ANALISIS ALGORITMA BACKPROPAGATION LEVENBERG MARQUARDT UNTUK PENGENALAN MOTIF KAIN BATIK

Roni Ahdiat[1]

[1] Teknik Informatika - Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
E-mail : roni.ibex@gmail.com[1]

ABSTRAK

Batik merupakan budaya yang telah lama berkembang dan dikenal oleh masyarakat Indonesia. Motif batik di setiap daerah memiliki keunikan dan ciri khas masing-masing, baik dalam ragam hias maupun tata warna batik itu sendiri. Salah satu cara untuk mengidentifikasi motif pada kain batik yaitu melalui pengenalan pola.

Algoritma Levenberg Marquadt merupakan pengembangan dari algoritma *backpropagation*. Algoritma ini dibangun untuk mengatasi beberapa kekurangan yang ada pada algoritma *backpropagation* dengan memanfaatkan teknik optimisasi numerik standar yaitu menggunakan pendekatan matriks *Jacobian.* Tujuan dari Levenberg Marquadt adalah meminimalkan total *error.*

Pengujian dari algoritma Levenberg Marquardt dilakukan dengan metode *cross validation*. Data yang digunakan pada proses pelatihan berjumlah 50 buah motif dasar dengan masing-masing motif 10 buah dan 45 buah motif campuran dengan masing-masing motif 5 buah. Sedangkan jumlah motif yang akan dilakukan pengujian yaitu 42 buah, dengan masing-masing motif 3 buah. Dari 4 skenario pengujian yaitu menggunakan 1 sampai 4 *hidden layer*, hasil pengujian menunjukkan akurasi tertinggi sebesar 62.5% dengan akurasi pada batik motif dasar sebesar 80% dan akurasi pada batik motif campuran sebesar 15.79%. Akurasi ini didapatkan dari susunan variabel yang memiliki jumlah *neuron* pada *hidden layer* sebanyak 3 *neuron* dan 4 *neuron*, *learning rate* yang bernilai 0.01, batas epoh sebesar 7000, dan target *error* sebesar 0.035.

Kata Kunci : *backpropagation*, levenberg marquardt, batik, *pattern recognition*

1. PENDAHULUAN

Batik merupakan budaya yang telah lama berkembang dan dikenal oleh masyarakat Indonesia. Secara etimologi kata batik berasal dari bahasa Jawa, yaitu ”tik” yang berarti titik / matik (kata kerja, membuat titik) yang kemudian berkembang menjadi istilah ”batik” [1]. Motif batik di setiap daerah memiliki keunikan dan ciri khas masing-masing, baik dalam ragam hias maupun tata warna batik itu sendiri. Salah satu cara untuk mengidentifikasi motif pada kain batik yaitu melalui pengenalan pola.

Algoritma Levenberg Marquadt merupakan pengembangan dari algoritma *backpropagation*. Algoritma ini dibangun untuk mengatasi beberapa kekurangan yang ada pada algoritma *backpropagation* dengan memanfaatkan teknik optimisasi numerik standar yaitu menggunakan pendekatan matriks *Jacobian.* Tujuan dari Levenberg Marquadt adalah meminimalkan total *error* [8]. Algoritma Levenberg Marquardt telah diterapkan pada pengenalan pola wajah yang dilakukan oleh Mohammed Alwakeel [7] dengan membandingkan algoritma Levenberg Marquardt dengan algoritma Haar Wavelet Transform dan algoritma Principal Component Analysis. Dari penelitian tersebut terbukti algoritma Levenberg Marquardt lebih akurat, cepat, dan stabil dalam pengenalan pola wajah dibandingkan algoritma Haar Wavelet Transform dan algoritma Principal Component Analysis. Pada penelitian lain dilakukan oleh Tulus Bangkit Pratama mengenai pengenalan pola warna citra *google maps* menggunakan algoritma Levenberg Marquardt [17] dengan hasil algoritma ini telah berhasil membuat pola citra baru yang diidentifikasikan sebagai daerah daratan dan bukan daratan.

Dengan latar belakang masalah ini, maka akan dilakukan penelitian mengenai algoritma Levenberg Marquardt yang akan diimplementasikan untuk mengidentifikasi motif batik tidak hanya motif batik dasar, dapat juga mengidentifikasi motif batik khusus atau campuran. Pada penelitian ini juga akan diketahui tingkat akurasi penerapan algoritma Levenberg Marquardt terhadap pengenalan motif batik.

1.1 Kecerdasan Buatan

Kecerdasan Buatan atau AI (*Artificial Intelligence*) merupakan suatu ilmu yang mempelajari bagaimana membuat komputer

melakukan sesuatu pada suatu kejadian atau peristiwa yang mana orang melakukannya dengan baik. AI merupakan proses di mana peralatan mekanik dapat melaksanakan kejadian kejadian dengan menggunakan pemikiran atau kecerdasan seperti manusia [11].

1.2 Jaringan Syaraf Tiruan

Jaringan syaraf tiruan (JST) adalah sistem pengolahan informasi yang didasari fisolofi struktur perilaku syaraf makhluk hidup [13]. Dengan begitu, jaringan syaraf tiruan tak diprogram selayaknya mekanisme pada komputer digital konvensional. Begitu juga dari segi arsitekturnya. Dalam arsitekturnya, jaringan syaraf mempelajari bagaimana menghasilkan keluaran yang diinginkan pada saat diberikan sekumpulan masukan. Proses ini dilakukan secara internal, yaitu dengan memerintahkan sistem untuk mengidentifikasikan hubungan antar masukan kemudian mempelajari respon tersebut. Dengan metoda pensintesisan hubungan, jaringan syaraf dapat mengenal situasi yang sedang dan telah dijumpai sebelumnya. Berbeda dengan proses internal, proses eksternal lebih tergantung pada aplikasinya. Sistem bisa menggunakan umpan balik eksternal atau sinyal tanggapan yang diinginkan, untuk membentuk prilaku jaringan. Ini disebut sebagai *supervised learning*. Dengan cara lain, jaringan dapat membangkitkan sinyal tanggapan yang diinginkan sendiri dalam skenario *unsupervised learning*.

1.3 *Algoritma Levenberg Marquardt*

Algoritma Levenberg Marquardt dikembangkan oleh Kenneth Levenberg dan Donald Marquardt, memberikan solusi numerik untuk masalah meminimalkan fungsi nonlinear [16]. Di bidang jaringan syaraf tiruan, algoritma ini cocok untuk pelatihan kecil dan menengah.

Algoritma Levenberg Marquadt merupakan pengembangan dari algoritma *backpropagation*. Algoritma ini dibangun untuk mengatasi beberapa kekurangan yang ada pada algoritma *backpropagation* dengan memanfaatkan teknik optimisasi numerik standar yaitu menggunakan pendekatan matriks *Jacobian*. Tujuan dari Levenberg Marquadt adalah meminimalkan total *error*.

2. ISI PENELITIAN

2.1 Analisis Masalah

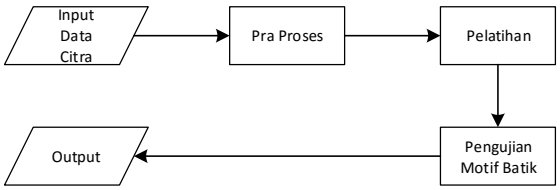
Pada penelitian ini akan dilakukan pengenalan motif kain batik menggunakan algoritma Levenberg Marquardt. Berdasarkan hasil studi literatur mengenai penggunaan algoritma Levenberg Marquardt terhadap pengenalan pola wajah pada jurnal Muhammed Alwakeel dan Zyad Shaaban [7]. Pada jurnal ini membahas tentang pengenalan pola wajah menggunakan algoritma Levenberg Marquardt dan menghasilkan output yang lebih akurat dibandingkan algoritma Haar Wavelet Transform dan Principal Component Analysis. Selain itu, algoritma Levenberg Marquardt belum pernah dilakukan penelitian untuk pengenalan motif batik. Dengan alasan ini, maka penelitian mengenai algoritma Levenberg Marquardt akan diimplementasikan terhadap pengenalan motif kain batik.

2.2 Analisis Sistem

Analisis adalah aktivitas yang memuat sejumlah kegiatan seperti menguraikan sebuah pembahasan. Dalam hal ini pembahasan mengenai pengenalan motif kain batik menggunakan algoritma Levenberg Marquardt

2.3 Analisis Arsitektur Sistem

Arsitektur yang diterapkan terdiri dari data masukan berupa file citra berekstensi *jpg, pra proses, proses pelatihan menggunakan algoritma Levenberg Marquardt, pengujian motif batik menggunakan Levenberg Marquardt, lalu menghasilkan output berupa akurasi. Arsitektur sistem yang digunakan dapat dilihat pada gambar 3.1 di bawah ini.



Gambar 2.1 Arsitektur Umum Sistem

Langkah-langkahnya yaitu :

1. Input data citra merupakan tahapan bahwa data yang dimasukkan adalah berupa file berekstensi *jpg.
2. Pada tahap pra proses, sistem melakukan tahapan yang terdiri dari perubahan warna citra ke *grayscale* untuk memudahkan dalam proses binerisasi. Pada tahap ini pun dilakukan *histogram equalization* untuk memperjelas citra batik sehingga pada proses binerisasi dapat lebih terlihat motif dari citra batik tersebut.
3. Pada tahapan pelatihan, sistem akan melakukan pelatihan untuk pembentukan jaringan syaraf tiruan sebagai data sumber ketika melakukan identifikasi terhadap motif batik yang nantinya akan diuji.
4. Pengujian motif batik, motif kain batik dilakukan pengujian untuk mengetahui jenis batik berdasarkan motifnya.

Output akan dihasilkan berupa akurasi dalam pengujian pengenalan motif batik menggunakan algoritma Levenberg Marquardt.

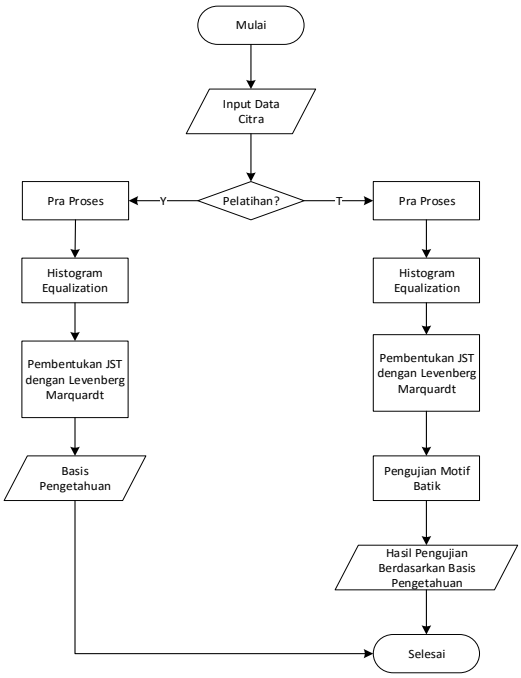
2.4 Analisis Pemrosesan

Proses awal yang menggambarkan alur sistem yang akan dibangun. Tahapan awal sistem adalah pengecekan data masukan. Data masukan yang diperbolehkan merupakan data citra berformat *.jpg. Data masukan ini terdiri atas file citra berupa motif batik yang akan digunakan untuk pembentukan jaringan syaraf tiruan (JST) dan pengujian untuk mengidentifikasi motif batik yang diinput. Kemudian sistem akan menyediakan pilihan menu, yaitu Pelatihan dan Pengujian.

Pada menu pelatihan, data citra yang diinput akan dilakukan tahap pra proses, yaitu mengkonversi data citra ke dalam bentuk grayscale. Lalu untuk memperjelas pola atau motif dari data citra yang sudah berwarna *grayscale*, maka data citra tersebut dilakukan *histogram equalization*. Setelah itu data citra dikonversikan kembali ke dalam bentuk biner sehingga data citra akan menghasilkan matriks bernilai 1 dan 0. Pada tahapan selanjutnya yaitu pembentukan JST menggunakan algoritma Levenberg Marquardt. Setelah itu data disimpan sebagai basis pengetahuan.

Selanjutnya pada menu pengujian prosesnya hampir sama dengan pelatihan, yaitu melakukan konversi data citra ke dalam bentuk grayscale, melakukan *histogram equalization* untuk memperjelas pola atau motif citra, selanjutnya pembentukan JST menggunakan algoritma Levenberg Marquardt. Setelah JST terbentuk, maka data untuk pengujian akan dibandingkan dengan data yang ada pada basis pengetahuan sehingga dapat menghasilkan output berupa jenis batik beserta tingkat akurasinya.

Secara garis besar, flowchart aplikasi pengenalan motif kain batik ini adalah sebagai berikut.



Gambar 2.2 Flowchart Sistem

2.5 Analisis Data Masukan

Tahapan ini menjelaskan mengenai data masukan dan tata cara penulisan data yang diperbolehkan sistem. Data masukan yang diperbolehkan merupakan data citra berformat *.jpg. Data masukan ini terdiri atas file citra berupa motif batik yang akan digunakan untuk pembentukan JST dan pengujian untuk mengidentifikasi motif batik yang diinput

2.6 Analisis Pra Proses

2.6.1. Resizing

Untuk merubah ukuran citra batik, dapat menggunakan fungsi resize() pada matlab. Pada contoh ini, data citra batik dirubah ke dalam ukuran 7 x 5 piksel. Jika dilihat ke dalam bentuk matriks akan menghasilkan 3 dimensi, yaitu 7 x 5 x 3 karena data citra masih berbentuk RGB (*Red Green Blue*).

2.6.2. Grayscale

Pada proses ini matriks citra RGB akan dikonversi ke dalam bentuk *grayscale*. Untuk mengkonversi matriks citra RGB ke *grayscale* dapat dilakukan dengan persamaan berikut.

$$\text{Grayscale} = (R+G+B)/3 \quad (2.1)$$

2.6.3. Histogram Equalization

Konsep dasar dari *histogram equalization* adalah dengan men-*strecth* histogram, sehingga perbedaan piksel menjadi lebih besar atau dengan kata lain informasi menjadi lebih kuat sehingga mata dapat menangkap informasi yang disampaikan. Persamaan yang digunakan untuk HE yaitu persamaan (2.2).

$$S_k = \frac{(L-1)}{n}\sum_{j=0}^{k} n_j \quad (2.2)$$

S_k = output HE
k = nilai yang muncul dalam citra
L = derajat keabuan
n = jumlah seluruh piksel dalam citra
n_j = jumlah yang muncul di setiap nilai k

2.7 Analisis Pelatihan

2.7.1. Inisialisasi Bobot Menggunakan Nguyen Widrow

Nguyen Widrow merupakan modifikasi sederhana bobot-bobot dan bias ke unit tersembunyi yang mampu meningkatkan kecepatan jaringan dalam proses pelatihan jaringan. Metode ini secara sederhana dapat diimplementasikan dengan prosedur sebagai berikut.

1. Tetapkan :

$$\beta = 0.7(p)^{1/n} \quad (2.3)$$

dimana
n = jumlah neuron pada lapisan input
p = jumlah neuron pada lapisan tersembunyi
β = faktor penskalaan

2. Kerjakan untuk setiap unit pada lapisan tersembunyi (j = 1, 2…., p).
   a. Inisialisasi bobot-bobot dari lapisan input ke lapisan tersembunyi:
   b. Hitung: $||V_j||$
   c. Set bias :
      $b1_j$ = bilangan *random* antara -β sampai β.

2.7.2. Pembentukan Jaringan Syaraf Tiruan

Tahap-tahap pembentukan JST dapat dilakukan dengan cara berikut [14].

1. Inisialisasi bobot = 0.
2. Menetapkan maksimum epoh, target error, dan learning rate
3. Inisialisasi Epoh = 0, MSE = 1.
4. Kerjakan langkah-langkah berikut selama (Epoh < maksimum epoh) dan (MSE > target error):
   a. Epoh = Epoh + 1
   b. Untuk tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, kerjakan:
      - Tiap-tiap unit input (x_i, i=1,2,3…,n) menerima sinyal x_i dan meneruskan sinyal tersebut ke semua unit pada lapisan yang ada di atasnya (lapisan tersembuyi)
      - Tiap-tiap unit pada suatu lapisan tersembunyi (z_j, j=1,2,3…,p) menjumlahkan sinyal-sinyal input terbobot:

$$z_in_j = b1_j + \sum_{i=1}^{n} x_i\, v_{ij} \qquad (2.4)$$

Gunakan fungsi aktivasi untuk menghitung sinyal outputnya:

$z_j = f(z_in_j)$ (2.5)

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit output)

- Tiap-tiap unit output (y_k, k=1,2,3,…,m) menjumlahkan sinyal-sinyal input terbobot.

$$y_in_k = b2_k + \sum_{i=1}^{p} z_i\, w_{jk} \qquad (2.6)$$

Gunakan fungsi aktivasi untuk menghitung sinyal outputnya:

$y_k = f(y_in_k)$ (2.7)

dan kirimkan sinyal tersebut ke semua unit di lapisan atasnya (unit-unit output)

- Tiap-tiap unit output (yk, k=1,2,3,…,m) menerima target pola yang berhubungan dengan pola input pembelajaran, hitung informasi errornya:

$\delta 2_k = (t_k - y_k)\, f'(y_in_k)$ (2.8)

$\varphi 2_{jk} = z_j$ (2.9)

$\beta 2_k = \delta_k$ (2.10)

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai w_{jk}):

$\Delta w_k = \alpha\, \varphi 2_{jk}$ (2.11)

hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai $b2_k$):

$\Delta b2_k = \alpha \beta 2_k$ (2.12)

Langkah di atas juga dilakukan sebanyak jumlah lapisan tersembunyi, yaitu menghitung informasi error dan suatu laposan tersembunyi ke lapisan tersembunyi sebelumnya.

- Tiap-tiap unit tersembunyi (z_j, j=1,2,3,…,p) menjumlahkan delta inputnya (dari unit-unit berada pada lapisan di atasnya):

$$z_in_j = \sum_{k=1}^{m} \delta 2_k\, w_{jk} \qquad (2.13)$$

Kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi error: (2.5)

$\delta 1_j = \delta_in_j\, f'(z_in_j)$ (2.14)

$\varphi 1_{ij} = \varphi 1_j\, x_j$ (2.15)

$\beta 1_j = \delta 1_j$ (2.16)

kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai v_{ij}):

$\Delta v_{ij} = \alpha\, \varphi 1_{ij}$ (2.17)

Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai $b1_j$):

$\Delta b1_j = \alpha \beta 1_j$ (2.18)

- Tiap-tiap unit output (Y_k, k = 1,2,3,…,m) memperbaiki bias dan bobotnya (j=0,1,2,…,p):

$w_{jk}(baru) = w_{jk}(lama) + \Delta w_{jk}$ (2.19)

$b2_k(baru) = b2_k(lama) + \Delta b2_k$ (2.20)

tiap-tiap unit tersembunyi (z_j, j=1,2,3,…,p) memperbaiki bias dan bobotnya (i=0,1,2,…,n):

$v_{ij}(baru) = v_{ij}(lama) + \Delta\, v_{ij}$ (2.21)

$b1_j(baru) = b1_j\,(lama) + \Delta\, b1_j$ (2.22)

Hitung MSE

2.7.3. Algoritma Levenberg Marquardt

Pada algoritma *backpropagation* proses *update* bobot dan bias menggunakan *negative gradient descent* secara langsung, sedangkan algoritma Levenberg Marquardt menggunakan pendekatan matrik Hesian (H) yang dapat dihitung dengan:

$H = J^T J$ (2.23)

Sedangkan *gradient* dapat dihitung dengan:

$g = J^T e$ (2.24)

Dalam hal ini J merupakan sebuah matrik *jacobian* yang berisikan turunan pertama dari *error* jaringan terhadap bobot dan bias jaringan. Perubahan pembobot dapat dihitung dengan:

$\Delta W = [J^T J + \mu I] - J^T e$ (2.25)

Sehingga perbaikan pembobot dapat ditentukan dengan:

$W = W + \Delta W$ (2.26)

$W = W + [J^T J + \mu I] - J^T e$ (2.27)

Keterangan :

W = fungsi bobot-bobot jaringan dan bias

e = vektor yang menyatakan semua error pada output jaringan

μ = konstanta learning

I = matrik identitas

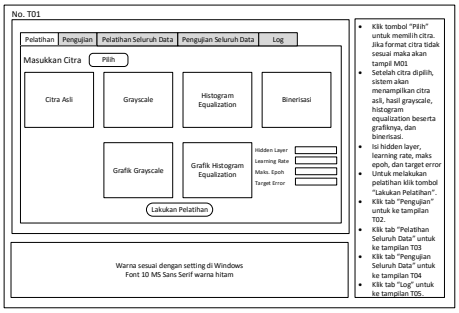
2.8 Analisis Pengujian

Pada tahapan ini yang dilakukan hanya sampai tahap maju saja, tidak ada tahap mundur dan tahap modifikasi bobot. Bobot-bobot yang digunakan merupakan bobot akhir dari hasil pelatihan sebelumnya.

2.9 Perancangan

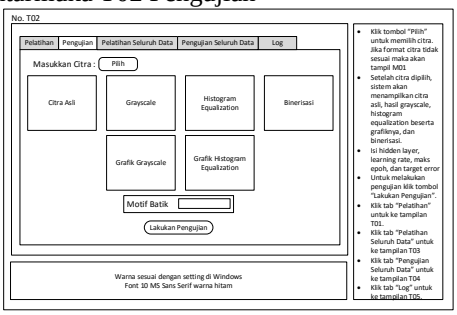
2.11.1. Perancangan Antarmuka

a. Antarmuka T01 Pelatihan



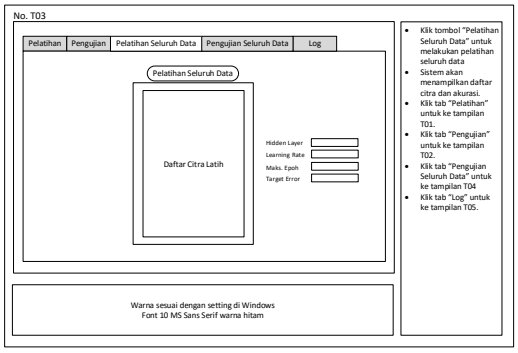
Gambar 2.3 Tampilan Antarmuka Pelatihan

b. Antarmuka T02 Pengujian



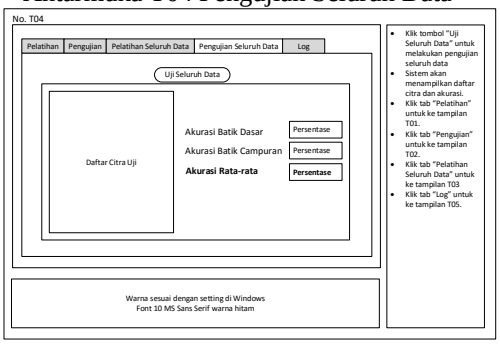
Gambar 2.4 Tampilan Antarmuka Pengujian

c. Antarmuka T03 Pelatihan Seluruh Data



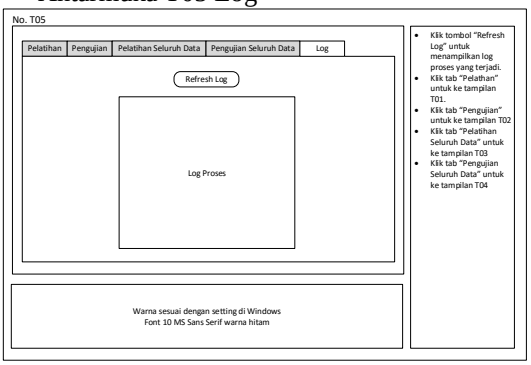
Gambar 2.5 Tampilan Antarmuka Pelatihan Seluruh Data

d. Antarmuka T04 Pengujian Seluruh Data



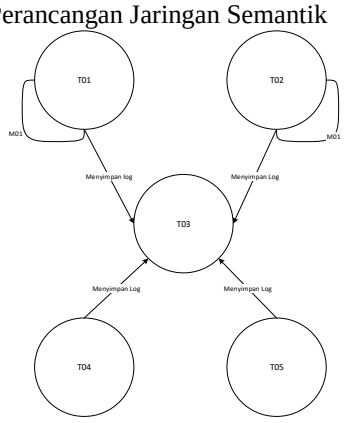
Gambar 2.6 Tampilan Antarmuka Pengujian Seluruh Data

e. Antarmuka T05 Log



Gambar 2.7 Tampilan Antarmuka *Log*

2.11.2. Perancangan Jaringan Semantik



Gambar 2.8 Perancangan Jaringan Semantik

3. PENUTUP

Dari hasil penelitian yang telah dilakukan dapat diperoleh kesimpulan sebagai berikut.

1. Algoritma Levenberg Marquardt dapat diterapkan untuk pengenalan motif kain batik dasar dan motif campuran.
2. Pengenalan motif kain batik dasar dan campuran menggunakan algoritma Levenberg Marquardt memiliki akurasi tertinggi sebesar 62.5% dengan akurasi pada batik motif dasar sebesar 80% dan akurasi pada batik motif campuran sebesar 15.79%. Akurasi ini didapatkan dari susunan variabel yang memiliki jumlah *neuron* pada *hidden layer* sebanyak 3 *neuron* atau 4 *neuron, learning rate* yang bernilai 0.01, batas epoh sebesar 7000, dan target *error* sebesar 0.035.

Sedangkan sampel data yang diuji pada penelitian ini berjumlah 137 buah. Sampel data yang digunakan pada proses pelatihan berjumlah 50 buah motif dasar dengan masing-masing motif 10 buah dan 45 buah motif campuran dengan masing-masing motif 5 buah. Jumlah motif yang dilakukan pengujian yaitu 42 buah, dengan masing-masing motif 3 buah.

Dari hasil penelitian ini dapat diberikan saran-saran yaitu untuk mengidentifikasi motif batik dasar atau campuran dapat menggunakan algoritma lain sehingga bisa mendapatkan algoritma yang cocok untuk pengenalan motif batik dasar maupun campuran.

DAFTAR PUSTAKA

[1] Anas, B. 1997. *Indonesia Indah "Batik"*. Jakarta: Yayasan Harapan Kita/BP 3 TMII.

[2] Ariesto Hadi Sutopo. 2002. *Analisis dan Desain Berorientasi Objek.* Penerbit J & J Learning : Yogyakarta.

[3] Bernardius Arisandi. 2011. *Pengenalan Motif Batik Dengan Rotated Wavelet Filter dan Neural Network*. Tugas Akhir periode Juli 2011. Surabaya: Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember.

[4] Doellah, S. 2008. *Batik "Pengaruh Zaman dan Lingkungan"*. Bandung: Danar Hadi.

[5] E. Alpaydin. 2010. *Introduction to Machine Learning*, The MIT Press.

[6] Johanes Widagdho Yodha. 2014. *Pengenalan Motif Batik menggunakan Deteksi Tepi Canny dan K-Nearest Neighbor.* Proceddings. Techno.COM, Vol. 13, No. 4, November 2014: 251-262. Semarang: Universitas Dian Nuswantoro.

[7] Mohammed Alwakeel. 2010. *Face Recognition Based on Haar Wavelet Transform and Principal Component Analysis via Levenberg-Marquardt Backpropagation Neural Network*. Proceddings. European Journal of Scientific Research, vol 42(1), 25-31

[8] Nurhayati, John Adler, Sri Supatmi. 2013. *Pengelompokkan Citra Warna Menggunakan Jaringan Syaraf Tiruan Dengan Software Matlab.* Tugas Akhir. Bandung: Teknik Komputer Universitas Komputer Indonesia

[9] Nazarudin Ahmad dan Arifyanto Hadinegoro. 2012. *Metode Histogram Equalization untuk Perbaikan Citra Digital.* Proceddings. Seminar Nasional Teknologi Informasi & Komunikasi Terapan 2012 (Semantik 2012). Semarang : Teknik Informatika Universitas Atma Jaya Yogyakarta.

[10] Rocky Yefrenes Dillak. 2012. *Pemanfaatan Algoritma Jaringan Syaraf Tiruan Levenberg Marquardt Untuk Mendeteksi Penyakit Alzheimer*. Proceddings. Seminar Nasional Informatika 2012 (semnasIF 2012) UPN "Veteran" Yogyakarta, 30 Juni 2012 Yogyakarta: Universitas Amikom.

[11] Rinaldi Munir. 2004. *Pengolahan Citra Digital dengan Pendekatan Algoritmik*. Penerbit Informatika : Bandung.

[12] Rosa A.S dan M. Shalahuddin. 2013. *Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek*. Penerbit Informatika: Bandung.

[13] Siswanto. 2005. *Kecerdasan Tiruan*. Penerbit Graha Ilmu : Yogyakarta

[14] Sri Kusumadewi, 2004. *Membangun Jaringan Syaraf Tiruan Menggunakan MATLAB & Exel Link*. Penerbit Graha Ilmu : Yogyakarta

[15] Sunda Ariana, Margareta Andriani, dan Andri. 2013. *Prototipe Perangkat Lunak Analisis Kesalahan Berbahasa Dalam Karya Ilmiah Berbahasa Indonesia*. Proceddings. Seminar Nasional Pendidikan "Mewujudkan Pendidikan Berkualitas Melalui Kerangka Kualifikasi Nasional Indonesia. Palembang : Universitas PGRI.

[16] Muhammad Arhami dan Anita Desiani. 2005. *Pemrograman MATLAB.* Penerbit Andi : Yogyakarta.

[17] Tulus Bangkit Pratama, John Adler. 2013. *Identifikasi Pola Warna Citra Google Maps Menggunakan Jaringan Syaraf Tiruan Metode Levenberg Marquardt Dengan Matlab Versi 7.8*. Tugas Akhir Tahun 2013. Bandung : Jurusan Teknik Informatika, Fakultas Teknik dan Ilmu Komputer, Universitas Komputer Indonesia.

[18] Wilamowski, Bogdan M., dan Yu, Hao. 2011. *Levenberg-Marquardt Training.* Industrial Electronics Handbook, 2nd Edition 5, 12-1 s/d 12-15.

[19] Witten, Ian. H. 2011. *Data Mining Practical Machine Learning Tools and Technique*. (3rd edition). New York: Morgan Kauffman.