

PENGELOMPOKKAN CITRA WARNA MENGGUNAKAN JARINGAN SYARAF TIRUAN DENGAN SOFTWARE MATLAB

Nurhayati¹, John Adler², Sri Supatmi³

^{1,2,3}Teknik Komputer, Universitas Komputer Indonesia (UNIKOM)

¹nuril24@yahoo.com, ²johnadler2zz@yahoo.com, ³srisupatmi@yahoo.com

ABSTRAK

Pengolahan citra adalah salah satu teknik yang digunakan untuk mengimplementasikan perbaikan kualitas suatu citra dan mengolah informasi yang terdapat pada suatu gambar. Tujuan pada Tugas Akhir ini adalah untuk mengelompokkan warna-warna kandungan mineral dari suatu batuan. Warna yang dikenali hanya tiga komponen pembentuk suatu batuan adalah yang bernilai 1 merupakan merah, yang bernilai 2 merupakan putih dan yang bernilai 3 merupakan biru. Pengambilan data diambil dari gambar yang sudah dikompresi menggunakan software microsoft paint kemudian digunakan sebagai data masukan yang akan diolah oleh Jaringan Syaraf Tiruan menggunakan software Matlab. Data masukan pada pelatihan sebelumnya dilakukan cropping sebanyak satu kali pada area yang terdapat warna merah, warna putih dan warna biru. Setelah itu dilakukan proses pembelajaran backpropagation dengan menggunakan fungsi aktivasi sigmoid biner. Dengan tingkat keakurasian pada pengelompokkan warna yang dikenali warna merah, putih, dan biru, berturut-turut 73%, 73%, dan 82%.

Kata Kunci : Pengolahan Citra, Jaringan Syaraf Tiruan, Cropping, Sigmoid Biner.

1. PENDAHULUAN

Citra digital merupakan proses yang bertujuan untuk memanipulasi dan menganalisis citra warna yang banyak melibatkan persepsi visual dengan bantuan komputer. Pengolahan citra bertujuan untuk melihat detail kualitas *image* agar mudah diinterpretasi oleh manusia atau mesin.[4] Pengolahan citra juga dapat diimplementasikan dalam bidang *geofisika* (fisika bumi) yang bertujuan untuk mengetahui letak dan lokasi kandungan mineral bumi dalam tanah. Jaringan Syaraf Tiruan adalah paradigma pemrosesan suatu informasi yang terinspirasi oleh sistem sel syaraf biologi, sama seperti otak yang memproses suatu informasi. Karena memiliki kemampuan mengolah informasi menjadi sebuah citra atau gambar untuk mengetahui letak kandungan mineral dalam tanah. Jenis Jaringan Syaraf Tiruan yang digunakan adalah *Multilayer Perceptron* dengan menggunakan metode Propagasi Balik (*Backpropagation*) dengan algoritma adalah *Levenberg-Marquadt*. Pengolahan data dan pemrograman Jaringan Syaraf Tiruan menggunakan *software* Matlab versi 2010b.

Warna yang akan dikenali hanya tiga warna yang merupakan komponen pembentuk suatu batuan, yaitu : merah, putih, dan biru.

2. LANDASAN TEORI

Pengolahan Citra Digital

Citra digital dapat didefinisikan sebagai fungsi dua variabel, $f(x,y)$, dimana x dan y adalah koordinat spasial dan nilai $f(x,y)$ adalah intensitas citra pada koordinat tersebut. Warna dasar untuk menciptakan dan menampilkan warna pada citra digital berdasarkan pada penelitian bahwa sebuah warna merupakan kombinasi dari tiga warna dasar, yaitu merah, hijau dan biru (*Red, Green, Blue* - RGB).[4]

Jaringan Syaraf Tiruan

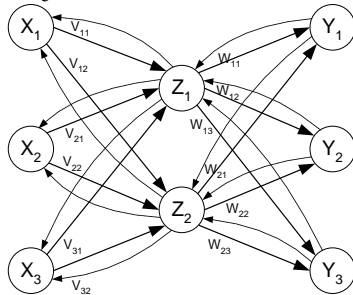
Jaringan Syaraf Tiruan (*Artificial Neural Network*) merupakan suatu model yang meniru cara kerja jaringan pada syaraf manusia, dimana otak manusia memiliki struktur yang sangat kompleks dan kemampuan yang luar biasa. Jaringan syaraf dapat digolongkan menjadi berbagai jenis berdasarkan pada arsitekturnya, yaitu pola hubungan antar *neuron-neuron*, dan

algoritma pelatihnannya(*training*). Arsitektur yang digunakan pada jaringan syaraf ini adalah *Multilayer Net*, dimana jaringan dengan banyak lapisan memiliki satu atau lebih lapisan yang terletak diantara lapisan masukan dan lapisan keluaran (memiliki satu atau lebih lapisan tersembunyi). Fungsi aktivasi yang digunakan dalam *backpropagation*, yaitu *sigmoid biner*. Fungsi *sigmoid biner* memiliki nilai range 0 dan 1.[2] Fungsi *sigmoid biner* dirumuskan sebagai berikut :

$$f(x) = \frac{1}{1 + e^{-x}} \quad (1)$$

Backpropagation

Algoritma *backpropagation* menggunakan *error* keluaran untuk mengubah nilai bobot-bobotnya dalam arah mundur (*backward*). Untuk mendapatkan *error* ini, tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu.



Gambar 1. Arsitektur Jaringan Backpropagation

Fungsi Kinerja

Fungsi kinerja yang sering digunakan untuk *backpropagation* adalah *Mean Square Error* (MSE). Fungsi ini akan mengambil rata-rata kuadrat *error* yang terjadi antara keluaran jaringan dan target. Fungsi lain yang dapat digunakan adalah *Sum Square Error* (SSE). Fungsi ini akan mengambil dan menjumlahkan seluruh kuadrat *error* yang terjadi antara keluaran jaringan dan target.

$$MSE = \frac{\sum_{i=1}^n e_i^2}{n} \quad (2)$$

$$SSE = \sum_{i=1}^K e(i)^2 \quad (3)$$

Levenberg-Marquadt

Algoritma *Levenberg-Marquadt* 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*.

Histogram

Histogram citra merupakan sebuah grafik yang menunjukkan distribusi intensitas warna pada sebuah citra berindeks atau citra intensitas. Fungsi *image histogram* diwakili dengan perintah *imhist* yang akan menciptakan sebuah grafik yang terbentuk dari garis-garis yang masing-masing garis ini mewakili *range* dari intensitas warna, kemudian akan menjumlahkan jumlah piksel didalam tiap *range*. [4]

Sumbu X (absis) yang menunjukkan tingkat derajat keabuan. Sumbu Y (ordinat) yang menunjukkan frekuensi kemunculan derajat keabuan. Kegunaan dari histogram adalah :

- Penentuan parameter digitasi
- Pemilihan batas ambang
- Pengenalan / pencocokan citra

Secara matematis histogram disimbolkan dengan $f(x,y)$, dimana:

(x,y) : koordinat pada bidang dwi warna.

$f(x,y)$: intensitas cahaya pada titik (x,y) .

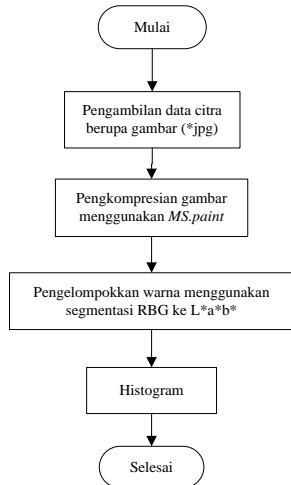
Berikut karakteristik histogram terhadap citra atau gambar yang memiliki kriteria tertentu, seperti :

- Gambar gelap : histogram cenderung ke sebelah kiri.
- Gambar terang: histogram cenderung ke sebelah kanan.
- Gambar low contrast : histogram mengumpul di suatu tempat.
- Gambar high contrast : histogram merata di semua tempat.

3. PERANCANGAN SISTEM

Rangkaian proses yang akan dilakukan oleh *software* ini adalah pengambilan data citra gambar (*.jpg), pengompresian gambar menggunakan *MS.paint*, segmentasi, dan histogram.

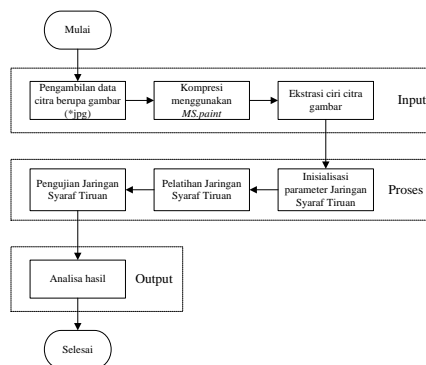
PENGELOMPOKKAN CITRA WARNA MENGGUNAKAN JARINGAN SYARAF TIRUAN DENGAN SOFTWARE MATLAB



Gambar 2. Diagram Pengelompokkan Warna

Proses Jaringan Syaraf Tiruan sendiri membutuhkan proses pelatihan (*training*) agar keluaran yang dihasilkan benar dan akurat.

Secara garis besar dapat dikatakan alur aplikasi ini dimulai dari masukan sebuah gambar. Dimana gambar tersebut akan melalui serangkaian manipulasi citra dalam proses yang akan mengubah gambar tersebut menjadi serangkaian nilai-nilai yang berguna untuk proses utama, yaitu: proses pelatihan (*training*) dan proses pengujian (*recognition process*).



Gambar 3 . Diagram Perancangan JST

Segmentasi Konversi Ruang Warna RGB ke $L^*a^*b^*$

Segmentasi adalah proses memisahkan area pengamatan (*region*) pada tiap karakter yang dideteksi. Segmentasi citra akan membagi-bagi suatu citra menjadi daerah-daerah atau objek-objek yang dimilikinya. Segmentasi

warna merupakan proses segmentasi dengan pendekatan daerah yang bekerja dengan menganalisa nilai warna dari tiap piksel pada citra dan membagi citra tersebut dengan fitur yang diinginkan. Salah satu ruang warna adalah ruang warna $L^*a^*b^*$ atau yang dikenal dengan CIELAB. Ruang warna ini mampu menggambarkan semua warna yang dapat dilihat dengan mata manusia dan sering digunakan sebagai referensi ruang warna.

Inisialisasi Parameter

Parameter yang diset untuk pembelajaran adalah *trainlm*. Trainlm adalah metode yang paling tepat digunakan untuk merancang Jaringan Syaraf Tiruan. Terdapat pula beberapa nilai parameter yang harus diset untuk pelatihan. [3] Parameter tersebut, yaitu:

a. Maksimum *epoch*

Instruksi : `net.trainParam.epochs = MaxEpoch`
Nilai standar untuk maksimum *epoch* adalah 25.

b. Kinerja tujuan

Instruksi : `net.trainParam.goal = TargetError`
Nilai standar untuk kinerja tujuan adalah 0.

c. *Learning rate*

Instruksi : `net.trainParam.lr = Learning rate`
Nilai standar untuk *learning rate* adalah 0.01.

d. Momentum

Instruksi : `net.trainParam.mc = momentum`
Nilai standar untuk momentum adalah 0 dan 1.

e. Jumlah *epoch* yang akan ditunjukkan kemajuannya.

Instruksi : `net.trainParam.show = Epochshow`
Nilai standar untuk jumlah *epoch* yang akan ditunjukkan adalah 10.

Metode Pelatihan Jaringan

Sebelum dilakukan pelatihan data maukan maupun target terlebih dahulu dilakukan normalisasi dengan rata-rata (*mean*) dan simpangan baku (*standar deviasi*). Untuk melakukan proses tersebut digunakan fungsi *prestd* sehingga data normal dengan *mean* = 0 dan standar deviasi = 1. Fungsi *prestd* menggunakan sintaks :

`[pn,meanp,stdp,tn,meant,stdt]=prestd(p,t)`

Kemudian jaringan saraf tiruan dibangun dengan metode *levenberg marquadt* (*trainlm*) dengan sintaks:

`net=newff(PR,[S1
S2.....SN},{TF1,TF2...TFN},BTF,BLF,PF)`

Setelah jaringan saraf tiruan dibangun dengan metode *levenberg marquadt* kemudian akan dilakukan identifikasi bobot dan bias dengan fungsi *net* yang memiliki persamaan sebagai berikut :

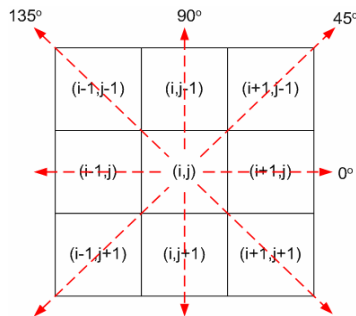
$$[W1, b1, W2, b2] = \text{net}(IW, b, LW)$$

Tahap selanjutnya adalah pelatihan jaringan. Pelatihan jaringan dilakukan dengan menggunakan fungsi *net* dengan persamaan sebagai berikut :

$$\text{net} = \text{train}(\text{net}, \text{pn}, \text{tn})$$

Ekstraksi Ciri

Analisis tekstur lazim dimanfaatkan sebagai proses untuk melakukan klasifikasi dan interpretasi citra. Suatu proses klasifikasi citra berbasis analisis pada umumnya membutuhkan ekstraksi ciri. Ekstraksi ciri statistik orde kedua dilakukan dengan matriks kookurensi, yaitu suatu matriks antara yang merepresentasikan hubungan ketetanggaan antar piksel dalam citra pada berbagai arah orientasi dan jarak spasial.

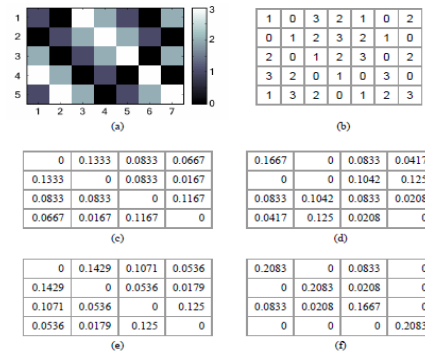


Gambar 4. Ilustrasi Ekstraksi Ciri Statistik

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.

Kookurensi berarti kejadian bersama, yaitu jumlah kejadian satu level nilai piksel bertetangga dengan satu level nilai piksel lain dalam jarak (d) dan orientasi sudut (θ) tertentu. Jarak dinyatakan dalam derajat. Orientasi dibentuk dalam empat arah sudut dengan interval sudut, yaitu 0° , 45° , 90° ,

135° . Sedangkan jarak antara piksel biasanya ditetapkan sebesar 1 piksel.



Gambar 5. Ilustrasi Pembuatan Matriks Kookurensi

(a)Citra Masukan, (b)Nilai Intensitas Citra Masukan, (c)Hasil Matriks Kookurensi 0° , (d) Hasil Matriks Kookurensi 45° , (e) Hasil Matriks Kookurensi 90° , (f) Hasil Matriks Kookurensi 135°

Matriks kookurensi merupakan matriks bujursangkar dengan jumlah elemen sebanyak kuadrat jumlah level intensitas piksel pada citra. Setiap titik (p, q) pada matriks kookurensi θ berisi peluang kejadian piksel bernilai p bertetangga dengan piksel bernilai q pada jarak d serta orientasi θ dengan $(180-\theta)$. Dalam modul ini dicontohkan perhitungan enam ciri statistik orde dua, yaitu *Angular Second Moment*, *Contrast*, *Correlation*, *Variance*, *Inverse Difference Moment*, dan *Entropy*.

- *Angular Second Moment*

Menunjukkan ukuran sifat hegemony citra, dimana $p(i, j)$ menyatakan nilai pada baris i dan kolom j pada matriks kookurensi

$$\text{ASM} = \sum_i \sum_j \{p(i, j)\}^2 \quad (4)$$

- *Contrast*

Menunjukkan ukuran penyebaran (momen inersia) elemen-elemen Matriks citra. Jika letaknya jauh dari diagonal utama, nilai kontrasnya besar. Secara visual, nilai kontras adalah ukuran variasi antara derajat keabuan suatu citra.

$$\text{CON} = \sum_k k^2 \left[\sum_i \sum_j p(i, j) \right] \quad (5)$$

- *Correlation*

Menunjukkan ukuran ketergantungan derajat keabuan citra sehingga dapat memberikan petunjuk adanya struktur linear dalam citra

$$\text{COR} = \frac{\sum_i \sum_j (i, j) \cdot p(i, j) - \mu_x \mu_y}{\sigma_x \sigma_y} \quad (6)$$

PENGELOMPOKAN CITRA WARNA MENGGUNAKAN JARINGAN SYARAF TIRUAN DENGAN SOFTWARE MATLAB

- Variance

Menunjukkan variasi elemen -elemen matriks kookurensi. Citra dengan transisi derajat keabuan kecil akan memiliki variansi yang kecil pula.

$$VAR = \sum_i \sum_j (i - \mu_x)(j - \mu_y) \cdot p(i, j) \quad (7)$$

- Inverse Difference Moment

Menunjukkan kehomogenan citra yang berderajat keabuan sejenis. Citra Homogen akan memiliki harga *IDM* yang besar

$$IDM = \sum_i \sum_j \frac{1}{1+(i-j)^2} \cdot p(i, j) \quad (8)$$

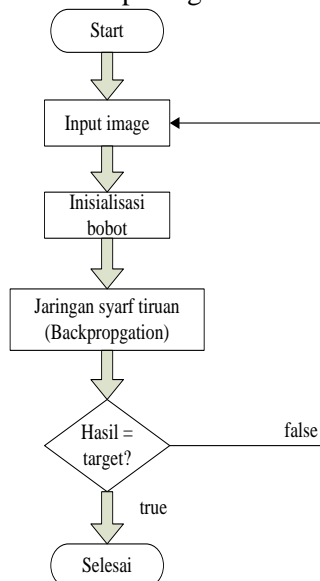
- Entropy

Menunjukkan ukuran ketidakteraturan bentuk. Harga *ENT* besar untuk citra dengan transisi derajat keabuan merata dan bernilai kecil jika struktur citra tidak teratur (bervariasi)

$$ENT_2 = \sum_i \sum_j p(i, j) \cdot \log p(i, j) \quad (9)$$

Perancangan Flowchart Program JST

Flowchart merupakan alat bantu pemrograman yang biasanya digunakan. Diagram alir (*flowchart*) membantu programmer dalam mengorganisasikan pemikiran dalam pemrograman.

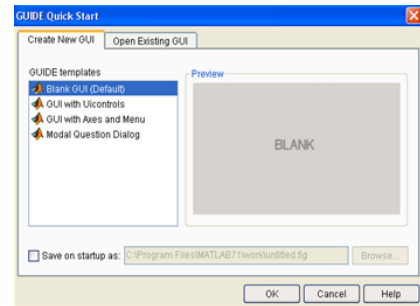


Gambar 6. Flowchart Program JST

Perancangan Antarmuka Menggunakan GUI

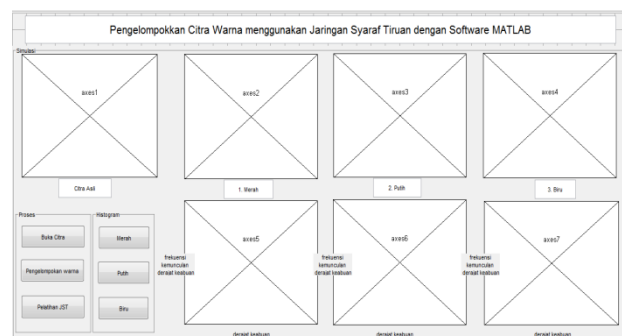
Pembuatan GUI ini merupakan tahap penting, karena GUI yang nanti akan mempermudah pengguna mengoperasikan program simulasi tersebut. Untuk membuat GUI baru setelah Matlab dibuka maka pilih GIUDE (*GUI builder*). Yang kemudian pada

layar akan muncul GUIDE *Quick Start*, ada dua pilihan yaitu *create new GUI* dan *Open Existing GUI*. Seperti yang dapat dilihat pada gambar dibawah ini.



Gambar 7. GUIDE Quick Start

Perancangan antarmuka merupakan rancangan pengelolaan informasi pada sistem. Perancangan ini dibuat sebagai gambaran dari suatu program atau aplikasi yang akan dibangun. Rancangan antarnuka yang dibangun sebagai berikut:



Gambar 8. Interface GUI

Pengujian

Untuk penguji sistem digunakan parameter akurasi dan *error*.

- Akurasi adalah ukuran ketepatan sistem dalam mengenali masukan yang diberikan sehingga menghasilkan keluaran yang benar. Secara sistematis dapat dituliskan sebagai berikut :

$$Akurasi = \frac{\text{jumlah data benar}}{\text{jumlah data keseluruhan}} \times 100 \% \quad (14)$$

- Error* adalah tingkat kesalahan sistem dalam mengenali masukan yang diberikan terhadap jumlah data secara keseluruhan. Secara sistematis dapat dituliskan sebagai berikut :

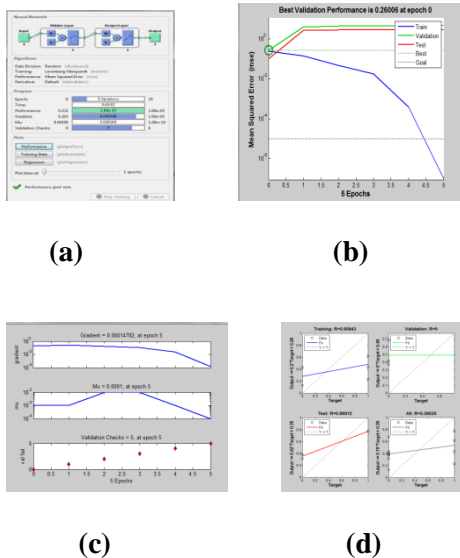
$$Error = \frac{\text{jumlah data benar}}{\text{jumlah data keseluruhan}} \times 100\% \quad (15)$$

4. Hasil dan pembahasan

Jaringan yang telah dibangun memiliki parameter seperti pada gambar dibawah ini :

- Jumlah *epoch* = 25; jumlah *hidden layer* = 5; *performance* = 0,260.
- Garis grafik *performace* jaringan terjadi pada *epoch* ke-5.
- Nilai gradien = 0,000147582, Mu = 0,0001, nilai cek validasi = 5 pada *epoch* ke-5.
- Nilai regresi untuk *training* = 0,50843; *validation* = 0; *test* = 0,99012.

Tampilan hasil pelatihan Jaringan Syaraf Tiruan sebanyak empat kali *cropping* adalah sebagai berikut :



Gambar 10. (a) Jaringan Neural Network, (b) Grafik Mean Square Error (MSE) sampai *epoch* ke-25, (c) Grafik Gradient, Mu dan Validasi, (d) Grafik Regresi Training, Validasi dan Test.

Nilai standar maksimum *epoch* adalah 25. *Epoch* adalah perulangan atau iterasi dari proses yang dilakukan untuk mencapai target yang telah ditentukan. maksimum *epoch* adalah jumlah *epoch* maksimum yang boleh dilakukan selama proses pelatihan. Iterasi akan berhenti apabila nilai *epoch* melebihi maksimum *epoch*.

Performance adalah fungsi kinerja dimana target kinerja tujuan pada iterasi akan dihentikan apabila nilai fungsi kinerja kurang dari atau sama dengan kinerja tujuan.

Mean Squared Error (MSE) adalah nilai yang diharapkan dari kuadrat *error*. *Error* yang ada menunjukkan seberapa besar perbedaan hasil estimasi dengan nilai yang akan diestimasi.

Gradient adalah akar dari jumlah kuadrat gradien (bobot input, bobot lapisan, bobot bias).

Mu adalah momentum dimana perubahan bobot yang didasarkan atas arah gradien pola terakhir dan pola sebelumnya.

Validasi checks adalah nilai kegagalan validasi.

Hasil keakurasian warna yang dikenali setelah dikompresi adalah sebagai berikut :

$$\text{Warna merah} = \frac{16}{22} \times 100 \% = 73 \%$$

$$\text{Warna putih} = \frac{14}{22} \times 100 \% = 63 \%$$

$$\text{Warna biru} = \frac{18}{22} \times 100 \% = 82 \%$$

5. Kesimpulan

- Apabila gambar tidak dikenali pada saat pengelompokkan warna kemungkinan ukuran resolusi gambar terlalu kecil atau besar tergantung pada saat pengompresian.
- Hasil akurasi warna yang dikenali terhadap 22 sampel batuan sayatan tipis batuan sebelum dikompresi dan setelah dikompresi 80% berturut-turut untuk warna merah $\Delta = 33\%$, warna putih $\Delta = 41\%$, dan warna biru $\Delta = 23\%$.

6. Daftar pustaka

- Gonzalez RC, Wood RE. "Digital Image Processing", Second Edition. Prentice Hall, Inc., New Jersey, 2002
- J. J. Siang, "Jaringan Syaraf Tiruan dan Pemogramannya Menggunakan MATLAB", Penerbit Andi. Yogyakarta, 2004
- Kusumadewi, Sri, "Membangun Jaringan Syaraf Tiruan Menggunakan Matlab dan Excel Link". Yogyakarta : Graha ilmu, 2004.
- Wijaya, Marvin Ch dan Prijono. A. "Pengolahan Citra Digital menggunakan MATLAB Image Processing Toolbox", Penerbit Informatika. Bandung, 2007