

Identifikasi Kupu-Kupu Menggunakan Ekstraksi Fitur Deteksi Tepi (*Edge Detection*) dan Klasifikasi *K-Nearest Neighbor* (KNN)

<http://dx.doi.org/10.28932/jutisi.v5i2.1744>

Rico Andrian^{#1}, Saipul Anwar [✉]#2, Meizano Ardhi Muhammad^{#3}, Akmal Junaidi^{#4}

[#]Jurusan Ilmu Komputer Universitas Lampung

Jl.Prof. Sumantri Brojonegoro No.1, Gedong Meneng, Bandar Lampung

¹rico.andrian@fmipa.unila.ac.id

²saifulsealovers0098@gmail.com

³meizano@eng.unila.ac.id

⁴akmal.junaidi@fmipa.unila.ac.id

Abstract — Lampung has the only breeding site of butterflies in Indonesia namely Gita Persada Butterfly Park, which has approximately 211 butterfly species. Butterflies can be classified according to patterns found on the wings of a butterfly. Butterfly species have different patterns based on pigment, the structure of the scales and the fall of sunlight. The weakness of the human eye in distinguishing patterns on butterflies is the foundation in building butterfly identification based on pattern recognition. This study uses 6 species of butterflies: *Papilio memnon*, *Troides helena*, *Papilio nephelus*, *Cethosia penthesilea*, *Papilio peranthus*, and *Pachliopta aristolochiae*. The butterfly dataset used is 600 images form of the upper wing side. The pre-processing stage uses the method of scaling, segmentation, and grayscale. The feature extraction stage uses the canny edge detection method by applying smoothing, edge strength, edge direction, non-maximum suppression, and hysteresis threshold. The classification phase uses the K-Nearest Neighbor (KNN) method with values $k = 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21$ and 23 obtained under the Rule of Thumb. The identification of butterfly requires a classification time of 8 seconds. The highest accuracy is obtained from testing with a value of $k = 5$ by 80%.

Keywords— butterflies, canny edge detection, gita persada, KNN, pattern recognition

I. PENDAHULUAN

Kupu-kupu termasuk ke dalam ordo Lepidoptera. Lepidoptera berarti hewan yang memiliki sayap bersisik. Tubuh dan sayap kupu-kupu berwarna yang ditentukan oleh pigmen, struktur sisik dan jatuhnya sinar matahari. Corak pada kupu-kupu inilah yang membedakan antar spesies kupu-kupu. Lampung memiliki taman penangkaran kupu-kupu bernama Taman Kupu-kupu Gita Persada yang memiliki kurang lebih 211 spesies kupu-kupu yang dikembangkan [1]. Pengenalan kupu-kupu dibutuhkan

oleh Taman Kupu-kupu Gita Persada untuk memisahkan kupu-kupu ke dalam spesiesnya. Metode yang biasa dilakukan oleh peneliti yang ada di Gita Persada dalam mengidentifikasi kupu-kupu hanya menggunakan mata manusia. Kelemahan mata manusia dalam membedakan corak pada kupu-kupu merupakan suatu landasan dalam membangun identifikasi kupu-kupu berbasis pengenalan pola.

Penelitian yang akan dilakukan adalah mengklasifikasikan citra kupu-kupu dengan menggunakan ekstraksi fitur deteksi tepi dengan metode klasifikasi *K-Nearest Neighbor*. Penelitian sebelumnya yang menggunakan metode deteksi tepi dan klasifikasi KNN telah banyak dilakukan, seperti [2] yang digunakan dalam mengidentifikasi telinga manusia yang menghasilkan, pada penelitian [3] mengidentifikasi Penyakit Kanker Serviks berdasarkan *dataset* medis, dan [4] dalam mengidentifikasi gambar otak manusia.

Penelitian sebelumnya mengenai identifikasi kupu-kupu telah dilakukan [5] dalam mengidentifikasi spesies kupu-kupu *family Papilionidae* yang ada di Gunung Ereng menggunakan metode ekstraksi fitur *Structural Similarity Measure* (SSIM) dan *Feature Similarity Index* (FSIM). Hasil klasifikasi menggunakan metode KNN untuk ekstraksi fitur SSIM adalah 99,16% dan untuk ekstraksi fitur FSIM adalah 98,84%.

Batasan masalah pada penelitian ini yaitu Kupu-kupu yang akan diidentifikasi berjumlah 6 spesies yaitu Kupu Raja Limau Jantan (*Papilio memnon*), Kupu Raja Helena (*Troides Helena*), Kupu Jojo (*Papilio nephelus*), Kupu Batik Cap (*Cethosia penthesilea*), Kupu Hijau Biru (*Papilio peranthus*), dan Kupu Pantat Merah (*Pachliopta aristolochiae*), *dataset* kupu-kupu yang digunakan berjumlah 600 citra dengan masing-masing spesies

berjumlah 100 citra kupu-kupu, dan sayap kupu-kupu bagian atas yang akan digunakan dalam identifikasi.

Penelitian ini berfokus untuk mengklasifikasikan citra kupu-kupu ke dalam spesiesnya. Tujuan yang ingin dicapai adalah mengidentifikasi citra kupu-kupu menggunakan ekstraksi fitur deteksi tepi dan klasifikasi KNN agar dapat membantu para developer dalam mengembangkan aplikasi identifikasi kupu-kupu.

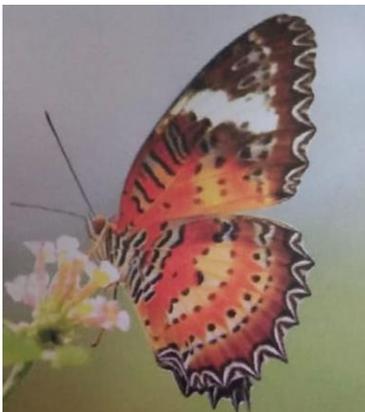
II. TINJAUAN PUSTAKA

A. Kupu-kupu

Kupu-kupu merupakan hewan *pollinator* yang membantu penyerbukan tumbuhan. Kupu-kupu memiliki peran dalam kelangsungan hidup dan perkembangbiakan sebagian besar tumbuhan. Kupu-kupu termasuk kelas insekta, seperti semua serangga, tubuh kupu-kupu terbagi menjadi 3 bagian Kepala, *Thorak*, dan *Abdomen*. Kepala mempunyai sepasang antena, diantara kedua mata majemuknya dan satu *probosis* yang merupakan tabung penghisap panjang. *Thorak* kupu-kupu mempunyai 2 pasang sayap serta 3 pasang kaki. Sayap yang digunakan untuk terbang berupa membran dengan *venasi* yang merupakan dasar klasifikasi yang penting [1].

1) Kupu Batik Cap

Kupu Batik Cap (*Cethosia penthesilea*) merupakan jenis kupu-kupu yang memiliki rentang sayap 7 cm sampai 10 cm, kupu jenis ini memiliki kecepatan terbang dalam kategori sedang, jenis kupu-kupu ini dapat dengan mudah diidentifikasi dikarenakan motif yang dimilikinya menyerupai pola batik cap [1]. Gambar Kupu Batik Cap ditunjukkan pada Gambar 1.



Gambar 1. Kupu Batik Cap

2) Kupu Raja Limau Jantan

Kupu Raja Limau Jantan (*Papilio memnon*) merupakan jenis kupu-kupu yang memiliki rentang sayap antara 15 cm sampai 17 cm, kupu-kupu ini dapat terbang dengan kategori kecepatan sedang, kupu-kupu ini sering terlihat terbang rendah untuk mencari bunga dan

pasangan [1]. Gambar Kupu Raja Limau Jantan ditunjukkan pada gambar 2.



Gambar 2. Kupu Raja Limau Jantan

3) Kupu Hijau Biru

Kupu Hijau Biru (*Papilio peranthus*) merupakan jenis kupu-kupu yang memiliki rentang sayap antara 10 cm sampai 11 cm, jenis kupu-kupu ini dapat terbang dengan cepat, namun kupu-kupu ini sangat mudah dikenali dikarenakan terdapat warna hijau biru yang mendominasi motif yang ada di sayapnya [1]. Gambar Kupu Hijau Biru ditunjukkan pada gambar 3.



Gambar 3. Kupu Hijau Biru

4) Kupu Raja Helena

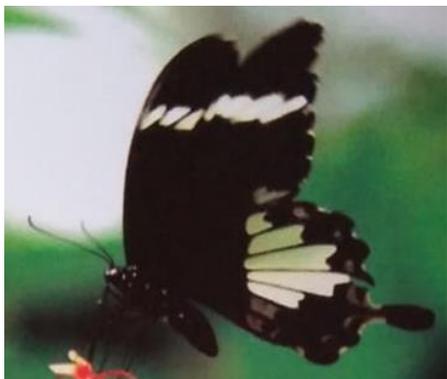
Kupu Raja Helena (*Troides helena*) merupakan salah satu jenis kupu-kupu besar yang memiliki rentang sayap antara 13 cm sampai 17 cm, namun jenis kupu-kupu ini tidak dapat terbang dengan cepat [1]. Gambar Kupu Raja Helena ditunjukkan pada gambar 4.



Gambar 4. Kupu Raja Helena

5) Kupu Jojo

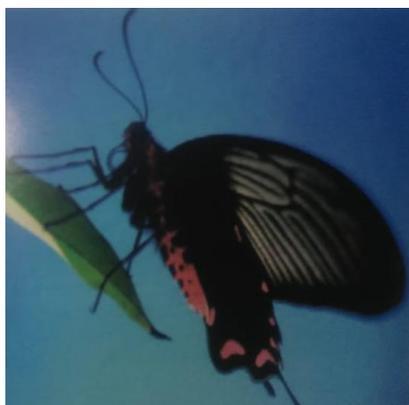
Kupu Jojo (*Papilio nephelus*) merupakan jenis kupu-kupu yang memiliki rentang sayap antara 12 cm sampai 14 cm, kupu-kupu ini dapat terbang dengan sangat cepat namun kupu-kupu ini dapat mudah untuk diidentifikasi dengan pola putih yang terdapat pada sayap [1]. Gambar Kupu Jojo ditunjukkan pada gambar 5.



Gambar 5. Kupu Jojo

6) Kupu Pantat Merah

Kupu Pantat Merah (*Pachliopta aristolochiae*) merupakan jenis kupu-kupu yang memiliki rentang sayap antara 9 cm hingga 10 cm, kupu-kupu ini memiliki kecepatan terbang yang lambat, warna sayap kupu-kupu ini keabu-abuan dan terdapat titik merah yang mengelilingi bagian belakang sayap [1]. Gambar Kupu Pantat Merah ditunjukkan pada gambar 6.

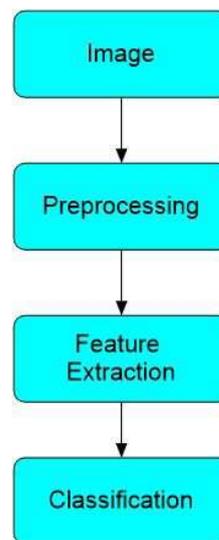


Gambar 6. Kupu Jojo

B. Pengenalan Pola

Pengenalan pola (*pattern recognition*) adalah kegiatan yang berkaitan dengan *machine recognition* terhadap *object* yang memiliki pola beraturan dan kompleks. Tujuan pengenalan pola yaitu untuk mengklasifikasi pola berdasarkan ciri-ciri objek tersebut. Klasifikasi dapat dilakukan dengan menerapkan tahapan yang dapat dilihat pada Gambar 7. Citra diolah tanpa menghilangkan informasi

yang penting. Citra kemudian diekstraksi fitur yang bertujuan untuk mendapatkan informasi dari citra yang berupa fitur [6].



Gambar 7. Proses Pengenalan Pola

1) Preprocessing

Preprocessing bertujuan untuk meningkatkan kualitas suatu citra. Metode yang dilakukan pada *preprocessing* yaitu *scaling*, *segmentation*, dan *grayscale*. *Scaling* adalah merubah ukuran piksel suatu citra menjadi lebih besar atau lebih kecil untuk diambil pola dari suatu citra. *Segmentation* adalah tahap menghilangkan bagian-bagian citra (*background*) yang tidak diperlukan. *Grayscale* yaitu mengubah suatu citra yang memiliki 3 layer warna RGB (*Red Green Blue*) menjadi citra *grayscale* (keabu-abuan) [6].

2) Feature Extraction

Tujuan *feature extraction* adalah untuk mendapatkan informasi (ciri) pada suatu citra, ekstraksi fitur merupakan proses yang penting dalam mengklasifikasikan objek pada pengenalan pola [6]. Citra kupu-kupu yang diklasifikasi dengan baik dapat digunakan untuk pengenalan spesies kupu-kupu.

3) Classification

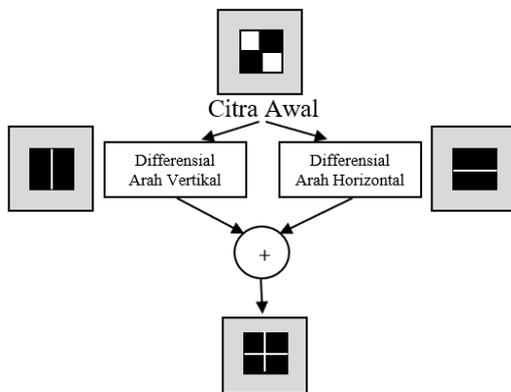
Classification adalah proses penemuan model yang membedakan antara kelas latih yang digunakan untuk memprediksi kelas uji yang belum diketahui kelasnya. Metode klasifikasi yang banyak digunakan yaitu *Rough Sets*, *Naive Bayes classifiers*, *Memory Based Reasoning*, *Neural Network*, *Classification Trees*, *K-Nearest Neighbor*, *Extreme Learning Machine*, dan *Support Vector Machines*. Proses ini dilakukan agar data atau citra dapat dikategorikan dalam suatu kelas tertentu yang telah ditentukan [7].

C. Deteksi Tepi

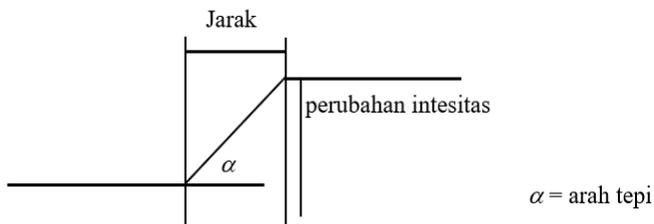
Tepi citra (*edge*) merupakan perubahan intensitas derajat keabuan dalam jarak yang singkat. Deteksi tepi (*edge detection*) adalah metode yang menghasilkan tepi-tepi dari objek-objek pada citra. Deteksi tepi dijalankan dengan mendeteksi garis tepi (*edges*) yang membatasi dua wilayah citra homogen yang memiliki tingkat kecerahan yang berbeda [8]. Deteksi tepi memiliki tujuan, yaitu:

- 1) Untuk menandai bagian yang menjadi detil citra.
- 2) Untuk memperbaiki detil dari citra yang kabur,
- 3) Untuk mengubah citra 2D menjadi bentuk kurva.

Titik (x,y) dikatakan sebagai tepi apabila titik tersebut mempunyai perbedaan yang tinggi dengan tetangganya. Gambar 8 dan 9 menunjukkan bagaimana tepi suatu gambar diperoleh.



Gambar 8. Proses Pengenalan Pola



Gambar 9. Model Tepi

D. Klasifikasi K-Nearest Neighbor

Metode klasifikasi *K-Nearest Neighbor* merupakan metode klasifikasi dihitung berdasarkan data pembelajaran yang jaraknya paling dekat dengan data uji. Algoritma KNN adalah sebagai berikut [10]:

- 1) Tentukan k (jumlah tetangga terdekat yang akan dipilih)
- 2) Hitung jarak antara data yang akan diklasifikasi dengan semua data pelatihan
- 3) Urutkan jarak yang terbentuk secara *ascending*
- 4) Tentukan jarak terdekat sejumlah k
- 5) Pasangkan kelas yang bersesuaian
- 6) Cari jumlah kelas tetangga yang terbanyak dan tetapkan kelas tersebut sebagai kelas data yang akan dievaluasi

E. Perhitungan Tingkat Akurasi

Perhitungan akurasi (*accurate*) sebagai tolak ukur evaluasi dalam sistem. Pengukuran akurasi dapat menggunakan berbagai cara salah satunya menggunakan *recognition rate*. Persamaan (1) *recognition rate* yang digunakan dapat dilihat pada rumus sebagai berikut [7]:

$$\text{Recognition Rate} = \frac{\sum \text{Correct}}{\sum \text{Sample}} \times 100\% \quad (1)$$

Keterangan:

Correct = Jumlah data yang diklasifikasi dengan benar

Sample = Jumlah seluruh data

F. Confusion Matrix

Confusion Matrix adalah digunakan untuk mengukur kinerja suatu metode klasifikasi yang merepresentasikan hasil evaluasi model dengan menggunakan tabel

matriks. Evaluasi menggunakan *confusion matrix* menghasilkan nilai *accuracy*, *precision*, *recall*, dan *error rate* [11]. Tabel *confusion matrix* ditunjukkan pada gambar 10.

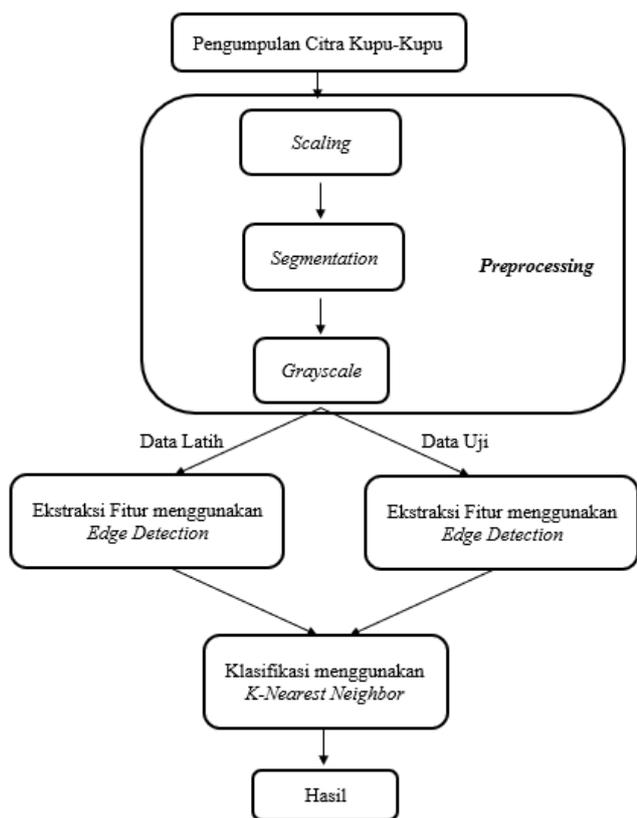
		Nilai sebenarnya	
		TRUE	FALSE
Nilai prediksi	TRUE	TP (True Positive) <i>Correct result</i>	FP (False Positive) <i>Unexpected result</i>
	FALSE	FN (False Negative) <i>Missing result</i>	TN (True Negative) <i>Correct absence of result</i>

Gambar 10. Tabel *confusion matrix*

True Positives (TP) merupakan jumlah data uji yang diklasifikasi dengan benar. *True Negatives* (TN) merupakan jumlah uji kelas lain yang diklasifikasi dengan benar. *False Positives* (FP) merupakan jumlah data uji yang salah klasifikasi. *False Negatives* (FN) merupakan jumlah data uji kelas lain yang salah klasifikasi kedalam kelas tersebut.

III. METODOLOGI

Metodologi penelitian yang dilakukan dalam identifikasi citra kupu-kupu menggunakan ekstraksi fitur deteksi tepi dan klasifikasi KNN ditunjukkan pada Gambar 11.



Gambar 11. Tahapan Penelitian

A. Pengumpulan Citra Kupu-Kupu

Citra yang digunakan pada penelitian ini yaitu Kupu Batik Cap (*Cethosia penthesilea*) (Gambar 12), Kupu Raja Limau Jantan (*Papilio memnon*) (Gambar 13), Kupu Hijau Biru (*Papilio peranthus*) (Gambar 14), Kupu Raja Helena (*Troides helena*) (Gambar 15), Kupu Jojo (*Papilio nephelus*) (Gambar 16), dan Kupu Pantat Merah (*Pachliopta aristolochiae*) (Gambar 17). Citra diambil pada bulan september 2018 di Taman Kupu-kupu Gita Persada sebanyak 100 untuk setiap kelas dengan format JPG (*Joint Photographic Group*) dengan total citra sebanyak 600 citra kupu-kupu. Citra kupu-kupu yang digunakan berupa sisi sayap bagian atas. Kesulitan yang dihadapi dalam pengambilan *dataset* yaitu objek kupu-kupu yang selalu bergerak sehingga memerlukan waktu untuk mendapatkan citra yang baik.



Gambar 12. Contoh Citra Kupu Batik Cap



Gambar 13. Contoh Citra Kupu Raja Limau Jantan



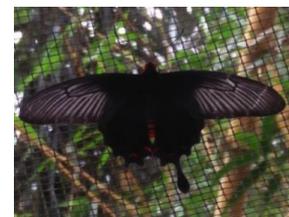
Gambar 14. Contoh Citra Kupu Hijau Biru



Gambar 15. Contoh Citra Kupu Raja Helena



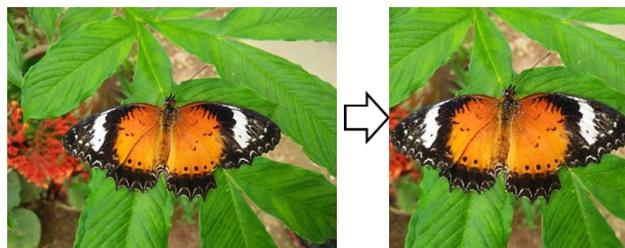
Gambar 16. Contoh Citra Kupu Jojo



Gambar 17. Contoh Citra Kupu Pantat Merah

B. Scaling

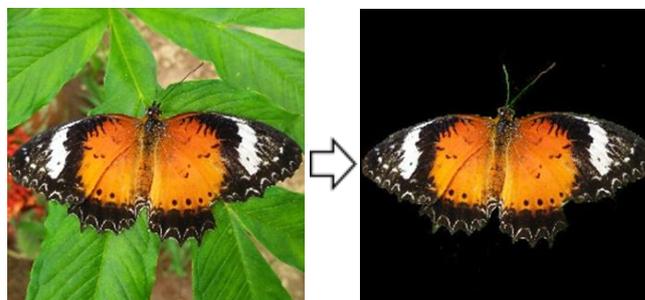
Scaling merupakan proses *resize* ukuran citra bertujuan agar citra yang digunakan memiliki ukuran piksel yang sama [12]. Citra yang berukuran 6000 x 4000 piksel kemudian dipotong (*crop*) sehingga hanya tersisa objek kupu-kupunya saja. Citra kemudian diubah ukurannya menjadi 256 x 256 piksel agar memudahkan dalam proses klasifikasi ditunjukkan pada Gambar 18.



Gambar 18. Transformasi ukuran citra menjadi 256 x 256 piksel

C. Segmentation

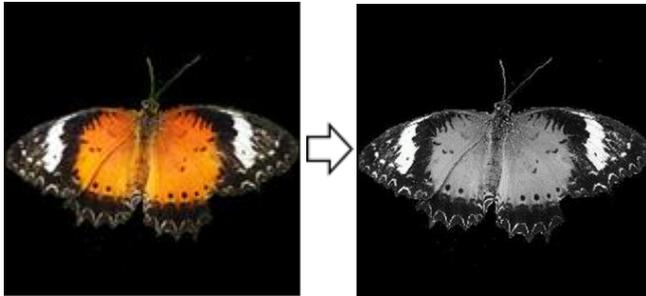
Segmentation merupakan proses pemisahan antara objek (*foreground*) dengan latar belakang (*background*) pada citra yang digunakan [10]. Proses *segmentation* ditunjukkan pada gambar 19.



Gambar 19. Citra sebelum dan sesudah proses *segmentation*

D. Grayscale

Grayscale merupakan proses untuk mengkonversi warna citra menjadi keabu-abuan. Data yang telah dirubah menjadi grayscale nantinya hanya akan memiliki satu nilai untuk setiap pikselnya, dengan menggunakan nilai yang ada ini nantinya gambar-gambar akan diolah [7]. Proses grayscale ditunjukkan pada gambar 20.



Gambar 20. Citra sebelum dan sesudah proses grayscale

E. Ekstraksi Fitur Deteksi Tepi

Penelitian ini menggunakan metode ekstraksi fitur deteksi tepi Canny dilakukan dengan menghitung nilai-nilai pixel hasil deteksi tepi. Deteksi tepi canny terbagi dalam lima tahapan, yaitu:

- 1) Penghalusan citra (*smoothing*) menggunakan *Gaussian Filter*
- 2) Mendapatkan kekuatan tepi (*edge strength*) dengan menggunakan persamaan 2

$$G = \sqrt{|Gx|^2 + |Gy|^2} \quad (2)$$

- 3) Menentukan arah tepi dengan menggunakan persamaan 3

$$\sigma = \tan^{-1} \left(\frac{Gy}{Gx} \right) \quad (3)$$

- 4) Garis tepi diperkecil dengan menerapkan *nonmaksimum suppression*
- 5) Binerisasi dengan menggunakan *hysteresis thresholding*

F. Klasifikasi KNN

KNN bekerja dengan cara mengklasifikasikan suatu objek yang memiliki kemiripan paling dekat dengan objek lainnya. KNN memiliki atribut yang diinisialisasikan sebagai k , yaitu jumlah nilai tetangga yang dijadikan acuan pada klasifikasi KNN untuk dibedakan berdasarkan kelasnya. Penelitian ini menggunakan *Rule of Thumb* pada persamaan 4 untuk menentukan nilai k , sehingga nilai k yang digunakan yaitu 1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, dan 23.

$$k = \sqrt{n} \quad (4)$$

Keterangan :

k = jumlah tetangga terdekat yang akan dipilih

n = jumlah *dataset* yang digunakan

Jarak tetangga dapat dihitung dengan menggunakan algoritma *manhattan distance*. Persamaan 5 adalah persamaan *manhattan distance*:

$$d_{ij} = \sum_{k=1}^n |X_i - X_j| \quad (5)$$

Keterangan:

n = jumlah variabel

X_i = Point Awal

X_j = Target Point

IV. PEMBAHASAN

A. Ekstraksi Fitur

Ekstraksi fitur dilakukan menggunakan metode *canny edge detection* terhadap seluruh citra *dataset*. Gambar 21 menunjukkan hasil *canny edge detection* pada Kupu Batik Cap.



Gambar 21. Hasil *canny edge detection* pada Kupu Batik Cap

Citra yang telah melalui *canny edge detection* kemudian dihitung fitur *area*, *perimeter*, dan *eccentricity* dari garis-garis tepi dengan menggunakan fungsi *bwarea*, *bwperim*, dan *regionprops*. Penelitian ini menggunakan dua algoritma perhitungan *area* yaitu *bwarea* dan *regionprops*. *Bwarea* dan *regionprops* digunakan dalam penghitungan *area* untuk membandingkan nilai fitur *area* pada setiap citra hasil ekstraksi *canny edge detection* (Tabel I).

TABEL I
HASIL PENGUJIAN DENGAN MENGGUNAKAN KLASIFIKASI KNN

<i>Eccentricity</i>	<i>Perimeter</i>	<i>Area1 (regionprops)</i>	<i>Area2 (bwarea)</i>	Nama Kelas
0.96949 4878231 054	8.562379	489	2.891375	Kupu Batik Cap

Hasil ekstraksi fitur dibagi menjadi 2 bagian yaitu data latih yang digunakan pada tahap pelatihan (*training*) dan data uji yang digunakan pada tahap pengenalan (*recognition*). Penelitian ini menggunakan perbandingan data latih dan data uji sebesar 70:30. Data latih yang digunakan berjumlah 420 citra dan data uji berjumlah 180 citra.

B. Pengujian Umum

Hasil ekstraksi fitur *canny edge detection* pada data latih dan data uji kemudian disimpan ke dalam *file* latih.mat yang berukuran 9 KB dan uji.mat yang berukuran 5 KB akan diklasifikasikan dengan menggunakan metode KNN. Hasil klasifikasi kemudian dilakukan pengujian dengan menghitung tingkat akurasi dengan menggunakan persamaan *recognition rate* pada persamaan 1. Hasil pengujian ditampilkan pada Tabel II yang berupa tingkat akurasi dari setiap percobaan.

TABEL III
HASIL PENGUJIAN DENGAN MENGGUNAKAN KLASIFIKASI KNN

Jumlah nilai <i>k</i>	Tingkat akurasi pada setiap kelas						Tingkat akurasi
	BC	HB	J	PM	RH	RL	
1	86,6 %	66,6 %	86,6 %	73,3 %	73,3 %	76,6 %	77,2%
3	96,6 %	70%	86,6 %	70%	83,3 %	70%	79,4%
5	100 %	63,3 %	86,6 %	70%	83,3 %	76,6 %	80%
7	96,6 %	56,6 %	90%	70%	80%	76,6 %	78,3%
9	100 %	56,6 %	90%	70%	83,3 %	70%	78,3%
11	100 %	50%	90%	66,6 %	80%	66,6 %	75,5%
13	100 %	50%	90%	66,6 %	80%	70%	76,1%
15	96,6 %	50%	90%	66,6 %	80%	63,3 %	74,4%
17	96,6 %	53,3 %	90%	63,3 %	80%	66,6 %	75%
19	96,6 %	53,3 %	90%	63,3 %	83,3 %	66,6 %	75,5%
21	96,6 %	56,6 %	90%	70%	83,3 %	66,6 %	77,2%
23	96,6 %	56,6 %	90%	70%	83,3 %	70%	77,7%

Keterangan :

BC = Kelas kupu Batik Cap

HB = Kelas kupu Hijau Biru

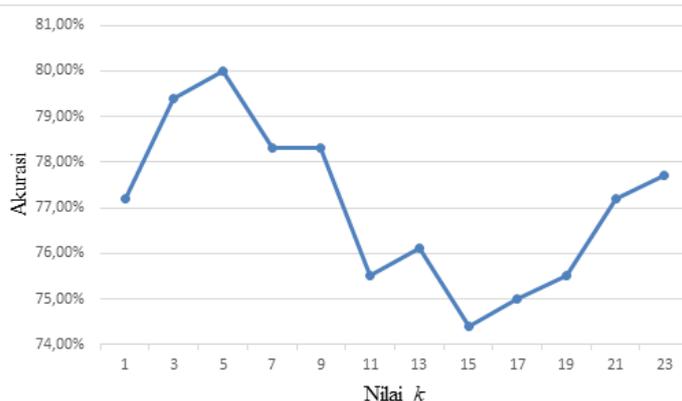
J = Kelas kupu Jojo

PM = Kelas kupu Pantat Merah

RH = Kelas kupu Raja Helena

RL = Kelas kupu Raja Limau Jantan

Tabel II menunjukkan tingkat akurasi pada setiap pengujian dengan nilai *k* yang bervariasi didapatkan bahwa tingkat akurasi tertinggi pada pengujian dengan nilai *k* = 5 sebesar 80%, sedangkan tingkat akurasi terendah didapatkan pada pengujian dengan nilai *k* = 15 sebesar 74,4%. Perbedaan tingkat akurasi pada setiap percobaan disajikan ke dalam bentuk grafik yang ditunjukkan pada Gambar 22.



Gambar 22. Grafik tingkat akurasi Identifikasi kupu-kupu menggunakan fitur ekstraksi edge detection dan klasifikasi KNN

C. Waktu Eksekusi

Waktu yang diperlukan untuk menjalankan Sistem Identifikasi Kupu-Kupu Menggunakan Ekstraksi Fitur Deteksi Tepi (*Edge Detection*) dan Klasifikasi *K-Nearest Neighbor* (KNN), pada tahap ekstraksi data latih sebanyak 420 citra yang dilakukan secara serentak, pada tahap ekstraksi data uji sebanyak 180 yang dilakukan secara serentak, dan pada tahap klasifikasi data uji diidentifikasi yang secara keseluruhan ditunjukkan pada Tabel III.

TABEL IIIII
WAKTU YANG DIPERLUKAN UNTUK SISTEM BEKERJA

	Data Latih	Data Uji
Tahap Ekstraksi	18 detik	8 detik
Tahap Klasifikasi	2 detik	

D. Akurasi dan Temu Balik

Hasil pengujian berdasarkan ekstraksi fitur *canny edge detection* (Tabel II) didapatkan tingkat akurasi tertinggi pada *k* = 5 sebesar 80% yang dijadikan acuan dalam pembuatan *confusion matrix* ditunjukkan pada Tabel IV.

TABEL IVV
CONFUSION MATRIX HASIL KLASIFIKASI KNN DENGAN *k* = 5

	Kupu Batik Cap	Kupu Hijau Biru	Kupu Jojo	Kupu Pantat Merah	Kupu Raja Helena	Kupu Raja Limau Jantan
Kupu Batik Cap	30	0	0	0	0	0
Kupu Hijau Biru	5	19	0	4	0	2
Kupu Jojo	1	0	26	3	0	0
Kupu Pantat Merah	0	3	0	21	6	0

	Kupu Batik Cap	Kupu Hijau Biru	Kupu Jojo	Kupu Pantat Merah	Kupu Raja Helena	Kupu Raja Limau Jantan
Kupu Raja Helena	0	0	3	2	25	0
Kupu Raja Limau Jantan	1	2	0	0	4	23

Data *confusion matrix* pada Tabel IV digunakan untuk menghitung nilai *recall*, *precision*, *accuracy*, dan *error rate* dari setiap kelas kupu-kupu. Hasil perhitungan *recall*, *precision*, *accuracy* dan *error rate* dapat dilihat pada Tabel V.

1) Recall

Recall digunakan untuk membandingkan nilai TP dengan jumlah nilai TP dan FN, dihitung dengan persamaan 6 [11].

$$Recall = \frac{\sum_{i=1}^l TP_i}{\sum_{i=1}^l (TP_i + FN_i)} \times 100\% \quad (6)$$

Keterangan :

TP_i = data positif yang terdeteksi benar pada kelas ke- i
 FN_i = data positif yang terdeteksi sebagai data negatif pada kelas ke- i
 l = jumlah kelas

2) Precision

Precision digunakan untuk membandingkan nilai TP dengan jumlah nilai TP dan FP, yang dihitung dengan menggunakan persamaan 7 [11].

$$Precision = \frac{\sum_{i=1}^l TP_i}{\sum_{i=1}^l (TP_i + FP_i)} \times 100\% \quad (7)$$

Keterangan :

TP_i = data positif yang terdeteksi benar pada kelas ke- i
 FP_i = data negatif yang terdeteksi sebagai data positif pada kelas ke- i
 l = jumlah kelas

3) Accuracy

Accuracy adalah proporsi jumlah prediksi yang benar, dihitung menggunakan persamaan 8 [11].

$$Accuracy = \frac{\sum_{i=1}^l \frac{TP_i + TN_i}{TP_i + TN_i + FP_i + FN_i}}{l} \times 100 \quad (8)$$

Keterangan :

TP_i = data positif yang terdeteksi benar pada kelas ke- i
 TN_i = data negatif yang terdeteksi dengan benar pada kelas ke- i
 FP_i = data negatif yang terdeteksi sebagai data positif pada kelas ke- i
 FN_i = data positif yang terdeteksi sebagai data negatif pada kelas ke- i
 l = jumlah kelas

4) Error Rate

Error Rate adalah persentase *error* yang terjadi dalam klasifikasi. *Error rate* dapat dihitung menggunakan persamaan 9 [11].

$$Error Rate = 100\% - Accuracy \quad (9)$$

Keterangan :

Accuracy = proporsi jumlah prediksi yang benar

TABEL V
RECALL, PRECISION, ACCURACY DAN ERROR RATE HASIL KLASIFIKASI KUPU-KUPU

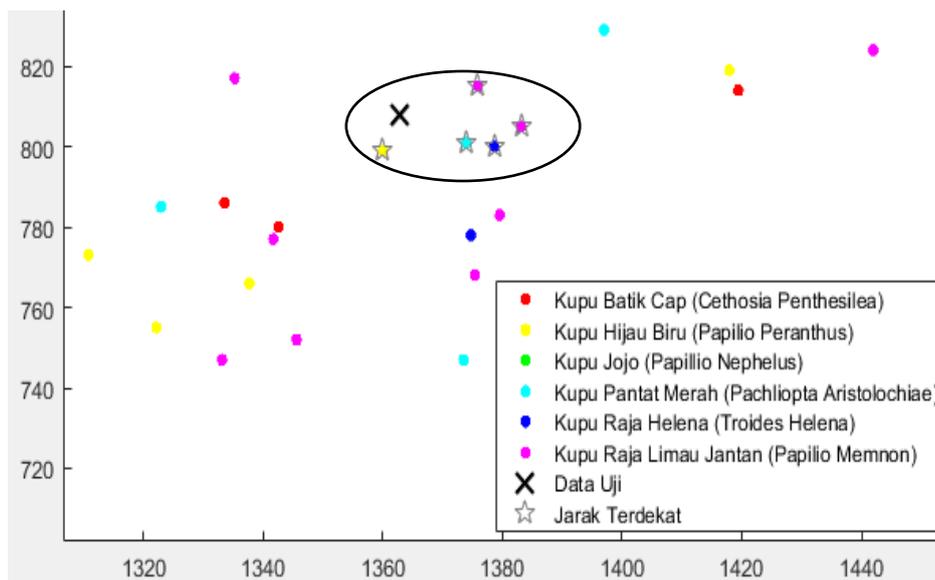
Kelas Kupu-kupu	Hasil	
	Recall	Precision
Kupu Batik Cap	81,1%	100%
Kupu Hijau Biru	79,2%	63,3%
Kupu Jojo	89,7%	86,6%
Kupu Pantat Merah	70%	70%
Kupu Raja Helena	71,4%	83,3%
Kupu Raja Limau Jantan	92%	76,6%
Accuracy	80%	
Error Rate	20%	

Nilai *precision* digunakan untuk mengukur tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem, Tabel V menunjukkan bahwa kelas Kupu Batik Cap memiliki nilai *precision* terbaik yaitu 100%, kelas Kupu Hijau Biru memiliki nilai *precision* terendah yaitu 63,3%, semakin tinggi nilai *precision* maka semakin baik sistem mendeteksi dengan benar.

Nilai *recall* digunakan untuk mengukur tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi. Tabel V menunjukkan bahwa kelas Kupu Raja Limau Jantan memiliki nilai *recall* tertinggi yaitu 92%, sedangkan kelas Kupu Pantat Merah memiliki nilai *recall* terendah yaitu 70%. Nilai *recall* yang tinggi menunjukkan bahwa sistem dapat memisahkan kelas kupu dengan benar dalam jumlah yang besar, sehingga kelas Kupu Raja Limau Jantan lebih mudah dikenali dibandingkan dengan kelas Kupu Pantat Merah.

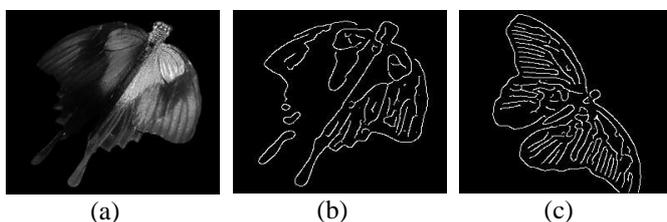
E. Evaluasi Fitur

Kelas data uji yang memiliki tingkat akurasi terkecil terdapat pada kelas Kupu Hijau Biru yang benar diklasifikasi berjumlah 19, sedangkan 5 data uji citra Kupu Hijau Biru diklasifikasi sebagai Kupu Batik Cap, 4 data uji citra Kupu Hijau Biru diklasifikasi sebagai Kupu Pantat Merah, dan 2 data uji citra Kupu Hijau Biru diklasifikasi sebagai Kupu Raja Limau Jantan. Kesalahan identifikasi dapat terjadi karena nilai fitur pada data uji Kupu Hijau Biru mirip dengan kelas Kupu Batik Cap, Kupu Pantat Merah, dan Kupu Raja Limau Jantan yang ditunjukkan pada Gambar 23.



Gambar 23. Sebaran data uji Kupu Hijau Biru

Gambar 23 menunjukkan bahwa data uji Kupu Hijau Biru memiliki tetangga Kupu Raja Limau Jantan berjumlah 2, sehingga data uji Kupu Hijau Biru diklasifikasi ke dalam kelas Kupu Raja Limau Jantan. Kesalahan identifikasi dapat terjadi dikarenakan terdapat fitur yang dihasilkan *canny edge detection* tidak sempurna sehingga dapat mempengaruhi hasil ekstraksi yang ditunjukkan pada gambar 24.



Gambar 24. (a) Citra uji Kupu Hijau Biru (b) Hasil ekstraksi fitur *canny edge detection* (c) Hasil klasifikasi kupu hijau biru

Gambar 24 menunjukkan data uji Kupu Hijau Biru yang salah klasifikasi sebab citra uji (a) memiliki pencahayaan yang kurang yang mengakibatkan hasil ekstraksi fitur *canny edge detection* yang tidak sempurna (b) sehingga citra uji Kupu Hijau Biru diklasifikasikan kedalam kelas kupu Raja Limau Jantan (c).

V. KESIMPULAN

Kesimpulan yang diperoleh berdasarkan penelitian yang telah dilakukan adalah sebagai berikut:

- 1) Implementasi sistem dalam mengidentifikasi kupu-kupu dengan menggunakan ekstraksi fitur *Canny Edge Detection* dan klasifikasi *K-Nearest Neighbor* telah berhasil diimplementasikan dengan tingkat akurasi tertinggi diperoleh pada pengujian pada nilai

$k = 5$ yaitu sebesar 80% dengan tingkat kesalahan klasifikasi sebesar 20%.

- 2) Sistem dapat mendeteksi dengan baik pada kelas Kupu Batik Cap dengan nilai *precision* tertinggi diperoleh pada kelas Kupu Batik Cap yaitu 100% dan sistem dapat mengenali kelas Kupu Raja Limau Jantan dengan cepat dibandingkan dengan kelas lainnya dengan nilai *recall* sebesar 92%.
- 3) Kesalahan identifikasi dapat terjadi dikarenakan fitur yang dihasilkan *canny edge detection* tidak sempurna yang dapat mempengaruhi hasil klasifikasi.
- 4) Waktu yang diperlukan sistem pada tahap ekstraksi sebesar 26 detik yang dibagi menjadi data latih sebanyak 420 citra sebesar 18 detik dan data uji sebanyak 180 citra sebesar 8 detik. Waktu yang diperlukan sistem pada tahap klasifikasi sebanyak 180 citra sebesar 2 detik.

Saran yang diberikan untuk penelitian selanjutnya adalah sebagai berikut:

- 1) Mengembangkan sistem identifikasi kupu-kupu dengan menggunakan metode ekstraksi fitur lain seperti *Rotated Wavelet Filter*, fitur warna, fitur bentuk, dan fitur tekstur.
- 2) Menggunakan metode *scaling* dan *segmentation* yang dapat mempermudah dan mempercepat proses pada tahap preprocessing.
- 3) Mengembangkan sistem identifikasi kupu-kupu dengan menggunakan metode klasifikasi lain seperti *K-Mean Clustering*, *Neural Network*, *Support Vector Machine*, dan *Extreme Machine Learning*.
- 4) Kelas kupu-kupu yang digunakan perlu untuk ditambahkan agar sistem dapat mengidentifikasi jenis kupu-kupu lainnya.

- 5) Jumlah *dataset* yang digunakan perlu ditambahkan agar sistem dapat mengenali dengan berbagai macam data.
- 6) Pengambilan citra *dataset* kupu-kupu dengan menggunakan sudut pandang yang berbeda agar sistem dapat mengidentifikasi bentuk kupu-kupu dari sisi lain.

DAFTAR PUSTAKA

- [1] Soekardi et al, Kupu-Kupu Lampung Taman Kupu-Kupu Gita Persada, Yayasan Sahabat Alam, 2016.
- [2] Jayachandra Chinni & B. Sruthi Patel, *An Ear Recognition Approach using Edge Detection*, IASTER Vol. 1 Issue 1, 38-45, 2013.
- [3] Sharma et al, Classification of Clinical Dataset of Cervical Cancer using KNN, IJST Vol.9, 2016.
- [4] Charde, P.A., & S.D.Lokhande, *Classification Using K Nearest Neighbor for Brain Image Retrieval*, IJSER Vol. 4 Issue 8, 760-765, 2013.
- [5] Faruk et al, Identification of Butterfly Species by Similarity Indexes Based on Prototypes, IJCV Vol. 1, 11-19, 2015.
- [6] Duda, Richard O., Peter E. Hart, dan David G. Stork, *Pattern Classification 2 edition*. New York: A Wiley-Intersciences, 2000.
- [7] Yodha, Johannes Widagdho & Achmad Wahid Kurniawan, *Pengenalan Motif Batik Menggunakan Deteksi Tepi Canny dan K-Nearest Neighbor*, Techno.COM, Vol. 13, No. 4, 2014.
- [8] Wankhade, Mayuri R. dan Narendra M. Wagdarikar, *Feature Extraction of Edge Detected Images*, International Journal of Computer Science and Mobile Computing, Vol. 6 Issue. 6, 2017.
- [9] Putra, Darma. 2010. *Pengolahan Citra Digital*. Yogyakarta: CV. Andi Offset, ISBN : 978-979-291-443-6.
- [10] Andono et al, *Pengolahan Citra Digital*, Andi, 2017.
- [11] Badu, S. Zemi, Penerapan Algoritma *K-Nearest Neighbor* Untuk Klasifikasi Dana Desa, Jurnal Informatika, Universitas Ichsan Gorontalo, 2016.
- [12] Danahy et al, *Algorithms for the resizing of binary and grayscale images using a logical transform*, ISOE Vol. 6497, 2007.