

KLASIFIKASI CITRA BATIK BESUREK BERDASARKAN EKSTRAKSI FITUR TEKSTUR MENGUNAKAN JARINGAN SYARAF TIRUAN *SELF ORGANIZING MAP (SOM)*

Yuri Brasilka¹, Ernawati², Desi Andreswari³

^{1,2,3}Program Studi Teknik Infomatika, Fakultas Teknik, Universitas Bengkulu.

Jl. WR. Supratman Kandang Limun Bengkulu 38371A INDONESIA

(telp: 0736-341022; fax: 0736-341022)

¹yuribrasilka@gmail.com,

²w_ier_na@yahoo.com,

³dezieandrez@yahoo.co.id

Abstrak: Motif batik umumnya adalah motif-motif gabungan. Diperlukan suatu cara untuk menentukan suatu ciri khas dari jenis batik tertentu untuk membedakannya dari jenis-jenis batik lainnya yang ada di Indonesia. Dengan demikian diperlukan aplikasi pengklasifikasian batik besurek berdasarkan ekstraksi fitur tekstur menggunakan jaringan syaraf tiruan Self Organizing Map (SOM). Aplikasi ini menggunakan metode *Point Minutiae* sebagai pengekstraksi fitur tekstur citra yang mampu mendeteksi persebaran *Crossing Number* pada citra. Aplikasi ini menggabungkan metode *Point Minutiae* dengan algoritma Jaringan *Self Organizing Map* (SOM) yang memiliki kemampuan untuk melakukan pembelajaran tanpa pengarahan (*unsupervised learning*), sehingga hasil klasifikasi citra yang dihasilkan berdasarkan jarak terpendek dari klaster SOM. Citra yang digunakan dalam penelitian ini adalah citra batik besurek Bengkulu, citra non-batik besurek, dan kombinasi dua motif batik besurek. Aplikasi ini dibangun menggunakan bahasa pemrograman *Java 8.0.1*. Hasil pengujian menggunakan *Point Minutiae* dan *Self Organizing Map* (SOM) pada penelitian ini diperoleh nilai rasio keberhasilan 100% untuk temu kembali citra dan 60% untuk citra gabungan, sedangkan nilai *recall* 63.82%, *precision* 27.78%, dan *MRR* 67.6%. untuk citra uji. Motif batik besurek yang paling baik hasil ujinya adalah relung paku dan rembulan, sedangkan yang paling buruk hasil ujinya adalah burung kuau dan rafflesia.

Kata Kunci : Klasifikasi Citra, Fitur Tekstur, *Minutiae*, *Self Organizing Map*, Java

Abstract: Batik motifs generally are combined differentiate one batik from others types in motives. We need a way to specify the Indonesia. The availability of image classifier characteristic of certain types of batik to application based on texture feature extraction

using Self Organizing Map (SOM) is required to determine the classification of batik besureks dominant motifs. The application uses the Point Minutiae method of extracting features image texture that is able to detect the spread Crossing Number on the image. This application combines the Point Minutiae method with Neural Network Self Organizing Map (SOM) algorithm which the result an image classification based on smallest distance of SOM cluster. The images used in this research are the images of batik Besurek Bengkulu, non-batik besurek, and combination motifs of two batik besurek motifs. The application is built using Java programming language 8.0.1. The test results using Point Minutiae and Neural Network Self Organizing Map (SOM) algorithm in this research resulted in a 100% success ratio for the image retrieval system and 60% for the combination images of batik besurek, with recall value of 63.82%, precision value of 27.78%, and MRR of 67.6% for the test images. The best result of the test is achieved by relung paku and rembulan, while the worst result of the test are burung kua and rafflesia.

Keywords: Image Classification, Feature Texture, Minutiae, Self Organizing Map, Java

I. PENDAHULUAN

Sebagai warisan budaya Indonesia, tepat pada tanggal 2 Oktober 2009 batik telah diakui oleh *United Nations Educational, Scientific, and Culture Organization* (UNESCO) sebagai salah satu warisan budaya dunia. Dengan demikian sangatlah penting untuk menjaga dan mengembangkan seni budaya

batik Indonesia. Jenis dan corak batik tradisional tergolong amat banyak, namun corak dan variasinya sesuai dengan filosofi dan budaya masing-masing daerah. Khasanah budaya Bangsa Indonesia yang demikian kaya telah mendorong lahirnya berbagai corak dan jenis batik tradisional dengan ciri kekhususannya sendiri.

Bengkulu merupakan salah satu kawasan yang berada di wilayah Sumatera. Nama Bengkulu sebagai wilayah penghasil batik belum terkenal secara luas. Namun, provinsi dengan ibu kota Bengkulu ini memiliki ciri khas batik yang tidak kalah mutunya dari batik hasil daerah lainnya. Batik Bengkulu tersebut bernama batik besurek.

Batik besurek merupakan salah satu warisan budaya dari kota yang dikenal sebagai Gading Cempaka ini. Besurek merupakan bahasa Bengkulu yang berarti bersurat atau bertulis. Sehingga batik besurek merupakan batik yang bertulis dan bersurat. Sebagaimana batik-batik dari daerah lainnya, batik besurek ini mempunyai beberapa motif dasar yang membedakannya dengan jenis batik dari daerah lain. Motifnya antara lain yaitu motif kaligrafi, motif bunga rafflesia, motif burung kua, motif relung paku, dan motif rembulan.

Saat ini sudah banyak juga motif lainnya yang merupakan kombinasi dari beberapa motif dasar yang ada. Dengan kata lain satu batik besurek bisa mempunyai dua atau lebih motif sekaligus. Dengan demikian tentu akan muncul kesulitan dalam mengklasifikasikan batik besurek yang memiliki motif yang serupa. Dengan melakukan pengolahan citra digital, citra batik besurek dapat diproses untuk diklasifikasikan sesuai dengan motif dasar yang dimilikinya. Mengingat bahwa kemungkinan satu

jenis batik dapat memiliki dua atau lebih motif setelah adanya perkembangan dibidang seni kreasi batik. Selain itu, dengan melakukan pengklasifikasian batik berdasarkan ekstraksi fitur tekstur, maka batik besurek juga akan lebih mudah dibedakan dengan batik-batik lainnya.

Algoritma *Point Minutiae* mempunyai kelebihan dalam melakukan ekstraksi fitur tekstur dengan cara mendeteksi hasil *Crossing Number* (CN) pada citra yang diekstraksi dan mengabaikan hal-hal yang dianggap suatu *noise* pada citra, contohnya seperti guratan-guratan pada citra, sehingga hasil ekstraksi yang dihasilkan oleh algoritma ini merupakan nilai hasil persebaran *Point* yang dihasilkan oleh *Crossing Number* (CN). Selain itu, peneliti memilih algoritma Jaringan Syaraf Tiruan *Self Organizing Map* (SOM) untuk proses klasifikasi citra batik besurek pada penelitian ini. Algoritma ini memiliki kemampuan pembelajaran tanpa pengarahan (*unsupervised learning*), sehingga dalam proses klasifikasi yang dilakukan tidak membutuhkan target. Algoritma ini akan mengklasifikasikan unit-unit masukan ke dalam kelompok tertentu yang berada pada unit keluaran (*cluster units*).

Terkait dengan masalah pengklasifikasian citra, sebelumnya sudah ada beberapa penelitian yang telah dilakukan, salah satunya oleh Fathin (2014) yang berjudul “Rancang Bangun Aplikasi Pencarian Citra Batik Besurek Berbasis Tekstur Dengan Metode *Gray Level Co-Occurrence Matrix* dan *Euclidean Distance*”. Penelitian ini menghasilkan nilai *precision* dengan citra uji adalah batik besurek 1 motif sebesar 77% dan batik besurek lebih dari 1 motif sebesar 82%. Sistem pencarian citra dengan *Gray Level Co-Occurrence Matrix* ini hanya mampu

mengidentifikasi citra batik besurek saja berdasarkan jarak nilai *Euclidean* citra uji dan citra yang ada di dalam *database*, namun aplikasi ini belum dapat mengklasifikasikan citra batik besurek berdasarkan ekstraksi fitur tekstur. Oleh sebab itu, peneliti ingin melakukan pengembangan aplikasi sebelumnya, sehingga dapat diperoleh sebuah aplikasi baru yang bukan hanya dapat melakukan identifikasi citra batik besurek saja, namun dapat melakukan pengklasifikasian batik besurek berdasarkan motif dasar yang dimiliki. Yang dimaksud dengan motif dasar pada batik besurek adalah motif awal pembentuk seperti burung kua, raflesia, kaligrafi, rembulan, dan relung paku. Sedangkan jika batik pengembangan contohnya adalah motif pohon hayat yang bukan dasar batik besurek karena pencampuran dari motif-motif lainnya. Batik pengembangan adalah di luar wilayah dari penelitian ini.

Penelitian lainnya juga dilakukan oleh Yulianto, dkk (2010). Penelitiannya berjudul “Identifikasi Pola Batik Parang Dengan Algoritma *Point Minutiae* Menggunakan Metode *K-Means Clustering*”. Dari data hasil uji coba yang telah dilakukan, para peneliti menyimpulkan bahwa penelitian dengan metode ini menghasilkan persentase dengan mean dan median mendekati sebesar 100%, berdasarkan titik *centroid* dari masing-masing daerah dengan metode *K-Means* untuk klasifikasi pola batik parang dan *Point Minute* untuk ekstraksi fitur gambar [9].

Penelitian lainnya dilakukan oleh Asmadin, dkk (2011) yang berjudul “Pengelompokan Habitat Dasar Perairan Dangkal Berbasis Data Satelit Quickbird Menggunakan Algoritma *Self Organizing Map* (SOM)”. Dalam penelitian ini klasifikasi algoritma *Self Organizing Map* (SOM) dapat

mengklaster citra *quickbird* dari berbagai kombinasi kanal dan memperoleh hasil yang relatif baik [1].

Penelitian selanjutnya yang juga menggunakan SOM kohonen dilakukan oleh Wahyumianto, dkk (2010) yang berjudul "Identifikasi Tumbuhan Berdasarkan *Minutiae* Tulang Daun Menggunakan SOM Kohonen". Dalam penelitian ini didapat hasil rata-rata tingkat keberhasilan aplikasi dalam klasifikasi menggunakan kohonen SOM untuk menentukan kelas ketiga daun tumbuhan yakni 94,386% untuk jambu biji, 89,469% untuk terong hijau, 85,417% untuk cabai lokal dan 99,731% untuk ubi jalar [8].

Berdasarkan permasalahan dan beberapa penelitian terkait di atas, penulis ingin mengklasifikasikan batik besurek berbasis tekstur dengan melakukan penelitian yang berjudul "Klasifikasi Citra Batik Besurek Berdasarkan Ekstraksi Fitur Tekstur Menggunakan Jaringan Syaraf Tiruan *Self Organizing Map (SOM)*".

II. TINJAUAN PUSTAKA

A. Batik Besurek

Kata "batik" berasal dari gabungan dua kata bahasa Jawa: "*amba*", yang bermakna "menulis" dan "*titik*" yang bermakna "titik". Kain Batik Besurek adalah batik tulis tradisional khas Bengkulu yang termasuk batik pesisir dengan motif dominan kaligrafi Arab dihiasi perpaduan flora dan fauna yang sarat akan makna simbolis, melambangkan hubungan manusia dan alam dengan sang pencipta. Motif dasar batik besurek adalah motif Raflesia, motif Kaligrafi, motif Burung Kuau, motif Relung Paku, dan motif Rembulan.

B. Ekstraksi *Point Minutiae*

Proses ekstraksi dimulai dengan proses penandaan *minutiae*. Penandaan *minutiae* dilakukan dengan cara membagi citra menjadi blok-blok citra berukuran 3x3 piksel. Pendeteksian *minutiae* baik itu titik akhir ataupun percabangan dilakukan menggunakan konsep *crossing number* (CN).

Point Minutiae dideteksi dengan memindai tetangga lokal pada masing-masing piksel *ridge* (percabangan) pada citra menggunakan ukuran *window* 3 x 3. Kemudian nilai CN dihitung, yang didefinisikan sebagai separuh penjumlahan dari perbedaan antara pasangan-pasangan piksel yang bersebelahan pada *eight-neighbourhood*, dapat dilihat pada Tabel 1 berikut ini:

Tabel 1. *Properti Crossing Number*

CN	Properti
0	<i>Isolated Point</i>
1	<i>Ridge ending point</i>
2	<i>Continuing ridge point</i>
3	<i>Bifurcation point</i>
4	<i>Crossing point</i>

Nilai CN pada *ridge* piksel P didapat dari persamaan berikut :

$$CN = 0.5 \sum_{t=1}^8 P_t - P_{t+1} \quad (1)$$

keterangan :

CN = *Crossing Number*

P_t = Piksel tetangga (1-8)

$$P_9 = P_1 \quad (2)$$

P₉ = P₁, karena posisi kesembilan sama dengan posisi P, dan dimana P₁ merupakan nilai tetangga dari P.

Untuk suatu piksel P, kedelapan piksel tetangganya diperiksa dengan arah berlawanan jarum jam seperti pada Tabel 2 berikut ini :

Tabel 2. Pemberian Nomor pada 8 Tetangga Piksel P untuk Proses *Point Minutiae Detection*

P4	P3	P2
P5	P	P1

P6	P7	P8
----	----	----

C. Jaringan Syaraf Tiruan Self Organizing Map (SOM)

Kohonen *Self Organizing Map* (SOM) atau Jaringan Kohonen pertama kali diperkenalkan oleh Prof. Teuvo Kohonen pada tahun 1982. SOM merupakan salah satu metoda dalam Jaringan Syaraf Tiruan (*Neural Network*) yang menggunakan pembelajaran tanpa pengarahan (*unsupervised learning*).

Algoritma *Self Organizing Map* (SOM) adalah sebagai berikut [3] :

1. Menetapkan :
 - a. Input data yang dinormalisasi
 - b. Jumlah kelas
2. Inisialisasi :
 - a. Bobot Input (w_{ij}) dengan nilai sembarang atau hitung dengan :

$$w_{ij} = \frac{\text{Min}(x_i) + \text{Max}(x_i)}{2} \quad (3)$$
 dengan :

w_{ij} = bobot antara variabel input ke-j dengan neuron pada kelas ke-i.

$\text{Min}(x_i)$ = nilai minimum pada variable input ke-i.

$\text{Max}(x_i)$ = nilai maksimum dari variable input ke-i.
 - b. Set parameter *learning rate* (α)
 - c. Set maksimum *epoch* (MaxEpoch).
3. Set *Epoch* = 0

4. Kerjakan jika *Epoch* < MaxEpoch

- a. $\text{Epoch} = \text{Epoch} + 1$
- b. Pilih data secara acak, misalnya data terpilih data ke-j.
- c. Cari jarak antara data ke-j dengan tiap bobot input ke-i (D_j) :

$$D(j) = \sum_i (w_{ij} - x_i)^2 \quad (4)$$

- d. Cari bobot yang terkecil (pemenang)
- e. Update bobot yang baru :

$$w_{ij} \text{ (baru)} = w_{ij} \text{ (lama)} + \alpha [x_i - w_{ij} \text{ (lama)}] \quad ()$$

Mengurangi *learning rate* (*decay_rate*), dengan cara :

$$\alpha = \text{decay_rate} * \alpha_{\text{lama}} \quad (5)$$

Dalam penelitian ini, penerapan algoritma *Self Organizing Map* (SOM) dicoba dengan menggunakan nilai *epoch* maksimum yang ditentukan sistem secara random, *Learning rate* (α)=0.6, nilai *learning rate* minimal yang diharapkan ($\alpha_{\text{SOM minimum}}$)= 0.01, dan nilai laju penurunan *learning rate* (*decay_rate*)= 0.60.

Proses klasifikasi akan berhenti apabila sudah mencapai nilai atau mendekati nilai *epoch* maksimum yang diharapkan, sehingga *stop condition* yang digunakan adalah nilai *epoch*.

III. METODOLOGI

A. Jenis Penelitian

Penelitian tentang klasifikasi citra batik besurek ini termasuk dalam penelitian atau riset dasar (penelitian murni), karena penelitian ini bersifat pengembangan dari ilmu pengetahuan yang telah ada sebelumnya, yaitu melanjutkan sebuah penelitian oleh Fathin Ulfah Karimah (2014), yang berjudul

Rancang Bangun Aplikasi Pencarian Citra Batik Besurek Berbasis Tekstur Dengan Metode *Gray Level Co-Occurrence Matrix* dan *Euclidean Distance* [2].

Dalam penelitian ini, peneliti berusaha membuat aplikasi yang mampu mengklasifikasikan citra batik besurek berdasarkan ekstraksi fitur tekstur dengan menggunakan Jaringan Syaraf Tiruan *Self Organizing Map* (SOM).

B. Sumber Data

Sumber data dalam sebuah penelitian dibedakan menjadi dua yaitu sumber data *primary* (primer) dan sumber data *sekunder* (sekunder). Sumber data *primary* adalah suatu objek atau dokumen *original*. Sumber data *primary* (primer) antara lain meliputi dokumen historis dan legal, hasil eksperimen, data statistik. Sedangkan sumber *sekunder* (sekunder) merupakan data yang dikumpulkan dari tangan kedua atau dari sumber lain yang telah tersedia sebelum penelitian dilakukan. Sumber *sekunder* (sekunder) meliputi komentar, interpretasi, data yang diambil tidak secara langsung [6].

Dalam penelitian ini sumber data yang digunakan adalah sumber data *sekunder* (sekunder), yaitu citra batik besurek yang digunakan oleh Fathin Ulfah Karimah pada penelitiannya dan dari buku ajar Batik Besurek. Jumlah sampel citra batik besurek yang digunakan di dalam penelitian ini ada 70 citra.

C. Metode Pengumpulan Data

Dalam mengumpulkan data ada beberapa metode yang dapat digunakan. Adapun metode yang digunakan dalam penelitian ini adalah metode studi pustaka dan survei. Berikut ini merupakan penjelasan mengenai metode yang digunakan dalam penelitian ini yaitu Studi Pustaka (dokumentasi).

Studi kepustakaan dilakukan dengan mengumpulkan data dan informasi yang digunakan sebagai acuan dalam pembuatan aplikasi pengenalan citra batik besurek. Data dan informasi berupa buku-buku ilmiah, laporan penelitian, skripsi, jurnal dan sumber-sumber tertulis lainnya yang berhubungan dengan pemahaman metode yang digunakan, desain *Unified Modelling Language* (UML), pembuatan aplikasi dengan Netbeans 8.0.1, dan *database* dengan MySQL.

IV. ANALISIS SISTEM

A. Analisis Kebutuhan

1) *Analisis Kebutuhan Fungsional*: Merupakan paparan mengenai fitur-fitur yang akan dimasukkan ke dalam sistem yang dibuat. Adapun fitur-fitur aplikasi pada penelitian ini adalah:

- a. Mampu mengenali dan mengklasifikasikan citra batik besurek 1 motif berdasarkan motif dasar dengan menggunakan metode *Minutiae* dan *Self Organizing Maps* (SOM). Keluaran berupa citra dan tabel hasil pengklasifikasian.
- b. Mampu melakukan penambahan citra batik besurek baru ke *database*.
- c. Memberikan informasi tentang aplikasi dan petunjuk penggunaan aplikasi.

2) *Analisis Non-Fungsional*: Terdiri dari:

- a. Kebutuhan Perangkat Keras (*Hardware*)

Perangkat keras yang mendukung dalam penelitian ini adalah 1 unit Laptop Toshiba L460 dengan

spesifikasi monitor VGA atau SVGA (1366 x 768) dan *processor* Intel Core i3, RAM 2 GB, dan Harddisk 500 GB.

- b. Kebutuhan Perangkat Lunak (*Software*)

Perangkat lunak yang mendukung aplikasi dalam penelitian ini adalah berupa Sistem Operasi *Windows Seven* (7) 32 bit, *Java Netbeans* 8.0.1, *MySQL* untuk merancang *database*, *XAMP 2.6*, dan *Microsoft Office Visio 2007* untuk pembuatan diagram alir sistem, *Unified Modelling Language* (UML) dan merancang diagram alur sistem.

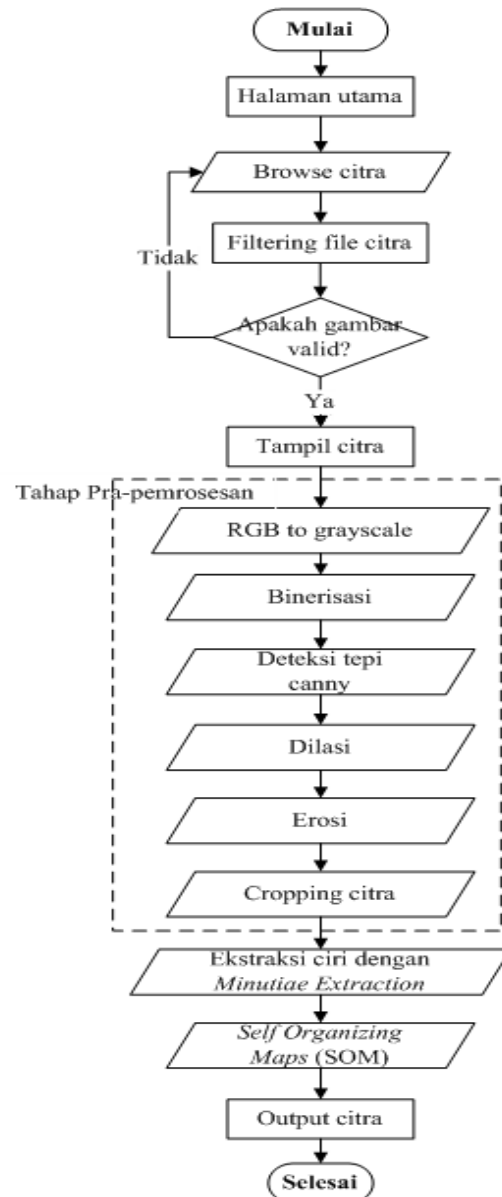
- c. Literatur-Literatur

Literatur-literatur berupa buku (cetak maupun elektronik) dan jurnal ilmiah mengenai bahasa pemrograman *Java Netbeans*, metode *Minutiae*, algoritma Jaringan Syaraf Tiruan *Self Organizing Maps* (SOM), pengolahan citra digital, rekayasa perangkat lunak, dan analisis sistem.

B. Analisis Alur Kerja Sistem

Berikut alur kerja sistem diperlihatkan pada Gambar 4.2. Dari gambar ini dapat dilihat bahwa untuk setiap citra yang akan dikenali akan melalui sebuah tahap yang disebut dengan *preprocessing* atau tahap pra-pemrosesan. Pada tahap ini akan dilakukan beberapa aktifitas, yaitu merubah citra asli (RGB) ke citra abu-abu (*grayscale*), binerisasi untuk merubah citra menjadi nilai biner, deteksi tepi *Canny*, dilasi untuk melakukan penebalan sehingga *noise* yang

tampak dapat dikurangi, erosi yang digunakan untuk penipisan citra, dan *cropping* citra.



Gambar 1. Diagram Alur Kerja Sistem

Pra-pemrosesan merupakan langkah awal sebelum dilakukan ekstraksi ciri terhadap citra batik yang akan dikenali dan diklasifikasi. Ada beberapa proses dalam pra-pemrosesan ini yakni: konversi citra berwarna ke *grayscale*, deteksi tepi *Canny*,

dilasi, erosi, dan *cropping* citra batik besurek. Berikut akan penulis jelaskan lebih mendalam mengenai proses pada implementasi tersebut.

1. Konversi Citra RGB to *Grayscale*

Citra *grayscale* merupakan citra digital yang mengandung matriks data I yang merepresentasikan nilai dalam suatu *range*. Jumlah warna pada citra *grey* adalah $2^8 = 256$, karena citra *grey* jumlah bitnya adalah 8, dengan nilai berada pada jangkauan 0-255. Pengubahan dari citra warna ke bentuk *grayscale* menggunakan persamaan berikut [5] :

$$I_{grayscale}(x, y) = \alpha I_{colour}(x, y, r) + \beta I_{colour}(x, y, g) + \gamma I_{colour}(x, y, b) \dots \dots \dots (6)$$

dengan keterangan:

- $I_{grayscale}$ = citra *grayscale*
- I_{colour} = citra RGB
- (x, y) = koordinat citra
- (x, y, c) = piksel pada kordinat (x, y) , r untuk merah, g untuk hijau, dan b untuk biru
- α_{RGB} = koefisien untuk warna merah (Red)
- β_{RGB} = koefisien untuk warna hijau (Green)
- γ_{RGB} = koefisien untuk warna biru (Blue)

Koefisien α_{RGB} , β_{RGB} , dan γ_{RGB} memiliki rentang nilai 0-1 dengan tujuan untuk merubah citra RGB menjadi *grayscale*. Apabila nilai mendekati 0, maka citra *grayscale* yang dihasilkan semakin gelap, namun apabila nilai mendekati 1 maka citra *grayscale* yang dihasilkan akan semakin terang.

Elemen–elemen dalam matriks intensitas merepresentasikan berbagai nilai intensitas atau derajat keabuan, dimana nilai 0 merepresentasikan warna hitam dan 255 merepresentasikan intensitas penuh atau warna putih. [7].

2. Binerisasi

Binerisasi digunakan untuk membedakan objek gambar dengan latar belakang pada gambar tersebut. Proses ini akan menghasilkan citra hitam putih yang bersih dari tingkat keabuan (*grayscale*), atau dengan kata lain metode ini mengkonversi citra *gray-level* ke citra *bilevel* (*binary image*). Untuk mendapatkan citra *grayscale* digunakan persamaan (6) sebelumnya [5].

Setelah mendapatkan citra *grayscale*, citra biner dibentuk dengan teknik *thresholding*. Jika $g(x, y)$ adalah sebuah nilai ambang (*threshold*) batas dari $f(x, y)$ dengan nilai *threshold* T . Nilai T digunakan untuk memisahkan antara objek dengan *background*-nya, hasil *threshold* dapat ditulis sebagai berikut :

$$g(x, y) = \begin{cases} 1 & f(x, y) \geq T \\ 0 & f(x, y) < T \end{cases} \dots \dots \dots (7)$$

3. Deteksi Tepi *Canny*

Deteksi tepi *Canny* ditemukan oleh Marr dan Hildreth yang meneliti pemodelan persepsi visual manusia. Ada beberapa kriteria pendeteksi tepian paling optimum yang dapat dipenuhi oleh algoritma *Canny* [4] :

- a. Mendeteksi dengan baik (kriteria deteksi)
Kemampuan untuk meletakkan dan menandai semua tepi yang ada sesuai

dengan pemilihan parameter-parameter konvolusi yang dilakukan.

- b. Melokalisasi dengan baik (kriteria lokalisasi)

Dengan metode *Canny* dimungkinkan dihasilkan jarak yang minimum antara tepi yang dideteksi dengan tepi yang asli.

- c. Respon yang jelas (kriteria respon). Hanya ada satu respon untuk tiap tepi, sehingga mudah dideteksi dan tidak menimbulkan kerancuan pada pengolahan citra selanjutnya.

4. Dilasi dan Erosi

Operator erosi merupakan operasi pengecilan sedangkan operator dilasi merupakan operasi ekspansi. Operasi erosi dihasilkan dari perbedaan dan interseksi, sedangkan operasi dilasi dihasilkan dari perbedaan dan gabungan. Transformasi yang melalui dilasi tergantung pada erosi. Notasi dari operator dilasi dan erosi dapat dilihat pada Persamaan (8) dan Persamaan (9).

Proses dilasi dilakukan dengan membandingkan setiap piksel citra input dengan nilai pusat *structuring element* dengan cara melapiskan (*superimpose*) *structuring element* dengan citra sehingga pusat *structuring element* tepat dengan posisi piksel citra yang diproses. Jika paling sedikit ada 1 piksel pada *structuring element* sama dengan nilai piksel objek (*foreground*) citra, maka piksel *input* diset nilainya dengan nilai piksel *foreground* dan bila semua piksel yang berhubungan adalah *background* maka piksel *input* diberi nilai piksel *background*. Proses serupa dilanjutkan dengan

menggerakkan (translasi) *structuring element* piksel demi piksel pada citra *input*.

Semakin besar ukuran *structuring element* maka semakin besar perubahan yang terjadi. Efek dilasi terhadap citra biner adalah memperbesar batas dari objek yang ada sehingga objek terlihat semakin besar dan lubang-lubang yang terdapat di tengah objek akan tampak mengecil. Berikut ini notasi operator dilasi :

$$A \oplus B = \{x \mid (B)_x \cap A \neq \emptyset \dots\dots\dots (8)$$

Suatu objek (citra *input*) dinyatakan dengan A , *structuring element* / SE dinyatakan dengan B , serta $(B)_x$ menyatakan translasi B sedemikian sehingga pusat B terletak pada x , maka operasi dilasi A dengan B dapat dinyatakan sebagai $A \oplus B$ dan $\emptyset$ adalah himpunan kosong.

Sama seperti dilasi, proses erosi dilakukan dengan membandingkan setiap piksel citra input dengan nilai pusat *structuring element* dengan cara melapiskan *structuring element* dengan citra sehingga pusat *structuring element* tepat dengan posisi piksel citra yang diproses. Jika semua piksel pada *structuring element* tepat sama dengan semua nilai piksel objek (*foreground*) citra maka piksel *input* diset nilainya dengan piksel *foreground*. Jika tidak, maka piksel input diberi nilai piksel *background*. Proses serupa dilanjutkan dengan menggerakkan *structuring element* piksel demi piksel pada citra input. Proses erosi akan menghasilkan objek yang menyempit (mengecil). Lubang pada objek juga akan membesar seiring menyempitnya batas objek tersebut. Berikut ini notasi erosi :

$$A \ominus B = \{x \mid (B)_x \subseteq A \dots\dots\dots (9)$$

Operator erosi pada citra digital akan mencari titik-titik yang bernilai minimum di dalam

lingkungan tetangga, sedangkan operator dilasi akan mencari titik-titik yang bernilai maksimum.

5. Cropping Citra

Cropping adalah memotong satu bagian citra sehingga diperoleh citra yang berukuran lebih kecil. Operasi ini pada dasarnya adalah operasi translasi, yaitu menggeser koordinat titik citra. Rumus yang digunakan untuk operasi ini adalah :

$$x' = x - x_L \text{ untuk } x - x_L \text{ sampai } x_R \dots\dots\dots (10)$$

$$y' = y - y_T \text{ untuk } y - y_T \text{ sampai } y_B \dots\dots\dots (11)$$

(x_L, y_T) dan (x_R, y_B) masing-masing adalah koordinat titik pojok kiri atas dan pojok kanan bawah bagian citra yang hendak di-crop. Sehingga, ukuran citra berubah menjadi :

$$h' = y_B - y_T \dots\dots\dots (12)$$

dan transformasi baliknya adalah :

$$x = x' - x_L \text{ untuk } x' = 0 \text{ sampai } w' - 1 \dots\dots (13)$$

$$y = y' - y_T \text{ untuk } y = 0 \text{ sampai } h' - 1 \dots\dots (14)$$

V. HASIL DAN PEMBAHASAN

A. Hasil Implementasi

Implementasi yang dilakukan terdiri dari implementasi antarmuka dan implementasi prosedur. Implementasi antarmuka bertujuan untuk mengetahui apakah sistem yang dibangun sudah berjalan dengan baik dan siap digunakan. Sedangkan implementasi prosedur merupakan implementasi pada pemrograman aplikasi yang menjadi inti dari aplikasi klasifikasi citra batik besurek berdasarkan ekstraksi fitur tekstur menggunakan jaringan syaraf tiruan *Self Organizing Map* (SOM).

B. Hasil Uji Kelayakan Sistem

1) Pengujian Menggunakan Citra "Training" Pada Database

Pengujian ini dilakukan sebanyak 35 kali dengan rincian 7 citra setiap motif dasar batik besurek. Seperti yang telah dijelaskan sebelumnya, bahwa motif dasar batik besurek ada 5, yaitu motif dasar Kaligrafi, Raflesia, Burung Kuau, Relung Paku, dan Rembulan.

Dari hasil pengujian, diperoleh hasil sebagai berikut :

- Nilai *recall* tertinggi diperoleh motif relung paku, yaitu 81.6% dan *recall* terendah diperoleh motif rembulan, yaitu 28.6%
- Nilai *precision* tertinggi diperoleh motif rembulan, yaitu 40.7% dan *precision* terendah diperoleh motif kaligrafi, yaitu 28.92%
- MRR yang dihasilkan adalah 100% untuk semua citra uji menggunakan citra *training*.
- Kinerja keseluruhan dari aplikasi, yaitu *recall* 51.07% dan *precision* 32.57%.

Nilai *recall* motif relung paku tinggi karena motif dasarnya paling jauh berbeda dari motif lainnya. Sedangkan nilai *recall* motif rembulan rendah karena motif yang dimilikinya mempunyai kemiripan dengan motif raflesia. Selain itu, rembulan tidak ada motif pasti atau terdapat banyak variasi, sehingga sering terklasifikasi ke motif raflesia.

2) Pengujian Menggunakan Citra yang Belum Pernah Dilatihkan ke Database

Pada penelitian ini digunakan 15 citra yang belum pernah dilatihkan ke sistem untuk tujuan pengujian. Pengujian ini dilakukan sebanyak 15 kali dengan rincian 3 citra untuk masing-masing motif dasar batik besurek.

Dari hasil pengujian, diperoleh hasil sebagai berikut :

- a. Nilai *recall* tertinggi diperoleh motif relung paku, yaitu 90.5% dan *recall* terendah diperoleh motif rembulan, yaitu 42.86%
- b. Nilai *precision* tertinggi diperoleh motif rembulan, yaitu 51.9% dan *precision* terendah diperoleh motif burung kuau, yaitu 13.2%
- c. MRR tertinggi diperoleh motif rembulan, yaitu 83% dan MRR terendah diperoleh motif relung paku, yaitu 19.5%
- d. Kinerja keseluruhan dari aplikasi, yaitu *recall* 51.07%, *precision* 32.57%, dan MRR 67.6%.

Sama seperti hasil pengujian sebelumnya, pada pengujian ini nilai *recall* dan *precision* yang dihasilkan berbanding terbalik berdasarkan pengujian yang telah dilakukan. Nilai *recall* motif relung paku tinggi karena motif dasarnya paling jauh berbeda dari motif lainnya. Sedangkan nilai *recall* motif rembulan rendah karena motif yang dimilikinya mempunyai kemiripan dengan motif raflesia. Selain itu, rembulan tidak ada motif pasti atau terdapat banyak variasi, sehingga sering terklasifikasi ke motif raflesia. Dari tabel di atas, dapat dilihat bahwa untuk citra yang belum pernah dilatihkan pada

database, terjadi 7 kesalahan klasifikasi dari 15 uji coba yang dilakukan.

Jika dibandingkan dengan rata-rata selisih jarak nilai dalam *cluster*, diperoleh kesimpulan bahwa tidak bisa ditetapkan suatu standar rata-rata selisih jarak dalam *cluster* untuk memisahkan antara satu motif citra dengan motif citra lainnya. Hal ini disebabkan oleh bobot yang digunakan *algoritma Self Organizing Map* (SOM) selalu diinisialisasi secara acak pada setiap percobaan, sehingga berpengaruh pada *distance* atau jarak yang dihasilkan oleh *algoritma Self Organizing Map* (SOM). Dan lebih jauh lagi, hal ini mengakibatkan jumlah *cluster* yang ada pada setiap percobaan dimungkinkan untuk berbeda jumlahnya, demikian pula terdapat variasi dalam *cluster* serta jarak antar *cluster* yang selalu berubah-ubah pada setiap percobaan. Hal ini memberi dampak, sebagai contoh : sebuah motif pada kondisi ideal seharusnya semuanya terletak pada 1 *cluster* yang sama akan tetapi motif tertentu tersebut akibat bobot yang acak mengakibatkan jarak terlalu jauh untuk digolongkan ke dalam 1 *cluster*, melainkan tersebar lebih dari 1 *cluster* dengan demikian standar rata-rata selisih jarak tidak dapat ditetapkan dengan suatu nilai tertentu untuk memisahkan pola motif dasar dominan 1 dengan pola motif dasar dominan lainnya.

3) Pengujian Menggunakan Citra Non-Batik Besurek

Pada pengujian ini dilakukan 10 kali pengujian menggunakan citra non-batik

besurek. Hasil pengklasifikasian menghasilkan selisih jarak yang cukup kecil. Tabel 3 di bawah ini adalah tabel hasil pengujian menggunakan citra non-batik besurek.

Tabel 3. Hasil Pengujian Menggunakan Citra Non-Batik Besurek

No	Nama Citra	Motif Dasar Dominan	Rata-Rata Selisih Jarak
1	BATIK_abimanyu.jpg	Rafflesia	2,371.431
2	BATIK_ceplok.jpg	Burung Kuau	2,620.728
3	BATIK_kawung.jpg	Burung Kuau	494.614
4	BATIK_madura1.jpg	Rembulan	3,245.659
5	BATIK_madura2.jpg	Burung Kuau	2,279.724
6	BATIK_mendung.jpg	Burung Kuau	415.743
7	BATIK_papua2.jpg	Rafflesia	3,398.595
8	BATIK_solo1.jpg	Burung Kuau	936.199
9	BATIK_solo2.jpg	Burung Kuau	638.232
10	BATIK_udanliris.jpg	Burung Kuau	1,352.013

Hasil percobaan pada Tabel 3 akan menuju ke *cluster* dimana rata-rata selisih jaraknya terpendek. Dalam hal ini, fakta yang ditemui adalah batik-batik non-batik besurek dari 10 kali pengujian tidak pernah ter-*cluster* menjadi sebuah *cluster* dominan pada jenis batik non-batik tertentu. Artinya motif-motif citra non-batik besurek yang dimasukkan ke sistem kurang kuat untuk membentuk suatu *cluster* tersendiri terpisah dari beragam motif batik besurek.

Rasionalisasi dibalik ketidakberhasilan algoritma *Self Organizing Map* (SOM) dalam mengenali motif non batik besurek adalah disebabkan oleh karena dia ketika ditarik oleh

Crossing Number Point Minutiae lalu diproses oleh *Self Organizing Map* (SOM), jarak yang dihasilkan selalu mendekati suatu pola motif tertentu dari batik besurek.

4) Pengujian Menggunakan Citra Batik Besurek Gabungan

Pada pengujian ini diharapkan agar aplikasi dapat mengklasifikasikan citra uji batik besurek gabungan berdasarkan motif dasar yang dominan dari kedua motif yang digabungkan menjadi satu. Hasil pengujian ini menunjukkan bahwa 6 dari 10 pengujian berhasil dilakukan oleh aplikasi, sehingga diperoleh rasio keberhasilan sebesar 60%.

Berikut ini adalah Tabel 4 yang merupakan hasil pengujian menggunakan citra batik besurek gabungan :

Tabel 4. Hasil Pengujian Menggunakan Citra Batik Besurek Gabungan

No	Nama Citra	Motif Dasar Dominan	Rata-Rata Selisih Jarak	Kesimpulan
1.	Kuau_Kalig.jpg	Burung Kuau	1,373.811	Klasifikasi Benar
2.	Kuau-Kuau.jpg	Burung Kuau	489.933	Klasifikasi Benar
3.	Kuau_Rembulan.jpg	Burung Kuau	821.395	Klasifikasi Benar
4.	Raf_Kalig1.jpg	Burung Kuau	2,726.33	Klasifikasi Salah
5.	Raf_Kalig2.jpg	Burung Kuau	1,684.119	Klasifikasi Salah
6.	Raf_Kalig3.jpg	Burung Kuau	2,585.424	Klasifikasi Salah
7.	Raf_Kalig4.jpg	Kaligrafi	2,559.784	Klasifikasi Benar
8.	Raf_Kalig5.jpg	Rafflesia	2,576.555	Klasifikasi Benar
9.	Raf_Kalig6.jpg	Burung Kuau	2,370.899	Klasifikasi Salah
10.	Raf_Kalig7.jpg	Kaligrafi	2,591.085	Klasifikasi Benar
Total Benar				6 Benar
Total Salah				4 Salah

Pada Tabel 4, citra yang diujikan merupakan gabungan dari 2 buah citra, sebagai contoh pada

percobaan pertama Tabel 4 citra uji gabungan dari motif burung kua dan motif kaligrafi dengan motif dasar dominan yang ditemukan pada cluster dengan selisih jarak terpendek menunjukkan motif dasar dominan burung kua, menghasilkan kesimpulan benar. Artinya pada Tabel 4 ini diberi asumsi jika salah satu atau kedua kombinasi gabungan citra uji dari citra uji gabungan terpanggil pada motif dasar dominan *cluster* pemenang (*cluster* dengan rata-rata selisih jarak terpendek), maka kesimpulannya adalah benar. Selain itu, akan diklasifikasikan sebagai salah.

VI. KESIMPULAN

Berdasarkan implementasi dan hasil pengujian aplikasi yang telah dilakukan, maka peneliti dapat menyimpulkan bahwa :

1. Penelitian ini telah menghasilkan sebuah aplikasi klasifikasi citra batik besurek yang dapat digunakan sebagai aplikasi pengenalan, temu kembali citra dengan selisih jarak 0 untuk citra “*training*”, dan sekaligus sebagai klasifier dengan menggunakan ekstraksi fitur tekstur *Point Minutiae* dan jaringan syaraf tiruan *Self Organizing Map* (SOM) dengan baik. Hasil uji dengan menggunakan citra *non-“training”* memiliki nilai *recall* tertinggi mencapai 0.905 untuk citra batik besurek yang belum pernah diujikan, dan nilai *precision* tertinggi 0.528 untuk citra batik yang belum pernah dilatihkan, MRR 0.853 untuk citra yang belum pernah diujikan. Citra “*training*” yang dimaksud adalah 35 sampel citra yang terdiri dari 5 motif masing – masing 7 varian.
2. Aplikasi pengklasifikasian batik besurek dengan menerapkan *Point Minutiae* dan jaringan syaraf

tiruan *Self Organizing Map* (SOM) untuk 35 citra “*training*” yang dijadikan sebagai citra uji menghasilkan tingkat pencocokan temu kembali 100%.

3. Aplikasi pengklasifikasian batik besurek dengan menerapkan *Point Minutiae* dan jaringan syaraf tiruan *Self Organizing Map* (SOM) untuk citra batik besurek yang belum pernah dilatihkan menghasilkan nilai rata-rata *recall* tertinggi oleh motif relung paku 0.905, dan *recall* terendah diperoleh motif rembulan, yaitu 0.4286. Nilai rata-rata *precision* tertinggi diperoleh motif rembulan, yaitu 0.519 dan *precision* terendah diperoleh motif burung kua, yaitu 0.132. Nilai MRR tertinggi diperoleh motif rembulan, yaitu 0.83 dan MRR terendah diperoleh motif relung paku, yaitu 0.195.
4. Aplikasi pengklasifikasian batik besurek dengan menerapkan *Point Minutiae* dan jaringan syaraf tiruan *Self Organizing Map* (SOM) untuk citra non-batik besurek diperoleh nilai selisih jarak terpendek oleh citra Batik_mendung.jpg, yaitu 415.743 dan nilai selisih jarak terjauh diperoleh citra Batik_Madura1.jpg, yaitu 3,245.569.
5. Aplikasi pengklasifikasian batik besurek dengan menerapkan *Point Minutiae* dan jaringan syaraf tiruan *Self Organizing Map* (SOM) untuk citra gabungan diperoleh rasion keberhasilan 60% .
6. Tidak bisa ditetapkan suatu standar rata-rata selisih jarak dalam cluster untuk memisahkan antara satu motif citra dengan motif citra lainnya, untuk mengklasifikasikan citra batik besurek.

Aplikasi pengklasifikasian batik besurek ini mampu melakukan klasifikasi motif dasar dominan batik dengan motif relung paku sebagai motif yang

memiliki tingkat relevansi tertinggi (rata-rata *recall* 0.905) dan motif rembulan sebagai motif dengan tingkat relevansi terendah (rata-rata *recall* 0.4286). Motif relung paku memiliki nilai *recall* tertinggi, karena motif dasarnya paling jauh berbeda dari motif lainnya. Sedangkan nilai *recall* motif rembulan rendah karena motif yang dimilikinya mempunyai kemiripan dengan motif batik besurek lainya, yaitu raflesia. Selain itu motif rembulan tidak memiliki motif pasti atau terdapat banyak variasi, sehingga sering terklasifikasi ke dalam motif raflesia.

VII. SARAN

Berdasarkan analisis perancangan aplikasi, implementasi dan pengujian yang dilakukan, maka untuk pengembangan penelitian selanjutnya penulis menyarankan sebagai berikut:

1. Aplikasi klasifikasi citra batik besurek ini dapat diselidiki lebih jauh dengan menggunakan topologi lain pada *Self Organizing Map* (SOM) seperti *linear array*, *rectangular grid*, atau *hexagonal grid*. Pada aplikasi ini peneliti mencoba menerapkan topologi *randtop*, karena persebaran pola titik pada citra batik besurek bersifat acak atau tidak beraturan, sehingga topologi yang sesuai adalah topologi *randtop*.
2. Aplikasi ini dapat terus dikembangkan lebih lanjut dalam hal metode yang digunakan, ke depannya diharapkan untuk dapat menggunakan metode selain *Self Organizing Map* (SOM) yang merupakan metode *unsupervised learning* karena hasil dari metode ini kurang memuaskan. Untuk itu dapat digunakan metode *supervised learning*

seperti SVM atau bisa menggunakan model bertingkat (*cascaded model*)

3. Untuk menaikkan persentase keberhasilan sistem pengklasifikasian batik besurek ini, bisa dilakukan dengan dengan memperbanyak sampel citra dengan *instance* yang sama, sehingga motif-motif sejenis tapi berbeda *style* akan terkelompok dalam *cluster* walaupun tidak dalam *cluster* besar, tetapi akan menjadi *cluster-cluster* yang mewakili setiap *style*.

REFERENSI

- [1] Asmadin, dkk. 2012. *Pengelompokan Habitat Dasar Perairan Dangkal Berbasis Data Satelit Quickbird Menggunakan Algoritma Self Organizing Map*. Jurnal Aquasains, Tahun 2012, Vol 1, No.1, Halaman 9-16.
- [2] Karimah, Fatin U. 2014. *Skripsi Rancang Bangun Aplikasi Pencarian Citra Batik Besurek Berbasis Tekstur Dengan Metode Gray Level Cooccurrence Matrix dan Euclidean Distance*. Bengkulu : Fakultas Teknik Program Studi Teknik Informatika Universitas Bengkulu.
- [3] Larose, D. 2004. *Discovering Knowledge in Data: An Introduction to Data mining*. USA: John Wiley & Sons Inc.
- [4] Niam, Bahrin., Hadiyatno, Ahmad., Isnanto, Rizal., 2010. *Analisis Deteksi Tepi Citra Berdasarkan Perbaikan Kualitas Citra*. Jurnal Undergraduated Thesis. Surabaya : Institut Teknik Surabaya.
- [5] Putra, D. 2010. *Pengolahan Citra Digital*. Yogyakarta: Andi.
- [6] Silalahi, U. 2012. *Metode Penelitian Sosial*. Bandung: PT. Refika Aditama.
- [7] Wahyu, H. 2013. *Perancangan Sistem Perangkat Lunak Untuk Mengklasifikasikan Motif Menggunakan Metode Klasifikasi K-Nearest Neighbour (KNN)*.
- [8] Wahyumianto, Arga., Purnama, I.K.E., Christyowidiasmoro. 2012. *Identifikasi Tumbuhan Berdasarkan Minutiae Tulang Daun Menggunakan SOM Kohonen*. Jurnal Undergraduated Thesis Electrical Engineering, Tahun 2012.
- [9] Yulianto, R., Suprpto, Y.K., Hariadi, M. 2010. *Identifikasi Pola Batik Parang Dengan Algoritma Point Minutiae Menggunakan Metode K-Means Clustering*. Jurnal Java Journal, Tahun 2010, Vol 8, No.2.