JURNAL

ANALISIS IDENTIFIKASI POLA IKAN KOI MENGGUNAKAN METODE SOBEL EDGE DETECTION

ANALYSIS KOI FISH IDENTIFICATION USING SOBEL EDGE DETECTION METHOD

Oleh:

Arif Wicaksono

13.1.03.02.0080

Dibimbing oleh :

1. M. Rizal Arief, ST., M.Kom.

2. Risky Aswi Ramadhani, M. Kom.

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS NUSANTARA PGRI KEDIRI

TAHUN 2017

SURAT PERNYATAAN
ARTIKEL SKRIPSI TAHUN 2017

Yang bertanda tangan di bawah ini:

Nama Lengkap	: Arif Wicaksono
NPM	: 13.1.03.02.0080
Telepun/HP	: 085708197740
Alamat Surel (Email)	: arifboyz16@gmail.com
SJudul Artikel	: Analisis identifikasi ikan Koi menggunakan metode *Sobel Edge Detection*
Fakultas – Program Studi	: FT – Teknik Informatika
Nama Perguruan Tinggi	: Universitas Nusantara PGRI Kediri
Alamat Perguruan Tinggi	: Jl. K.H Achmad Dahlan No. 76 Kota Kediri

Dengan ini menyatakan bahwa :

a. artikel yang saya tulis merupakan karya saya pribadi (bersama tim penulis) dan bebas plagiarisme;
b. artikel telah diteliti dan disetujui untuk diterbitkan oleh Dosen Pembimbing I dan II.

Demikian surat pernyataan ini saya buat dengan sesungguhnya. Apabila di kemudian hari ditemukan ketidaksesuaian data dengan pernyataan ini dan atau ada tuntutan dari pihak lain, saya bersedia bertanggungjawab dan diproses sesuai dengan ketentuan yang berlaku.

Mengetahui		Kediri, 31 Juli 2017
Pembimbing I	Pembimbing II	Penulis,
M. Rizal Arief, S.T., M.Kom NIDN. 0716027504	Risky Aswi Ramadhani, M.Kom NIDN. 0708049001	Arif Wicaksono 13.1.03.02.0080

ANALISIS IDENTIFIKASI POLA IKAN KOI MENGGUNAKAN METODE SOBEL EDGE DETECTION

Arif Wicaksono
13.1.03.02.0080
FT – Teknik Informatika
arifboyz16@gmail.com
M. Rizal Arief, ST., M.Kom[1], Risky Aswi Ramadhani, M. Kom[2]
UNIVERSITAS NUSANTARA PGRI KEDIRI

ABSTRAK

Ikan koi merupakan salah satu primadona di kelas ikan hias. Bentuknya yang mirip ikan mas dengan warna dan jenis sisik yang beragam menambah daya tarik ikan ini. Tidak hanya warna dan pola sisik yang menarik, ikan koi ini juga terkenal dengan ukuran tubuhnya yang mampu mencapai ukuran jumbo. Memelihara ikan Koi adalah hobi yang menyenangkan dan diyakini dapat mengurangi tingkat stress. Dengan banyaknya jenis ikan Koi yang ada di pasaran saat ini akan menyulitkan orang yang belum mengetahui jenis ikan Koi untuk mengetahui jenis ikan Koi dan mengetahui harga ikan Koi di pasaran. Mendeteksi Jenis ikan Koi dengan menggunakan Gambar/Citra dapat di jadikan solusi untuk mengetahui jenis ikan Koi. Gambar/Citra ikan Koi akan mengalami proses *Greyscale* untuk mendapatkan nilai keabuan kemudian akan mengalami proses deteksi tepi dengan *Sobel Edge Detection* untuk mengetahui pola Ikan Koi. Setelah mendapatkan pola warna Ikan Koi aplikasi akan mencocokkan jenis ikan Koi berdasarkan nilai RGB dengan data jenis Ikan Koi yang ada di Database menggunakan *K-Nearest Neighbor*. Kelebihan dari cara pengenalan ikan Koi menggunakan Gambar/Citra ini adalah dapat mengenali jenis ikan koi secara cepat dan mengurangi penipuan yang dilakukan para penjual ikan Koi yang tidak bertanggung jawab.

Kata Kunci : Pengolahan Citra, *Sobel Edge Detection*, *K-Nearest Neighbor*,Pola Warna Ikan Koi.

I. LATAR BELAKANG

Ikan koi merupakan salah satu primadona di kelas ikan hias. Bentuknya yang mirip ikan mas dengan warna dan jenis sisik yang beragam menambah daya tarik ikan ini. Tidak hanya warna dan pola sisik yang menarik, ikan koi ini juga terkenal dengan ukuran tubuhnya yang mampu mencapai ukuran jumbo. (Yetri, 2015) Nishikigoi adalah nama jepangnya untuk Koi, yang mulai dikembangkan di jepang 200 tahun yang lalu. Awalnya petani beras di Jepang melihat bahwa beberapa ikan di sawahnya memiliki warna yang cemerlang, lalu diambil dan dipeliharanya. Pada tahun 1914, ketika Niigata Koi memamerkan dalam suatu expo, mulailah terjadi demam memelihara koi di seluruh Jepang. Hobi memelihara Koi tersebar keseluruh penjuru dunia setelah kantong plastik dan sarana transport sudah memadai untuk pengiriman ikan dengan selamat. Memelihara Koi adalah hobi yang menyenangkan dan diyakini dapat mengurangi tingkat stress.

Koi adalah ikan yang pintar dan bisa diajarkan untuk makan dari tangan anda. Namun kadang seperti ikan rakus yang akan memakan apa saja yang anda

lemparkan ke kolam. Koi juga bisa mendengar dan akan merespon suara-suara, terutama suara dari pemiliknya. [1] Di Jepang, ikan ini dinamakan Nishikigoi (*Cyprinus carpadie*) Artinya, ikan berwarna warni. Goi sendiri artinya ikan karper. Koi sendiri berasal dari bahasa Cina.Kata koi berasal dari bahasa Jepang yang berarti ikan karper. Lebih spesifik lagi ia merujuk pada nishikigoi yang kurang lebih bermakna ikan karper yang bersulam emas atau perak. Di Jepang, koi menjadi semacam simbol cinta dan persahabatan. Ini karena koi merupakan homofon untuk kata lain yang juga bermakna kasih sayang atau cinta. Koi dikembangbiakkan dari ikan karper biasa. Pada tahun 1820-an, di Jepang mulai muncul usaha mengembangbiakkan ikan karper untuk warnanya. Ini bermula di kota Ojiya di perfektur Niigata yang berada di wilayah timur laut Pulau Honshu.

Dengan banyaknya jenis ikan Koi akan menyulitkan orang yang belum mengetahui jenis ikan Koi untuk mengetahui jenis ikan Koi dan mengetahui harga di pasaran. Biasanya orang yang tidak mengetahui dari jenis ikan koi bertanya kepada penjual atau sekedar mencari tahu di *Internet*. Dengan ketidaktahuan orang dengan jenis ikan koi banyak penjual memanfatkannya untuk menipu para pembeli.

Melihat permasalahan diatas penelitian ini berusaha memberikan solusi untuk orang dapat mengetahui atau mengenali jenis ikan Koi. Penelitian ini dilakukan di Zen Koi Farm yang merupakan salah satu pembudidaya ikan koi di Kabupaten Kediri. Penelitian dilakakuan dengan cara mengambil beberapa sample Citra Gambar Ikan Koi. Citra/Gambar ikan Koi akan melalui proses greyscale untuk mendapatkan nilai keabuan, setelah melalui greyscale kemudian citra ikan akan melalui proses deteksi tepi dengan menggunakan *Sobel Edge Detection* untuk mengetahui pola ikan dan nilai RGB baru. Kemudian, aplikasi akan mencocokkan jenis ikan Koi berdasarkan nilai RGB baru dan data jenis Ikan Koi dari Database. Dengan demikian akan mempermudah orang yang tidak mengetahui jenis ikan Koi untuk mengetahui atau mengenali jenis ikan Koi. Kelebihan dari cara pengenalan ikan Koi menggunakan Citra/Gambar ini dapat mengenali jenis ikan koi secara cepat dan mengurangi penipuan yang dilakukan para penjual ikan Koi yang tidak bertanggung jawab.

II. METODE

Langkah-langkah yang dikembangkan dalam penelitian ini adalah sebagai berikut.

1. Memilih citra yang akan diteliti dari

tempat budidaya ikan Koi.

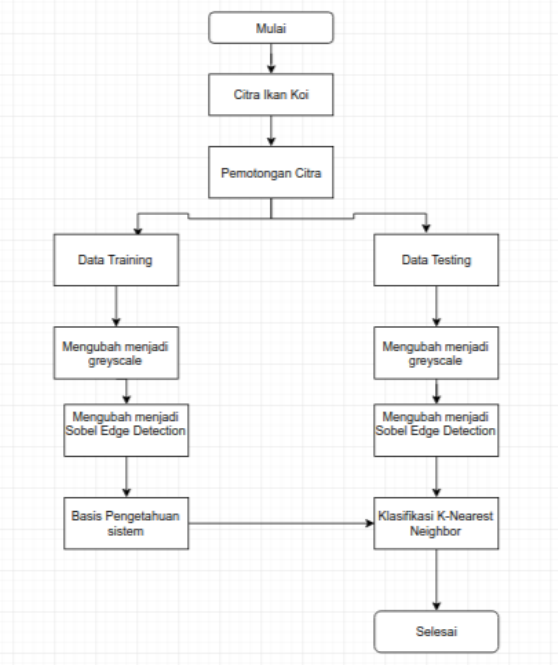
2. Melakukan pemotongan bagian citra yang akan diteliti menjadi ukuran 300 x 300 piksel.

3. Memisahan citra *training* dan *testing* serta kelas-kelasnya.

4. Mengubah citra menjadi *Grayscale*.

5. Mengubah citra menjadi *Sobel Edge Detection*

6. Melakukan klasifikasi citra menggunakan *K-Nearest Neighbor*



Gambar 1 Diagram proses klasifikasi Ikan Koi

A. *Grayscale*

Citra grayscale merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap pixelnya, dengan kata lain nilai bagian RED = GREEN = BLUE Nilai tersebut digunakan untuk menunjukkan tingkat intensitas. Warna yang dimiliki adalah warna dari hitam, keabuan, dan putih. Tingkatan keabuan di sini merupakan warna abu dengan berbagai tingkatan dari hitam hingga mendekati putih.Citra grayscale berikut memiliki kedalaman warna 8 bit yaitu 256 kombinasi warna keabuan. Untuk mengubah RGB menjadi *greyscale* dapat digunakan rumus sebagai berikut:

Contoh RGB sebuah Gambar/Citra :

Tabel 1 Table contoh nilai RGB sebuah Gambar/Citra

X \ Y	1	3	4	5	6
1	245	148	180	137	130
2	180	180	137	150	200
3	137	130	135	178	120
4	130	148	200	125	198
5	180	137	125	120	145

Dengan menggunakan rumus *Greyscale* Ketiga, nilai *pixel* f(x,y) = (100,200,200) dapat dihitung f(x,y)=(100+200+200)/3, sehingga nilai piksel f(x,y)= 166,60 digenapkan menjadi 167 demikian juga untuk piksel lainnya.

1. Tabel ke-(1,1)= (254,240,240) = (254+240+240)/3 = 244.6 = 245
2. Tabel ke-(1,2)= (200,200,45) = (200+200+45)/3 = 148.33 = 148
3. Tabel ke-(1,3)= (210,230,101) = (210,230,101)/3 = 180.33 = 180

4. Tabel ke-(1,4)= (190, 200,20) = (190, 200,20)/3 = 136.6 = 137
5. Tabel ke-(1,5)= (100, 190,100) = (100, 190,100)/3 = 130

Untuk piksel selanjutnya perhitungan nilai *grayscale* sama seperti cara di atas, hasilnya berupa matrik nilai *grayscale* seperti dibawah ini:

Tabel 2 Tabel Contoh Hasil Perhitungan Greyscale

X \ Y	1	3	4	5	6
1	245	148	180	137	130
2	180	180	137	150	200
3	137	130	135	178	120
4	130	148	200	125	198
5	180	137	125	120	145

B. Deteksi Tepi (*Sobel Edge Detection*)

Edge Detection merupakan salah satu proses yang fundamental dalam Pengolahan Citra yang bertujuan mengidentifikasikan titik-titik pada citra digital dimana tingkat kecerahan (*brightness*) berubah drastis atau terjadi diskontinuitas (Yetri, 2015). *Edge Detection* bertujuan membentuk sejumlah kurva yang saling terhubung yang mengindikasikan batas-batas objek, tanda-tanda permukaan, serta kurva-kurva yang mengindikasikan diskon-tinuitas pada orientasi permukaan. Penerapan *Edge Detection* secara signifikan dapat mengurangi jumlah data yang diproses dan menyaring informasi yang mungkin dianggap kurang relevan, sambil menjaga sifat struktural penting dari suatu gambar. *Edge Detection* banyak digunakan dalam aplikasi. Metode pendeteksian tepi *Sobel* adalah metode pendeteksian tepi terbaik dibandingkan dengan metode pendeteksian tepi lainnya seperti metode *Prewitt* dan metode *Robert*. Jika pada metode *Robert* atau *Prewitt* hanya menggunakan jendela 2x2 pada maskingnya, maka pada metode Sobel menggunakan jendela 3x3 pada maskingnya sehingga hasil yang di dapat menjadi lebih baik dari metode-metode pendeteksian tepi yang lainnya.

Peninjauan pengaturan pixel di sekitar pixelnya (x,y) adalah :

$$\begin{bmatrix} a_0 & a_1 & a_2 \\ a_7 & (x,y) & a_3 \\ a_6 & a_5 & a_4 \end{bmatrix}$$

Dalam hal ini turunan parsial dihitung dengan :

$S_x = (a_2 + ca_3 + a_4) – (a_0 + ca_2 + a_6)$

$S_y = (a_0 + ca_1 + a_{22}) – (a_6 + ca_5 + a_4)$

Dengan konstanta c = 2. Dalam bentuk s_x dan s_y dapat dinyatakan sebagai berikut :

$$S_X = \begin{vmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{vmatrix}$$

$$S_y = \begin{vmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{vmatrix}$$

Operator sobel adalah magnitudo dari gradien yang dihitung dengan :

$|G| = \sqrt{S_x^2 + S_y^2}$ .....(1)

Contoh Perhitungan :

Tabel 3 Tabel contoh nilai RGB sebuah Gambar/Citra

X \ Y	1	2	3	4	5
1	8	5	2	7	5
2	5	10	8	4	4
3	9	8	4	8	11
4	2	8	3	6	6
5	3	5	9	10	6

1. a. Nilai intensitas warna citra awal *f*(1,1) – *f*(3,3) di kali dengan operator *Gx*(1,1) – *Gx*(3,3):
   Sx = | (-1*8) + (-2*5) + (-1*2) + (1*9) + (2*9) + (1*7) | = 14

   b. Nilai intensitas warna citra awal f(1,1) – f(3,3) di kali dengan operator *Gy*(1,1) – *Gy*(3,3):
   Sy = | (1*2) + (2*20) + (1*7) + (-1*8) + (-2*5) + (-1*9) | = 22
   |G| = $\sqrt{14^2 + 22^2}$ = 26

2. a. Nilai intensitas warna citra awal *f*(2,1) – *f*(4,3) di kali dengan operator *Gx*(1,1) – *Gx*(3,3):
   Sx = | (-1*5) + (-2*2) + (-1*7) + (1*9) + (2*7) + (1*15) | = 22

   b. Nilai intensitas warna citra awal *f*(2,1) – *f*(4,3) di kali dengan operator *Gx*(1,1) – *Gx*(3,3):
   Sy = | (1*7) + (2*7) + (1*15) + (-1*5) + (-2*24) + (-1*9) | = |-26| = 26
   |G| = $\sqrt{22^2 + 26^2}$ = 34

C. *K-Nearest Neighbor*

K-Nearest Neighbor adalah suatu metode yang menggunakan algoritma *supervised* dimana hasil dari *query instance* yang baru diklasifikan berdasarkan mayoritas dari kategori pada *K-Nearest Neighbor*. Tujuan dari algoritma ini adalah mengklasifikasikan obyek baru berdasarkan atribut dan *training sample*. Algoritma metode *K-Nearest Neighbor* sangatlah sederhana, bekerja berdasarkan jarak terpendek dari *query instance* ke *training sample* untuk menentukan *K-Nearest Neighbor* -nya. *Training sample*

diproyeksikan ke ruang berdimensi banyak, dimana masing- masing dimensi merepresentasikan fitur dari data. Ruang ini dibagi menjadi bagian-bagian berdasarkan klasifikasi *training sample*. Sebuah titik pada ruang ini ditandai kelac *c* jika kelas *c* merupakan klasifikasi yang paling banyak ditemui pada *k* buah tetangga terdekat dari titik tersebut. Dekat atau jauhnya tetangga biasanya dihitung berdasarkan *Similarity* yang direpresentasikan sebagai berikut:

$$Similarity\ (T,S) = \frac{\sum_{i=1}^{n} f(T_i,S_i)\ x\ W_i}{W_i} \quad .....(2)$$

dengan :

T : kasus baru

S : kasus yang ada dalam penyimpanan

n : jumlah atribut dalam masing-masing kasus

i : atribut individu antara 1 s/d n

f : fungsi similarity atribut i antara kasus T dan kasus S

w : bobot yang diberikan pada atribut ke i

Contoh Perhitungan :

Misal nilai RGB Citra/Gambar yang ada di penyimpanan :

Tabel 4 Contoh Tabel Nilai RGB Gambar/Citra yang ada di penyimpanan

No	R	G	B	WI(Warna Ikan)	Jenis Ikan Koi
1	18	13	16	0.75	Kohaku Sisik
	0	7	7		u Sisik

Data *test*

Tabel 5 Contoh Tabel Nilai RGB Citra/ Gambar baru

A	B	C	Q
179	133	165	0.50

Dihitung :

Jarak Kohaku Sisik

$$= \left(\frac{(R*A)+(G*B)+(B*C)+(Q*WI)}{A+B+C+W}\right) \quad .....(3)$$

$$= \frac{(180*179)+(137*133)+(167*165)+(0.50*0.75)}{179+133+165+0.50}$$

$$= \frac{77996.375}{477.5}$$

= 163.34

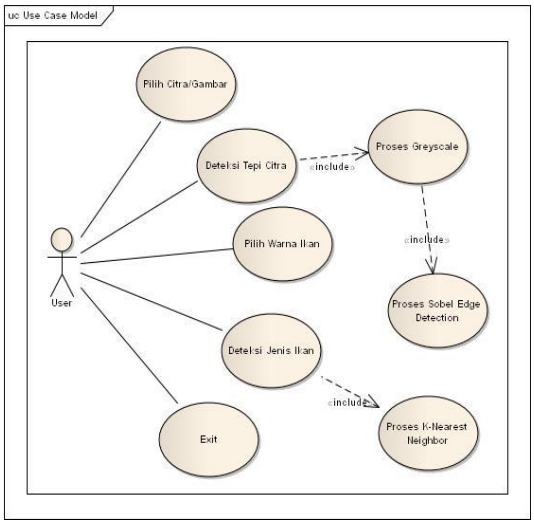
Jadi, sample data testing akan diklasifikasikan Ikan Koi jenis Kohaku Sisik karena pendekatan jarak 163.34

III. HASIL DAN KESIMPULAN

A. *USE CASE PROGRAM*

Berdasarkan kasus diatas, maka dapat dijabarkan dalam suatu pemodelan *Use Case*. Model *Use Case* ini terdiri dari aktor dan kasus penggunaan. Aktor mewakili pengguna sistem yang berinteraksi dengan sistem tersebut. Penggunaan *Use Case* ini mewakili perilaku dari Sistem, Skenario bahwa sistem berjalan melalui tanggapan dari seorang Aktor. *Use Case* pada kasus Analisis Identifikasi Pola Ikan Koi

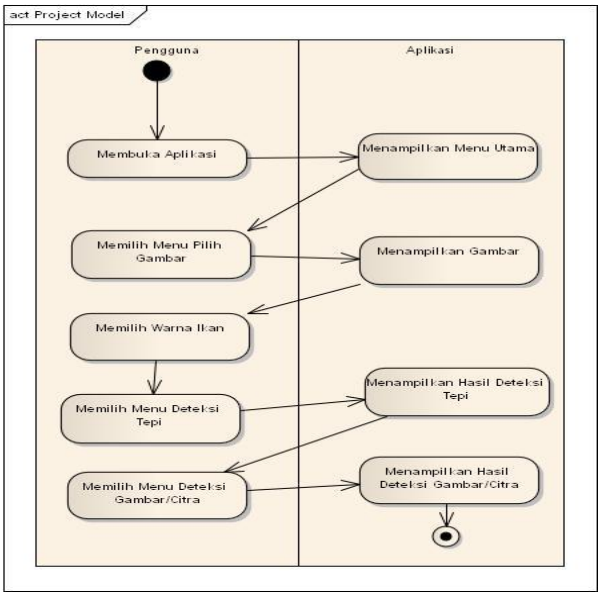
Menggunakan Metode *Sobel Edge Detection* sesuai dengan Gambar



Gambar 2 *Use Case*

B. *ACTIVITY DIAGRAM*

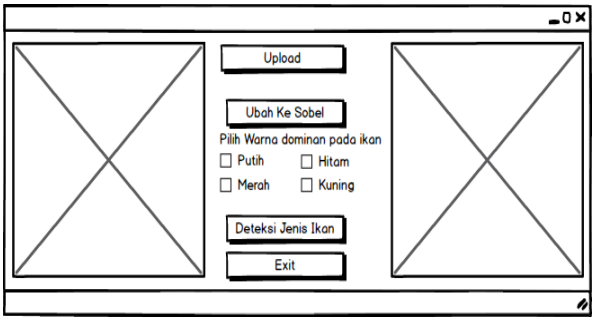
Berdasarkan *Use Case* diatas, maka dapat dijabarkan aktifitas pengguna pada aplikasi. *Activity Diagram* pada *Use Case* diatas sesuai dengan gambar dibawah ini.



Gambar 3 *Activity Diagram*

C. *DESAIN INTERFACE PROGRAM*

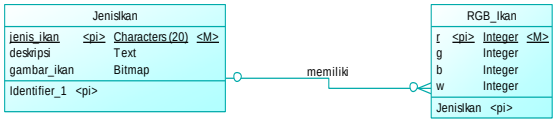
Pada suatu aplikasi *form* menu utama dikenal sebagai sebuah *window* utama yang disebut juga sebagai *window* induk yang di dalamnya terdapat *form* lainnya yang memberikan akses untuk masuk ke dalam *form* lainnya. Berikut ini merupakan gambar dari *form* menu utama yang akan



Gambar 4 Desain *Interface* Program

D. *DESAIN DATABASE PROGRAM*

Pada suatu aplikasi diperlukan Database untuk menyimpan data. Berikut ini merupakan desain Database dari aplikasi yang akan dirancang:



Gambar 5 Desain Database Program

E. KESIMPULAN

Kesimpulan peneliti dalam penelitian ini antara lain sebagai berikut:

1. Dengan menggunakan metode *Sobel Edge Detection* dan metode klasifikasi *K-Nearest Neighbor* kita dapat mengetahui jenis Ikan Koi.
2. Aplikasi pendeteksian jenis ikan Koi juga akan mempermudah orang awam untuk mengetahui jenis ikan Koi dan juga mengurangi penipuan yang

dilakukan para penjual ikan Koi yang tidak bertanggung jawab.

3. Dari percobaan identifikasi ikan Koi yang dilakukan menggunakan aplikasi, hasil yang benar dan sesuai dengan jenis Ikan Koi nya adalah 15 dari 20 gambar ikan koi yang dicocokkan dengan tingkat akurasi sebesar 75%.

IV. DAFTAR PUSTAKA

[1] Yetri, Milfa, Yusnidah, Mukhis Ramadhan, 2015, *Analisa identifikasi pola warna ikan Koi menggunakan Metode Sobel Edge Detection dalam karateristik Citra Sharpening*. (Online), tersedia: https://lppm.trigunadharma.ac.id/public/fileJurnal/hpnrJurnal%20Fix%20Milfa.pdf, diunduh 23 September 2016.

[2] Budianita, Elvia, Jasril, Lestari Handayani, 2015, *Implementasi Pengolahan Citra dan Klasifikasi K-Nearest Neighbour Untuk Membangun Aplikasi Pembeda Daging Sapi dan Babi*. (Online), tersedia: http://ejournal.uinsuska.ac.id/index.php/sitekin/article/download/1005/1009, diunduh 14 Juni 2017.

[3] Farsia, Laina, Taufik Fuadi Abidin, Khairul Munadi, 2013, *Klasifikasi Gambar Berwarna Menggunakan K-Nearest Neighbor dan Support Vector Machine*. (Online), tersedia: http://www.informatika.unsyiah.ac.id/tfa/pdf/papers/SNASTIKOM-2013.pdf, diunduh 22 November 2016.

[4] Lusiana,Veronica, 2013, *Deteksi Tepi pada Citra Digital menggunakan Metode Kirsch dan Robinson*. (Online), tersedia:http://www.unisbank.ac.id/ojs/index.php/fti1/article/download/1706/62, diunduh 14 Juni 2017.

[5] Listyanto , Sebastian Rori, 2015, *Implementasi K-Nearest Neighbor untuk mengenali pola citra dalam mendeteksi penyakit kulit*. (Online), tersedia: http://eprints.dinus.ac.id/15325/1/jurnal_15398.pdf, diunduh 14 Juni 2017.

[6] Wijayanto , Hanang, 2015 , *Klasifikasi Batik menggunakan metode K-Nearest Neighbor berdasarkan Gray Level Co-Occurence Matrices (GLCM)*. (Online), tersedia: http://eprints.dinus.ac.id/16509/1/jurnal_15459.pdf, diunduh 14 Juni 2017.