat rekomendasi dan lebih efektif di karenakan mudah dan mendekati dengan algorithma *viola-jones* untuk pengimplementasian.
2. proses identifikasi dengan konversi suatu *objek untuk Image classification* sangat di pengaruhi oleh resolusi gambar berupa RGB, format gambar, variasi gambar objek yang lebih detail.
3. Penerapan gambar objek di pengaruhi oleh spesifikasi Laptop/PC terutama prosesor dan memori karena proses konversi gambar dapat memakan atau menggunakan memori yang lebih banyak / besar, sehingga hasil identifikasi menjadi lebih baik.

## DAFTAR PUSTAKA

[1]Budiharto, W., 2016. Machine Learning and Computational Intelligence. Yogyakarta: Andi Offset.

Sumber: Prof. Dr. Jogyanto HM, MBA, Akt. 2011. Analisis dan Desain Sistem

- [2]Farndon, J., 2012. Ensiklopedia Mini Hewan. Jakarta: Erlangga.
- [3]Gusa, R. F., 2013. Pengolahan Citra Digital untuk Menghitung Luas Daerah Bekas Penambangan Timah. Volume 2, p. 28.
- [4]Hartono, Bambang. 2013. Sistem Informasi Manajemen Berbasis Komputer. Jakarta: Rineka Cipta. IKAPI, 2013. Ragam Aplikasi Pengolahan Image dengan Matlab. Jakarta: PT Elex Media Komputindo.
- [5]Kadir, A., 2014. Pengenalan Sistem Informasi Edisi Revisi. Yogyakarta: Andi Offset.
- [6]Kurniawan, L. M., 2014. Metode Face Recognition untuk Identifikasi Personil Berdasar Citra Wajah bagi kebutuhan Presensi Online Universitas Negeri Semarang. Scientific Journal of Informatics, Volume 1, pp. 212-213.
- [7]Kusrini. 2010. Strategi Perancang Dan Pengolahan Basis Data. Yogyakarta : Andi.
- [8]Laudon, Kenneth C., Laudon, Jane P. (2010). Management Information Systems (11<sup>th</sup> Edition). New Jersey : Pearson Prentice Hall.
- [9]Masithoh, R. E., Rahardjo, B., Sutiarto, L. & Hardjoko, A., 2011. Pengembangan Computer Vision System Sederhana untuk Menentukan Kualitas Tomat. Agritech, Volume 31, pp. 117-118.
- [10]Mustakini, Jogiyanto Hartono. 2010. Sistem Informasi Teknologi. Yogyakarta: Andi Offset.
- [11]Nugroho, Adi., 2010. Rekayasa perangkat lunak Berorientasi Objek Dengan Menggunakan Metode USDP. Yogyakarta: Andi.
- [12]Panggabean, S. Mutiara. 2012. Manajemen Sumber Daya Manusia. Bogor: Ghalia Indonesia.
- [13]Pratama, I. P. A. E., 2014. Sistem Informasi dan Implementasinya. Bandung: Informatika Bandung.
- [14]Stair, M. Ralph, George W. Reynolds. (2010). Principles of Information Systems: A Managerial Approach. (9th edition). Australia : Thomson Course Technology
- [15]Sujarweni Wiratna V., "Sistem Akuntansi", 1nd ed, Yogyakarta : Pustaka Baru Press, 2015
- [16]Sutabri, Tata. 2012. Konsep Dasar Informasi. Yogyakarta: Andi Konsep Sistem Informasi. Yogyakarta: Andi Offset.
- [17]Sutarman, 2012. Buku Pengantar Teknologi Informasi, edisi III. Yogyakarta: Andi.
- [18]Widodo, P. P., Handayanto, R. T. & H., 2013. Penerapan Data Mining dengan Matlab. Bandung: Rekayasa Sains.