

Identifikasi Pola Warna Citra Google Maps menggunakan Jaringan Syaraf Tiruan Metode Levenberg -Marquardt dengan MatLab Versi 7.8

Identification of the Google Maps Image Color Pattern using the Levenberg - Marquardt Artificial Neural Network with MatLab Version 7.8

J Adler*, T B Pratama²

¹)Program Studi Teknik Komputer, Fakultas Teknik dan Ilmu Komputer, Universitas Komputer Indonesia
Jl. Dipati Ukur No. 112 - 116, Bandung, Indonesia 40132

²)Program Studi Sistem Komputer, Fakultas Teknik dan Ilmu Komputer, Universitas Komputer Indonesia
Jl. Dipati Ukur No. 112 - 116, Bandung, Indonesia 40132

*email: john.adler@email.unikom.ac.id

ABSTRACT – Research on Artificial Neural Networks has been widely implemented in scientific fields, especially Geography, which identifies map image patterns to distinguish geographical features, and is applied to Google Maps. Image processing, by improving image quality so that it is easily interpreted by humans or machines (computers). The technique used is the Segmentation technique. The training method uses the Levenberg-Marquardt algorithm. The results of the training process obtained epoch values of 17 out of 100 iterations. The time used during the training process is 6 seconds, the performance value = 2.06; gradient value = 8.49; Mu = 1.00; check validation = 6.0; and the regression value (R) of the training process and the whole process of 0.95889 and 0.91483 respectively. The regression value shows a linear relationship between the output network and the target. Pattern identification has succeeded in creating a new image pattern identified as land area and not land. Overall, the application made successfully fulfills the desired function, namely to help people with color blindness.

Keywords – Artificial Neural Networks, Image Processing, Segmentation Techniques, Levenberg-Marquardt

ABSTRAK – Riset mengenai Jaringan Syaraf Tiruan telah banyak diimplementasikan dalam bidang keilmuan, khususnya Geografi, yang mengidentifikasi pola citra peta untuk membedakan kenampakan geografis, dan diterapkan pada Google Maps. Proses pengolahan citra, dengan memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau mesin (komputer). Teknik yang digunakan adalah teknik Segmentasi. Metode training menggunakan algoritma Levenberg-Marquardt. Hasil penelitian pada proses training diperoleh nilai epoch sebanyak 17 dari 100 iterasi. Waktu yang digunakan selama proses training sebanyak 6 detik, nilai performance = 2,06; nilai gradient = 8,49; Mu = 1,00; validation check = 6 dan nilai regresi (R) proses training dan keseluruhan proses berturut-turut sebesar 0.95889 dan 0.91483. Nilai regresi menunjukkan adanya hubungan linier antara output jaringan dengan target. Identifikasi pola telah berhasil membuat pola citra baru yang diidentifikasi sebagai daerah daratan dan bukan daratan. Secara keseluruhan, aplikasi yang dibuat berhasil memenuhi fungsi yang diinginkan, yaitu untuk membantu penderita buta warna.

Kata Kunci – Jaringan Syaraf Tiruan, Pengolahan Citra, Teknik Segmentasi, Levenberg-Marquardt

1. PENDAHULUAN

Pada peta dunia terdapat simbol warna untuk kenampakan geografis dan pada setiap warna memiliki makna tertentu. Misalnya kenampakan hipsografi atau relief muka bumi, menggunakan

warna dasar coklat, dari coklat muda sampai coklat tua, kenampakan hidrografi atau wilayah perairan (sungai, danau, laut), menggunakan warna dasar biru, dari biru muda (hampir putih) sampai biru tua (kehitaman) dan Kenampakan vegetasi (hutan,

perkebunan), menggunakan warna dasar hijau [1].

Warna hijau juga digunakan untuk menggambarkan wilayah dataran rendah. Berdasarkan permasalahan yang diuraikan di atas, maka penulis tertarik untuk mengimplementasikan Jaringan Syaraf Tiruan dengan metode *training Lavenberg-Marquardt* dalam mengidentifikasi pola warna citra *Google Maps* dan membuat aplikasi pengolahan citra untuk mengetahui area daratan dan bukan daratan pada peta *Google Maps*.

Dalam penggunaannya diharapkan aplikasi yang dibuat dapat memberikan informasi yang sesuai dengan tujuan atau kepentingan pengolahan citra dan dapat membantu penderita buta warna untuk mengenali objek-objek yang ada di permukaan bumi melalui gambar peta.

Google maps

Google maps adalah sebuah jasa peta globe virtual gratis dan online disediakan oleh Google yang dapat diakses di <http://maps.google.com>. Google Maps memberikan informasi mengenai peta dan pemetaan suatu daerah, selain itu Google Maps juga mampu memberikan informasi mengenai kepadatan lalu lintas suatu wilayah, bahkan memberikan pencitraan melalui darat ataupun udara

2. METODE DAN BAHAN

Algoritma Levenberg Marquardt

Algoritma Levenberg-marquardt merupakan pengembangan algoritma backpropagation standar. Pada algoritma backpropagation, proses update bobot dan bias menggunakan *Negative Gradient Descent* secara langsung sedangkan, algoritma Levenberg-Marquardt menggunakan pendekatan matrik Hessian (H) yang dapat dihitung dengan [6] :

$$H = J^T \cdot J \quad (1)$$

dimana J^T adalah sebuah *transpose* dari matrik *jacobian* dan J merupakan sebuah matrik *jacobian* yang berisikan turunan pertama dari *error* jaringan terhadap bobot dan bias jaringan.

$$J = \frac{\partial er}{\partial w} \quad (2)$$

∂er turunan pertama *error* jaringan dan ∂w turunan pertama bobot jaringan

Sedangkan gradien (g) dapat dihitung dengan :

$$g = J^T \cdot e \quad (3)$$

dimana e adalah vector yang menyatakan semua *error* pada output jaringan.

Perubahan pembobot (ΔX) dapat dihitung dengan :

$$\Delta X = [J^T \cdot J + \mu I]^{-1} \cdot J^T \cdot e \quad (4)$$

dimana μ adalah konstanta learning dan I adalah sebuah matrik identitas.

Sehingga dapat dihitung fungsi bobot-bobot jaringan dan bias (X) untuk perbaikan pembobot dengan :

$$X = X + \Delta X \quad (5)$$

$$X = X + [J^T \cdot J + \mu I]^{-1} \cdot J^T \cdot e \quad (6)$$

Pengolahan Citra

Pengolahan citra merupakan kegiatan memperbaiki kualitas citra agar mudah diinterpretasi oleh manusia atau mesin (komputer). Masukan dan keluarannya sama-sama citra tapi dengan kualitas citra keluaran lebih baik daripada citra masukan. Misal citra warnanya kurang tajam, kabur (*blurring*), mengandung noise (misal bintik-bintik putih) sehingga perlu ada pemrosesan untuk memperbaiki citra karena citra tersebut menjadi sulit diinterpretasikan karena informasi yang disampaikan menjadi berkurang [8].

Segmentasi

Segmentasi merupakan proses partisi gambar digital ke beberapa daerah dengan tujuan untuk menyederhanakan ataupun merubah representasi gambar menjadi sesuatu yang lebih bermakna dan mudah dianalisa [2], [3]. Ada beberapa metoda yang sering digunakan dalam segmentasi citra antara lain : metode *thresholding*, metode *shapebased*, metode *region growing*, dan metode statistik atau juga disebut metode *clustering*.

Ekstraksi Ciri Orde Dua

Salah satu teknik untuk memperoleh ciri statistik orde dua adalah dengan menghitung probabilitas hubungan ketetanggaan antara dua piksel pada jarak dan orientasi sudut tertentu. Pendekatan ini bekerja dengan membentuk sebuah matriks kookurensi dari data citra, Terdapat berbagai jenis 6 ciri tekstural yang dapat diekstraksi dari matriks kookurensi yaitu *Angular Second Moment*, *Contrast*, *Correlation*, *Variance*, *Inverse Difference Moment*, dan *Entropy*.

Jaringan Syaraf Tiruan (JST)

Jaringan Syaraf Tiruan adalah suatu teknologi komputasi yang berbasis hanya pada model syaraf biologis dan mencoba mensimulasikan tingkah laku dan kerja model syaraf terhadap berbagai macam masukan. Sebagai suatu teknologi komputasi, JST merupakan satu teknik pemrosesan informasi yang

menggunakan model kuantitatif [4].

Syaraf biologis mengilhami terciptanya suatu proses komputasi yang identik dengan kerja neuron dalam sistem syaraf manusia. Seperti halnya jaringan syaraf biologis, model matematik JST menghubungkan sejumlah masukan dan keluaran suatu sistem secara adaptif yang diorganisasikan dalam lapisan elemen pemroses seperti layaknya hubungan antar *neuron* syaraf biologi.

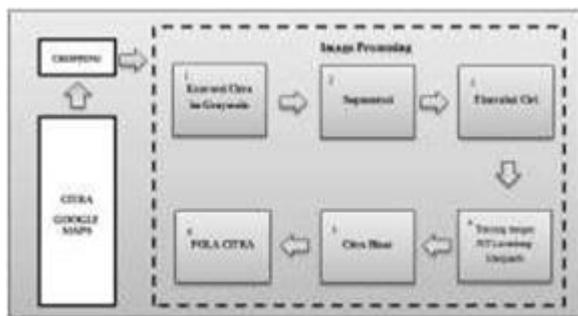
Secara garis besar, proses belajar JST dapat dibagi menjadi dua yaitu :

- JST yang menggunakan paket pelatihan sebagai proses belajar dan dikenal sebagai proses belajar dengan pengawasan.
- JST tanpa paket pelatihan pada proses belajar dan umumnya disebut proses belajar tanpa pengawasan

3. HASIL DAN PEMBAHASAN

Diagram Blok Sistem

Berikut merupakan gambar diagram blok sistem yang dibuat :



Gambar 1. Diagram blok sistem

Proses pengambilan citra

Pengambilan *image* input dilakukan dengan cara *cropping* langsung pada *Google Maps* yang dapat diakses melalui situs <https://maps.google.co.id>. Berikut adalah citra asli dari hasil *cropping* pada *Google maps* :



Gambar 2. Citra asli dari google maps

Pada gambar 2, ukuran *image* input yang akan diolah yaitu *image* yang berukuran 50x512 sampai 384x512 dengan format *image .jpg*.

Sistem Pengolahan Citra, diantaranya [7] :

a. Konversi Citra Grayscale

Pada perancangan sistem pengolahan citra dibutuhkan citra *grayscale*, karena citra warna asal merupakan citra RGB sehingga dibutuhkan teknik filter untuk mengubah nilai masing-masing *pixel* menjadi derajat keabuan dengan teknik *Grayscale filter* yaitu mengubah sebuah gambar berwarna menjadi gambar hitam putih dengan cara mengubah efek warna dari masing-masing *pixel* menjadi derajat keabuan.

b. Segmentasi Citra

Beberapa teknik dan fungsi segmentasi citra yang digunakan pada aplikasi *matlab* adalah teknik *thresholding*, fungsi *entropy* dan fungsi *strel*. Fungsi *entropy* adalah nilai statistik yang dapat digunakan untuk mengkarakterisasi tekstur citra *input*, karena sifatnya yang diskrit (nilai piksel tidak dalam bentuk pecahan) fungsi ini sesuai untuk mencari citra biner yang memiliki nilai '0' dan '1'.

Selanjutnya teknik *thresholding*, *thresholding* merupakan proses pemisahan piksel-piksel berdasarkan derajat keabuan yang dimilikinya. Piksel yang memiliki derajat keabuan lebih kecil dari nilai batas yang ditentukan akan diberikan nilai 0, sementara piksel yang memiliki derajat keabuan yang lebih besar dari batas akan diubah menjadi bernilai 1. Setelah teknik *thresholding* selanjutnya menjalankan fungsi *strel*, proses ini biasanya digunakan untuk kehalusan, pengisian atau menghapus objek dalam suatu citra *grayscale* atau biner.

c. Ekstraksi Ciri Orde Dua

Salah satu teknik untuk memperoleh ciri statistik orde dua adalah dengan menghitung probabilitas hubungan ketetanggaan antara dua piksel pada jarak dan orientasi sudut tertentu. Pendekatan ini bekerja dengan membentuk sebuah matriks *kookurensi* dari data citra, dilanjutkan dengan menentukan ciri sebagai fungsi dari matriks tersebut.

d. Arsitektur Jaringan Syaraf Tiruan

Secara garis besar jalannya proses pembuatan arsitektur Jaringan Syaraf Tiruan dari sistem. Arsitektur jaringan syaraf tiruan dari sistem menggunakan arsitektur multilayer network atau jaringan multilapis. Terdapat tiga lapis layer , yaitu :

1. Input layer sebanyak 1 neuron
2. Hidden layer sebanyak 20 neuron
3. Output layer sebanyak 6 neuron

Metode *training* yang digunakan sistem adalah *Levenberg Marquardt learning*. Sistem melakukan pembelajaran terhadap input dengan target yang sudah ditentukan untuk memperoleh nilai-nilai bobot dan bias yang optimum pada tiap unit *neuron hidden layer* dan *output layer*.

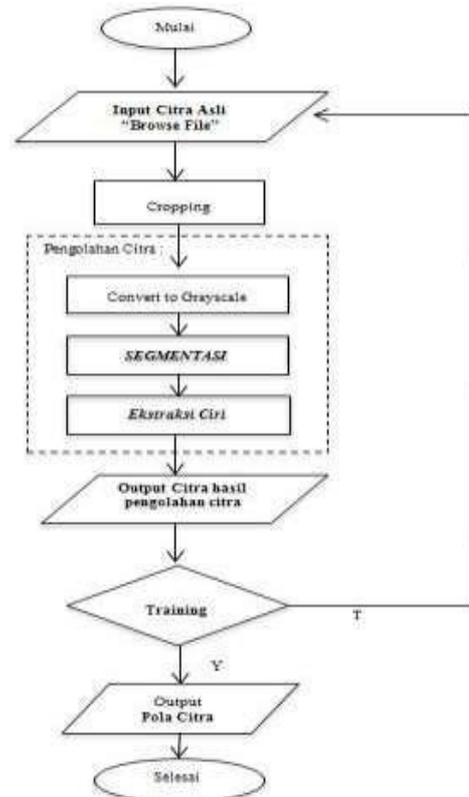
Proses *Training Levenberg-Marquardt*

Dalam rangka mengimplementasikan algoritma *Levenberg-Marquardt* untuk pelatihan jaringan syaraf tiruan, ada satu masalah harus dipecahkan yaitu bagaimana seseorang mengatur pelatihan proses iteratif untuk memperbarui bobot.

Pada bagian ini, pelaksanaan proses *training* dengan algoritma *Levenberg-Marquardt* akan disusun sebagai berikut :

- Menentukan input dan target *training*.**
 Nilai input diperoleh dari proses sebelumnya yaitu nilai dari enam ciri statistik orde dua diantaranya Angular Second Moment (CiriASM), Contrast (CiriCON), Correlation (CiriCOR), Variance (CiriVAR), Inverse Difference Moment (CiriIDM), dan Entropy (CiriENT).
- Pengambilan sampel sebanyak enam kali.**
 Dilakukan dengan cara *cropping* dari *image* hasil pengolahan citra. Kriteria sampel yang diambil adalah dua kali *cropping* daerah yang bernilai '0' (warna hitam), dua kali *cropping* daerah yang bernilai '1' (warna putih) dan dua kali *cropping* daerah yang bernilai '0' dan '1' (warna hitam dan putih). Proses ini bertujuan untuk menentukan enam nilai ciri statistik orde dua dari citra yang akan di *training* melalui perhitungan ciri orde dua.
- Training* citra.**
 Dengan menggunakan jaringan syaraf tiruan yang telah dibuat, sampel di *training* menggunakan jaringan syaraf tiruan *Levenberg-Marquardt (trainlm)*.
- Klasifikasi pola warna citra.**
 Deteksi tekstur pada citra dengan cara mensortir keseluruhan bagian *pixel* citra.

Flowchart Perancangan Aplikasi

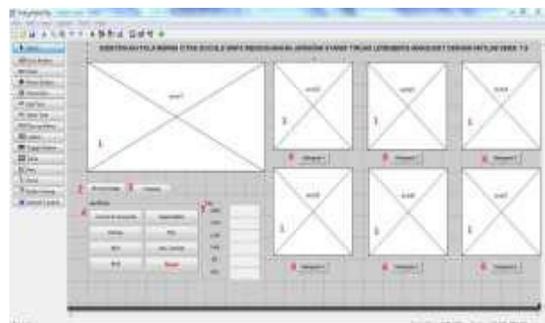


Gambar 3. Flowchart perancangan aplikasi

Dengan *flow chart*, proses dari alur program yang dibuat digambarkan dengan sederhana menggunakan simbol yang mudah untuk dimengerti, berikut *flowchart* dari perancangan aplikasi :

Software Antarmuka Sistem

Berikut merupakan tampilan dari *Software* antarmuka sistem Identifikasi Pola Warna Citra Google Maps Menggunakan Jaringan Syaraf Tiruan *Levenberg-Marquadt* dengan Matlab Versi 7.8.



Gambar 4. Antarmuka sistem

Berikut penjelasan dari tiap komponen pada

sistem yang dibuat pada gambar di atas :

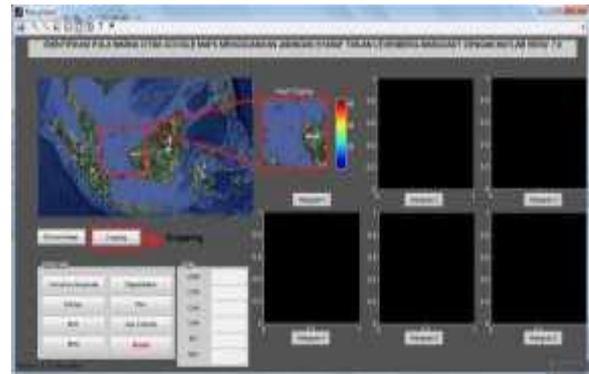
1. *Picture Box*, untuk menampilkan *image*.
2. *Pushbutton Browse Image*, untuk mengambil *File image* yang sudah disimpan.
3. *Pushbutton Cropping*, untuk *cropping image*.
4. Panel Identifikasi, didalamnya terdapat menu-menu proses pengoahan citra diantaranya :
 - a. *Pushbutton Convert to Grayscale*
 - b. *Pushbutton Entropy BW1*
 - c. *Pushbutton BW2*
 - d. *Pushbutton Segmentation*
 - e. *Pushbutton Fitur*
 - f. *Pushbutton Atur Contrast*
 - g. *Pushbutton Reset*
5. Panel Fitur, menampilkan nilai ciri orde dua yang merupakan output dari *pushbutton* Fitur.
6. Tombol Histogram, untuk menampilkan histogram dari *image* yang diolah.
 - a. *Pushbutton* Histogram1: Histogram dari citra hasil *cropping*.
 - b. *Pushbutton* Histogram2: Histogram dari citra *grayscale*.
 - c. *Pushbutton* Histogram3: Histogram dari citra *entropy*.
 - d. *Pushbutton* Histogram4: Histogram dari citra BW1.
 - e. *Pushbutton* Histogram5: Histogram dari citra BW2.
 - f. *Pushbutton* Histogram6: Histogram dari citra hasil segmentasi.

Langkah-langkah pengujian yang dilakukan yaitu setelah program Matlab dijalankan [5] :

Pengujian Program Aplikasi Pengolahan Citra

Berikut adalah langkah-langkah pengujian program aplikasi pengolahan citra yang telah dibuat, yaitu :

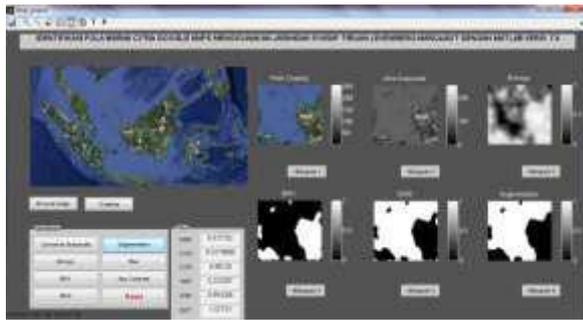
1. Open File, tekan *pushbotton Browse Image* untuk menampilkan citra asli. Citra asli adalah citra yang sebelumnya sudah di capture dari *Google Maps* dan disimpan ke dalam folder file.
2. Lakukan *Cropping* pada citra asli, pilihlah daerah mana yang akan dicari karakteristiknya sebagai sampel citra yang akan diolah dengan menekan *pushbutton Cropping*. Tekan *pushbutton* Histogram1 jika ingin melihat histogram dari hasil *cropping*.



Gambar 5. Cropping citra

3. Tekan *pushbutton Convert to Grayscale* untuk konversi citra RGB menjadi citra *Grayscale*.
4. Tekan *pushbutton* Histogram2 jika ingin melihat histogram dari citra *grayscale*.
5. Tekan *pushbutton Entropy* untuk mengkarakterisasi tekstur citra *grayscale*. Tekan *pushbutton* Histogram3 jika ingin melihat histogram dari citra hasil *entropy*.
6. Tekan *pushbutton* BW1, pada proses ini digunakan teknik *thresholding* untuk mendapatkan citra biner. Tekan *pushbutton* Histogram4 jika ingin melihat histogram dari citra BW1.
7. Selanjutnya, tekan *pushbutton* BW2 untuk menentukan mana daerah daratan dinyatakan oleh nilai '0' dan mana daerah bukan daratan dinyatakan dengan nilai '1'. BW2 merupakan negasi dari BW1 (\sim BW1). Tekan *pushbutton* Histogram5 jika ingin melihat histogram dari citra BW2.
8. Tekan *pushbutton Segmentation* untuk penghalusan citra biner.
9. Tekan *pushbutton* Histogram6 jika ingin melihat histogram dari citra hasil segmentasi.
10. Tekan *pushbutton* Fitur untuk menampilkan nilai ciri orde dua.

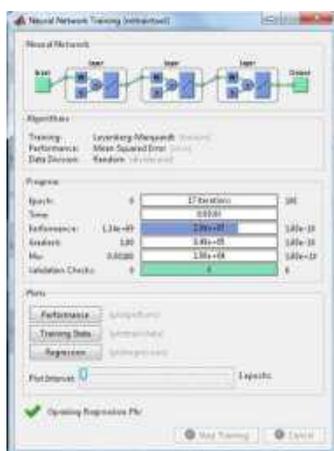
Dari langkah-langkah pengujian diperoleh keluaran, diperlihatkan pada Gambar 6, semua *pushbutton* telah berfungsi dengan baik. Pengujian dilakukan dengan mengambil sampel citra dari salah satu bagian citra asli yang dilakukan melalui proses *cropping* kemudian sampel citra di olah melalui proses pengolahan citra sehingga didapatkan citra biner sebagai output dari proses pengolahan citra tersebut. Dari citra output yang didapat. Langkah selanjutnya adalah proses *training* citra output hasil dari pengolahan citra.



Gambar 6. Antarmuka hasil running aplikasi pengolahan citra

Training citra biner menggunakan jaringan syaraf tiruan levenberg-marquardt

Caranya *running list* program `trainLM.m` pada *menu editor* matlab. List program `trainLM.m` sebelumnya sudah dibuat dan di simpan pada *file folder*. Lalu ambil sampel sebanyak enam kali dengan ketentuan kriteria sampel yang diambil adalah dua kali *cropping* daerah yang bernilai '0' (warna hitam), dua kali *cropping* daerah yang bernilai '1' (warna putih) dan dua kali *cropping* daerah yang bernilai '0' dan '1' (warna hitam dan putih). Setelah itu muncul *window nntraintool*.



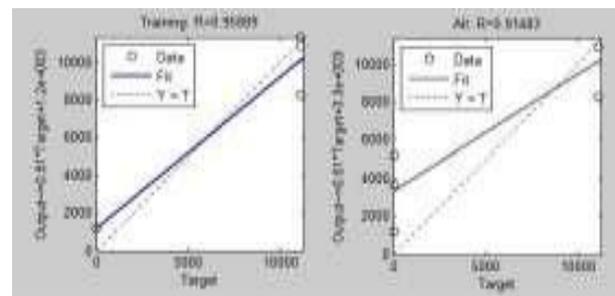
Gambar 7. Window nntraintool

Pada gambar 7 terlihat pada panel *progress* hasil dari *training* citra diatas, diperoleh nilai *epoch* sebanyak 17 *iterations* dari batas *maximum* 100 *iterations*., waktu yang digunakan selama proses *training* sebanyak 6 detik, nilai *performance* = 2,06. , nilai *gradient* = 8,49. , $\mu = 1,00$. dan *validation check* =6. Langkah pengujian selanjutnya mengklasifikasi pola warna citra, dilakukan untuk mengidentifikasi karakteristik pola warna citra dengan cara mensortir keseluruhan bagian *pixel* citra untuk mendeteksi adanya tekstur pada citra.

Pada gambar diatas, proses identifikasi membutuhkan waktu 3 menit 10 detik untuk menyapu ke semua daerah citra dengan ukuran citra $92 \times 102 \text{ pixel}$. Dengan demikian bentuk pola yang diperoleh dapat diklasifikasikan dimana daerah daratan berada di dalam garis dan yang berada diluar garis bukan daerah daratan.

Analisa Hasil Training Jaringan Syaraf Tiruan

Secara desain dan hasil, jaringan syaraf tiruan yang dirancang memang belum mencapai hasil yang terbaik.



Gambar 8. Plot Regression

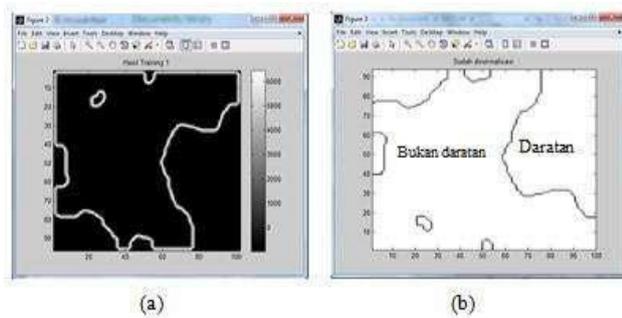
Pada gambar 8 terlihat pada panel *progress* hasil dari *training* citra diperoleh nilai *epoch* sebanyak 17 *iterations* dari batas *maximum* 100 *iterations*., waktu yang digunakan selama proses *training* sebanyak 6 detik, nilai *performance* = 2,06, nilai *gradient* = 8,49, $\mu = 1,00$ serta *validation check* =6.

Dari batas *maximum* 100 iterasi didapat 17 kali iterasi maksudnya pada proses *training* terjadi 17 siklus yang melibatkan seluruh pola data *training*, sebenarnya dalam proses *training* memerlukan banyak iterasi sampai mencapai batas *maximum epoch*. sehingga jumlah *epoch* yang telah ditentukan belum cukup mendapat hasil *performance* jaringan syaraf tiruan yang optimal.

Selanjutnya dalam hasil *training* Jaringan Syaraf Tiruan diperoleh plot regresi, yang menunjukkan hubungan antara *output* jaringan dan target. Jika pelatihan yang sempurna, *output* jaringan dan target akan persis sama, tetapi hubungan tersebut sangat sulit untuk disesuaikan dan harus berulang - ulang dalam proses *training* sampai nilai *output* mendekati nilai target.

Hasilnya ditunjukkan pada gambar 8. Nilai R merupakan indikasi hubungan antara output dan target. Jika $R = 1$, menunjukkan bahwa adanya hubungan linear yang tepat antara output dan target. Jika R mendekati nol, maka tidak ada hubungan linear antara output dan target. Dalam pengujian kali ini, data pelatihan menunjukkan kecocokan. Dalam pengujian kali ini, data pelatihan

menunjukkan kecocokan, karena nilai R pada proses *training* mendekati "1" terlihat pada gambar 8 Nilai R *training* = 0.95889 dan nilai R dari seluruh proses berturut-turut = 0.91483.



Gambar 9. Citra hasil training (a). Sebelum dinormalisasi (b). Sudah dinormalisasi

Proses selanjutnya adalah proses mengklasifikasi pola citra dengan cara mensortir keseluruhan bagian *pixel* citra untuk mendeteksi adanya tekstur pada citra, didapatkan pola citra seperti pada gambar 9.

Terlihat, citra hasil *training* membentuk pola yang dapat diklasifikasikan dimana daerah yang berada di dalam garis merupakan daerah daratan dan yang berada diluar garis merupakan daerah yang bukan daratan. Berdasarkan pengujian hasil ini belum cukup maksimal karena hanya garis tepi saja yang terdeteksi. Selain itu dalam pengujian proses menyapu ke semua daerah citra dibutuhkan waktu 3 menit 15 detik dengan ukuran citra 92 x 102. Hal tersebut menjadi pertimbangan akan kinerja dari sistem program yang telah dibuat, karena prosesnya masih memakan waktu yang cukup lama, telah dilakukan pengujian sebanyak 5 kali untuk mengetahui pengaruh apa saja yang menyebabkan membutuhkan waktu yang cukup lama. Berikut data hasil percobaan.

Tabel 1. Data Klasifikasi Citra

| Percobaan ke- | Ukuran Citra (pixel) | Waktu yang dibutuhkan |
|---------------|----------------------|-----------------------|
| 1 | 92 x 102 | 3 min 15 sec |
| 2 | 77 x 93 | 2 min 45 sec |
| 3 | 58 x 67 | 1 min 20sec |
| 4 | 89 x 98 | 3 min 35 sec |
| 5 | 126 x 145 | 11 min 56 sec |

Pada tabel 1 hasil percobaan yang telah dilakukan memperlihatkan bahwa ukuran citra *input* mempengaruhi lamanya waktu proses klasifikasi, semakin kecil ukuran citra *input* maka semakin sedikit waktu yang dibutuhkan, sebaliknya jika

ukuran citra *input* semakin besar maka semakin banyak waktu yang dibutuhkan.

4. KESIMPULAN

Dari perancangan, pengerjaan, dan analisa yang dilakukan selama penelitian ini, dapat diambil kesimpulan yaitu identifikasi pola warna citra pada peta Google Maps menggunakan aplikasi Jaringan Syaraf Tiruan Lavenberg-Marquadt telah berhasil membuat pola citra baru yang diidentifikasi sebagai daerah daratan dan bukan daratan. Dalam proses *training* diperoleh nilai *epoch* sebanyak 17 iterasi dari maksimal 100 iterasi, waktu yang digunakan sebanyak 6 detik, nilai *performance* sebesar 2,6, nilai *gradient* 8,49, nilai *Mu* serta *validation check* sebanyak 6. Untuk teknik pengolahan citra *Matlab*, sebagai pilihan akhir digunakan teknik segmentasi citra. Aplikasi pengolahan citra menghasilkan citra biner dari proses segmentasi pola warna citra peta Google Maps. Ukuran citra dengan *pixel* tinggi membutuhkan waktu lebih lama untuk diidentifikasi dibandingkan dengan ukuran citra dengan *pixel* yang rendah. Secara keseluruhan, aplikasi yang dibuat berhasil memenuhi fungsi yang diinginkan.

DAFTAR PUSTAKA

- [1] Gunawan, T., *Fakta dan Konsep Geografi untuk SMA/MA kelas XII*, Inter plus, Jakarta, 2007.
- [2] Hermawati, F. A., "*Pengolahan Citra Digital : Konsep dan Teori*", Penerbit Andi, Yogyakarta, 2013.
- [3] Prasetyo, E., *Pengolahan Citra Digital dan Aplikasinya Menggunakan Matlab*, Penerbit Andi, Yogyakarta, 2011.
- [4] H. Demuth, M. Beale, "*Neural Network Tool*", Edisi 6, 2000.
- [5] Adler, J., *Pengaruh Matriks dan Porositas Batuan Karbonat Formasi Parigi*, Majalah Ilmiah Unikom Vol.9, hal 51-60, Maret 2011.
- [6] Rahmat, *Pengenalan Pola Sinyal EKG menggunakan JST dengan Algoritma Levenberg - Marquardt*, Seminar Nasional Pasca Sarjana V ITS Surabaya. 2005.
- [7] Gonzales, R.C., and Woods, R.E, *Digital Image Processing Second edition*, New Jersey: Prentice Hall. 2002.
- [8] Artikel non-personal, 9 Februari 2013, Buta Warna, Wikipedia Bahasa Indonesia, http://id.wikipedia.org/wiki/Buta_warna, diakses 28 februari 2013