

DETEKSI PENYAKIT KULIT MENGGUNAKAN ANALISIS FITUR WARNA DAN TEKSTUR DENGAN METODE COLOR MOMENT, GRAY LEVEL COOCCURENCE MATRIX, DAN JARINGAN SARAF TIRUAN BACKPROPAGATON

Agus Wahyu Sanjaya¹, Achmad Rizal², Inung Wijayanto³

¹Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

Abstrak

Kulit sebagai organ terluar tubuh manusia sangat rentan terkena penyakit. Penyakit ini bisa disebabkan oleh virus, jamur, maupun bakteri. Penyakit kulit harus diidentifikasi berdasarkan jenisnya agar bisa mendapatkan penanganan yang tepat. Agar bisa dilakukan identifikasi, maka dibutuhkan beberapa informasi. Melalui sebuah citra dapat dipelajari informasi mengenai penyakit kulit tersebut seperti warna, tekstur, dan bentuk.

Kulit sebagai organ terluar tubuh manusia sangat rentan terkena penyakit. Penyakit ini bisa disebabkan oleh virus, jamur, maupun bakteri. Penyakit kulit harus diidentifikasi berdasarkan jenisnya agar bisa mendapatkan penanganan yang tepat. Agar bisa dilakukan identifikasi, maka dibutuhkan beberapa informasi. Melalui sebuah citra dapat dipelajari informasi mengenai penyakit kulit tersebut seperti warna, tekstur, dan bentuk.

Sistem ini mampu mengklasifikasikan penyakit kulit dengan akurasi mencapai 88,75 %. Waktu proses sistem ini berada pada rentang waktu 0,33 detik sampai 0,97 detik dengan waktu proses rata-rata sebesar 0,61 detik.

Kata Kunci : penyakit kulit, color moment, Gray Level Co-occurrence Matrix (GLCM), Jaringan Saraf Tiruan

Abstract

Skin as the outer part of human body is very easy to get infected from diseases. These diseases can be caused by virus, fungus, and bacteria. Skin diseases should be identified to get a correct treatment. Some information is needed for identification process. These informaton such as color, texture, and shape can be learnt from an image.

In this final project, a software system which capable to detect skin disease on digital images has been developed. Diagnose on the image will be done by using some method called Color Moment for color feature extraction, Gray Level Co-occurrence Matrix (GLCM) for texture feature extraction, and Artificial Neural Network with Backpropagation Architecture for classification.

The system is able to classify skin diseases with 88,75% accuracy and the system has a range from 0,33 seconds to 0,97 seconds of processing time, with 0,61 seconds average processing time.

Keywords : The system is able to classify skin diseases with 88,75% accuracy and the system has a range from 0,33 seconds to 0,97 seconds of processing time, with 0,61 seconds average processing time.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kulit merupakan organ tubuh terluar dari manusia. Kulit memegang peranan vital dalam melindungi organ dalam tubuh manusia dari ancaman yang berasal dari lingkungan di luar tubuh manusia^[5]. Oleh karena fungsinya tersebut kulit menjadi sensitif dan rentan terhadap serangan bakteri, virus, dan jamur yang bisa menimbulkan penyakit pada kulit.

Dalam penanganan penyakit kulit, pertama kali harus dilakukan deteksi untuk mengidentifikasi jenis penyakit kulit tersebut sehingga bisa dilakukan penanganan yang tepat. Jenis penyakit kulit ada bermacam-macam. Beberapa penyakit kulit mempunyai tekstur dan warna yang berbeda. Dari warna dan tekstur tersebut bisa dilakukan klasifikasi terhadap jenis – jenis penyakit kulit. Dengan berkembangnya teknologi *image processing* proses pendeteksian penyakit kulit bisa dilakukan melalui pengolahan citra digital. Dengan pengolahan citra digital sebuah citra berpenyakit kulit diekstrak fitur warna dan fitur teksturnya. Fitur warna dan fitur tekstur ini kemudian dianalisis dan diklasifikasikan sehingga bisa diketahui jenis penyakit kulit tersebut.

Pada penelitian sebelumnya fitur warna diekstrak dengan metode *Color Histogram* sedangkan fitur tekstur diekstrak menggunakan *Edge Histogram Descriptor* dengan *classifier* yang digunakan adalah *K-Nearest Neighbour*. Dimana hasil penelitian sebelumnya menghasilkan akurasi sebesar 75,2%^[14]. Untuk penelitian ini metode yang digunakan adalah *Color Moment* untuk analisis fitur warna dan *Gray Level Cooccurrence Matrix (GLCM)* untuk analisis fitur tekstur. *Color Moment* dipilih karena sangat efektif untuk pengolahan citra berbasis warna termasuk klasifikasi citra berdasar warna^[12]. Sedangkan *GLCM* sendiri merupakan metode yang mampu menyediakan informasi yang penting mengenai tekstur citra^[11]. Untuk *classifier* yang digunakan sistem adalah Jaringan Saraf Tiruan (JST) *Backpropagation* karena mampu digunakan untuk menyelesaikan problem non linier termasuk pengenalan pola^[2].

1.2 Tujuan

Hasil yang ingin dicapai melalui penelitian yang dilakukan pada tugas akhir ini antara lain:

1. Merancang aplikasi yang mampu mendeteksi jenis penyakit kulit berbasis pengolahan citra digital menggunakan analisis fitur warna dan tekstur dengan metode *Color Moment*, GLCM dan *JST Backpropagation*.
2. Menganalisis pengaruh masing-masing metode ekstraksi ciri terhadap akurasi sistem.
3. Menganalisis pengaruh parameter JST terhadap akurasi yang dihasilkan sistem.
4. Mengukur waktu proses pada sistem pendeteksian

1.3 Rumusan Masalah

Permasalahan yang dibahas dalam tugas akhir ini antara lain :

1. Bagaimana cara merancang aplikasi yang mampu mendeteksi jenis penyakit kulit berbasis pengolahan citra digital menggunakan *Color Moment* dan GLCM dengan metode Pengklasifikasian Jaringan Saraf Tiruan *Backpropagation*.
2. Bagaimana pengaruh masing-masing metode ekstraksi ciri terhadap akurasi sistem.
3. Bagaimana pengaruh parameter JST terhadap akurasi yang dihasilkan sistem.
4. Berapakah waktu proses pada sistem pendeteksian

1.4 Batasan Masalah

Beberapa hal yang dijadikan batasan masalah pada penelitian Tugas Akhir ini adalah:

1. *Input* (masukan) sistem berupa citra penyakit kulit yang telah dipotong (*di-crop*) dan tanpa tambahan *noise*.
2. Format penyimpanan *file* gambar adalah *Bitmap* (*.bmp) atau JPEG (*.jpg).

3. Sistem yang akan dibangun menggunakan *software* Matlab 7.8 (R2009a) dengan menggunakan *Color Moment* untuk ekstraksi ciri warna, GLCM untuk ekstraksi ciri tekstur dan *JST Backpropagation* untuk klasifikasi.
4. Keluaran sistem adalah jenis penyakit kulit pada citra yang dijadikan masukan sistem.
5. Jenis penyakit kulit yang dideteksi adalah *acne*, *dermatitis*, *herpes*, *scabies*.
6. Tidak dibahas mengenai cara pengambilan gambar.

1.5 Metode Penelitian

Metodologi penelitian dalam penyusunan tugas akhir ini meliputi:

1. Studi Literatur

Tahapan ini bertujuan untuk mengumpulkan bahan yang berupa buku referensi, artikel-artikel, jurnal-jurnal yang mendukung, dan dasar teori yang kuat tentang metode yang akan digunakan dalam penelitian ini.

2. Analisis Desain

Tahap ini meliputi analisis kebutuhan untuk merancang perangkat lunak pendeteksian penyakit kulit dengan metode *Color Moment*, GLCM, dan *JST Backpropagation*. Desain perancangan akan dibentuk dalam diagram alir.

3. Implementasi Sistem

Tahap ini meliputi pembangunan perangkat lunak yang telah dirancang sebelumnya. Pada tahap ini diimplementasikan perancangan perangkat lunak menggunakan *software* pemrograman Matlab 7.8 (R2009a).

4. Pengujian dan Analisis

Pada tahap ini dilakukan pengujian terhadap perangkat lunak yang telah dibangun dan sekaligus melakukan analisis peformansi dari perangkat lunak.

5. Penyusunan laporan

Pada tahap ini dilakukan penyusunan laporan hasil penelitian yang telah dilakukan dan membuat kesimpulan dari hasil penelitian tersebut.

1.6 Sistematika Penulisan

Secara umum keseluruhan Tugas Akhir ini dibagi menjadi lima bab bahasan. Penjelasannya adalah sebagai berikut:

BAB I Pendahuluan

Pada bab ini dibahas tentang latar belakang penelitian, tujuan penelitian, rumusan masalah, batasan masalah, metode penelitian, dan sistematika penulisan tugas besar.

BAB II Dasar Teori

Pada bab ini memuat berbagai dasar teori yang mendukung dan mendasari penulisan tugas akhir ini.

BAB III Perancangan Sistem

Pada bab ini dijelaskan tentang perencanaan sistem berdasarkan mekanisme dan batasan yang telah ditentukan.

BAB IV Pengujian Sistem dan Analisis

Pada bab ini dilakukan analisis hasil implementasi sistem sesuai skenario yang telah dirancang dan ditetapkan.

BAB V Penutup

Berisi kesimpulan dari seluruh rangkaian penelitian dan saran untuk keperluan lebih lanjut yang mungkin dilakukan.

Telkom
University

BAB III

PERANCANGAN SISTEM

3.1 Analisis Kebutuhan

Pada tahap ini dilaksanakan analisis tentang kebutuhan dalam perancangan sistem Deteksi Penyakit Kulit Menggunakan Analisis Fitur Warna dan Tekstur dengan Metode *Color Moment*, GLCM, dan *JST Backpropagation* yang untuk selanjutnya disebut sistem pendeteksian. Hal-hal yang dianalisis meliputi kebutuhan *hardware* atau perangkat keras, *software* atau perangkat lunak dan kebutuhan fungsionalitas sistem sehingga sistem dapat bekerja dan berfungsi seperti yang diharapkan. Atas dasar itulah perlu dilakukan analisis terhadap masing-masing kebutuhan tersebut agar perancangan dapat dilakukan dengan maksimal.

3.1.1 Analisis Kebutuhan Perangkat Keras dan Perangkat Lunak

Adapun *software* dan *hardware* yang diperlukan untuk membangun sistem pendeteksian meliputi:

a. Spesifikasi perangkat keras minimum

- *Intel x86* atau *AMD Processor*
- RAM 1024 MB
- *Hardisk* 4 GB
- VGA 128 MB

b. Spesifikasi perangkat lunak minimum

- Sistem Operasi *Microsoft Windows XP*
- MATLAB 7.8 (R2009a)

3.1.2 Analisis Kebutuhan Fungsionalitas

Pada sistem pendeteksian, terdapat tiga bagian penting yaitu proses ekstraksi ciri, *training* dan klasifikasi. Dari ketiga proses tersebut dibutuhkan fungsionalitas yang dapat membangun ketiga proses tersebut. Ketiga proses tersebut dilaksanakan secara sekuensial dimana hasil dari ekstraksi ciri digunakan pada proses *training* dan hasil dari proses *training* digunakan pada proses klasifikasi. Adapun fungsionalitas yang dibutuhkan dalam melakukan proses ekstraksi ciri, *training*, dan klasifikasi tersebut adalah sebagai berikut:

a. Proses Ekstraksi Ciri

- Yang pertama adalah menerima *input user* dimana *input* berupa gambar penyakit kulit dengan format *.bmp atau *.jpg
- Kemudian *color moment* digunakan untuk menghitung tiga momen pertama dari masing-masing *layer* (R,G,B) dari citra *input*
- Untuk fitur tekstur, citra *input* diubah menjadi *grayscale* terlebih dahulu. Citra *grayscale* kemudian diekstrak menggunakan GLCM

b. Proses *Training*

- Matriks yang berisi vektor-vektor ciri hasil ekstraksi ciri di-*training* agar mendapat bobot jaringan untuk digunakan pada proses klasifikasi
- Parameter *training* meliputi jumlah *hidden layer*, jumlah *neuron* per *hidden layer*, fungsi *training*, dan fungsi aktivasi di-*input* oleh *user*

c. Proses Klasifikasi

- Pada proses klasifikasi bobot dan arsitektur jaringan hasil *training* di-*load*
- Vektor ciri data uji disimulasikan dengan jaringan hasil *training* yang sudah di-*load* untuk memperoleh hasil klasifikasi

3.1.3 Analisis *Input* dan *Output* Sistem

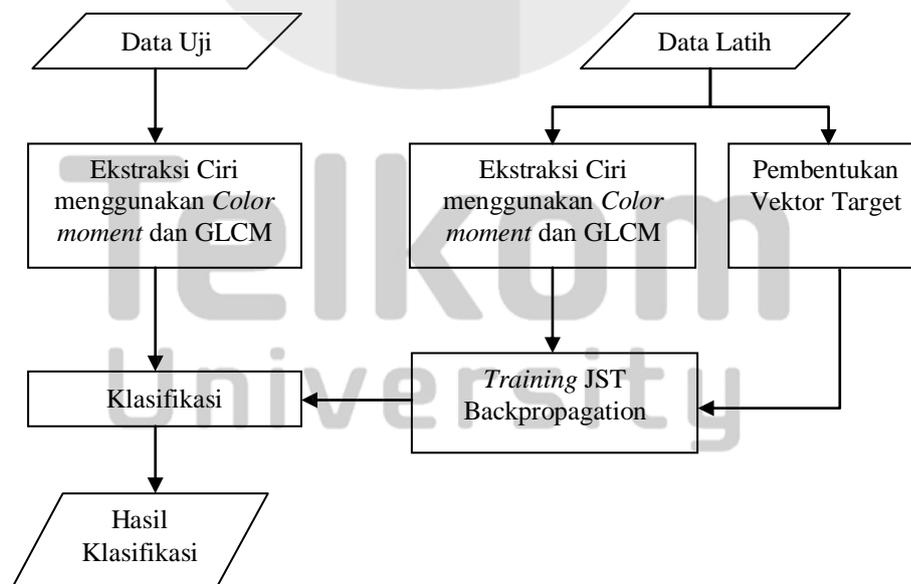
Sistem pendeteksian terdiri dari beberapa proses yang saling berkaitan dimana *output* dari salah satu proses menjadi *input* dari proses yang lain. Proses besar yang ada pada sistem pendeteksian adalah ekstraksi ciri, *training*, dan klasifikasi. Dalam proses besar tersebut terdapat beberapa proses kecil di

dalamnya. Untuk lebih mudah dalam pembangunan suatu proses maka sebelumnya harus dilakukan analisis terhadap *input* dan *output* dari masing-masing proses tersebut. Berikut adalah analisis *input output* dari masing-masing proses:

1. Ekstraksi Ciri : *input* dari proses ini adalah citra penyakit kulit yang sudah dipotong dengan ukuran 200x200 piksel. Semua citra diambil pada kondisi pencahayaan yang terang. *Output* dari proses ini adalah matriks Nx1 dimana N adalah jumlah vektor ciri.
2. *Training* : proses *training* menerima *input* berupa matriks yang berisi vektor-vektor ciri dari data latih dan targetnya sedangkan *output* proses *training* adalah jaringan yang sudah memiliki pengetahuan.
3. Klasifikasi : *input* dari proses pengujian adalah vektor ciri data uji dan *output* adalah kelas dari vektor ciri tersebut. Dari kelas tersebut didapatkan jenis penyakit kulit. Sistem ini hanya akan mendeteksi empat jenis penyakit kulit yaitu : *scabies*, *acne*, *dermatitis*, dan *herpes*

3.2 Pemodelan Sistem

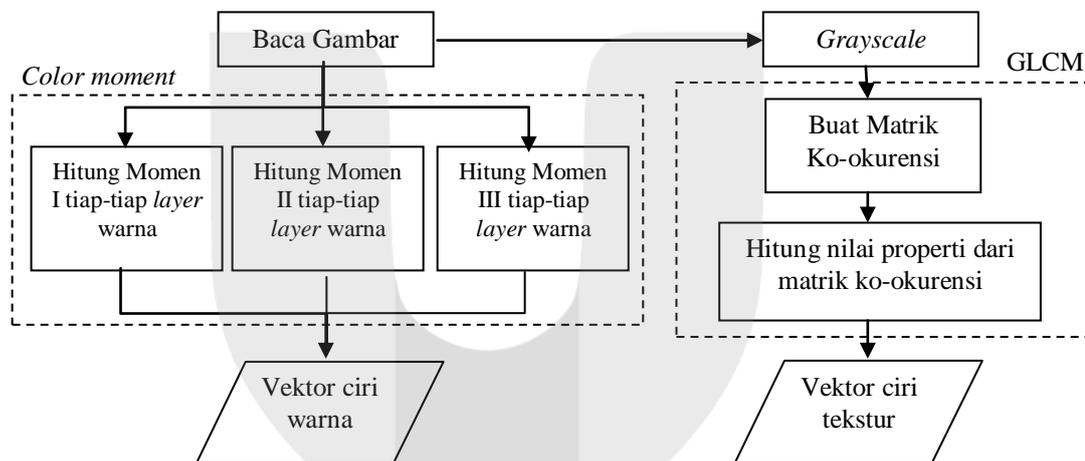
Untuk pengembangan sistem perangkat lunak ini akan digunakan *framework* atau gambaran umum aliran data pada sistem sebagai berikut.



Gambar 3.1 Diagram Alir Sistem

Dari blok diagram di atas dapat dilihat bahwa sistem mempunyai empat proses utama yaitu proses ekstraksi ciri menggunakan *Color Moment* dan GLCM, proses pembentukan vektor target, proses *training JST Backpropagation*, dan yang terakhir adalah proses klasifikasi. Untuk *input* sistem sendiri terdapat dua bagian yaitu data uji dan data latih. Data latih digunakan saat proses *training JST Backpropagation* dan data uji digunakan saat proses klasifikasi. Data latih yang digunakan adalah citra yang sudah diketahui label kelasnya. Sedangkan untuk data uji merupakan data yang kelasnya tidak diketahui oleh sistem.

3.2.1 Proses Ekstraksi Ciri Menggunakan *Color Moment* dan GLCM



Gambar 3.2 Diagram Alir Proses Ekstraksi Ciri menggunakan *Color Moment* dan GLCM

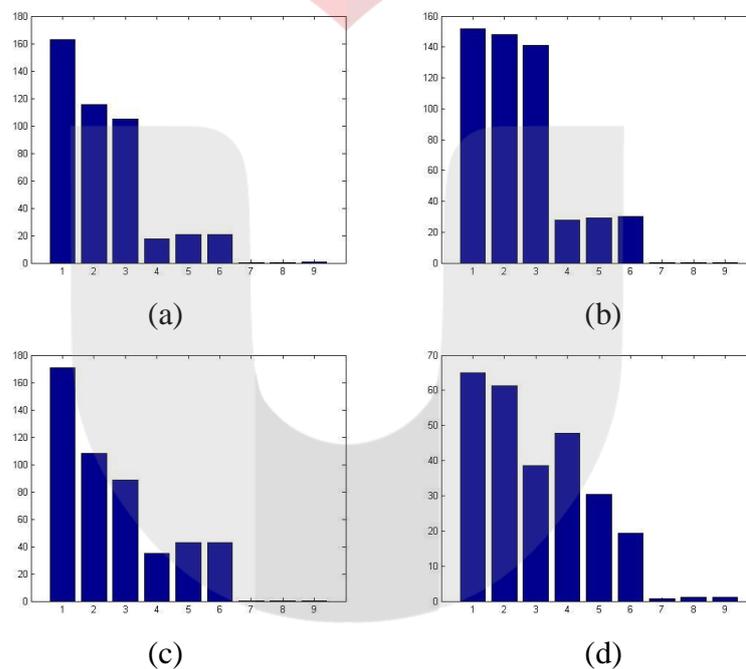
Pada proses ekstraksi ciri data *input* akan dibaca dan kemudian diolah dalam dua proses yang berlainan. Proses pertama adalah ekstraksi ciri warna dengan metode *Color Moment* dan yang kedua adalah ekstraksi fitur tekstur menggunakan metode GLCM. Berikut akan dijelaskan bagaimana proses ekstraksi ciri dilakukan.

a) Proses Baca Gambar

Pada proses ini citra *input* akan dibaca oleh sistem. Citra *input* dibaca sebagai matriks berukuran $m \times n \times 3$, dimana $m \times n$ merupakan ukuran dari citra dan angka 3 merupakan jumlah *layer* warna pada citra yaitu *layer* R (*red*), G (*green*), dan B (*blue*).

b) Proses *Color Moment*

Setelah gambar dibaca oleh sistem, sistem akan melakukan proses ekstraksi ciri warna dengan metode *Color Moment*. Dalam proses ini masing-masing momen (*mean*, standar deviasi, *skewness*) akan dihitung di semua *layer* warna. Jadi proses ini nantinya akan menghasilkan sembilan buah variabel yang terdiri dari tiga variabel *mean*, tiga variabel standar deviasi, dan tiga variabel yang menyatakan *skewness* dari citra *input*. Kesembilan variabel ini kemudian dibentuk menjadi vektor yang berukuran 9x1, dimana vektor tersebut merupakan *output* dari proses ini yang merepresentasikan fitur warna citra *input*.



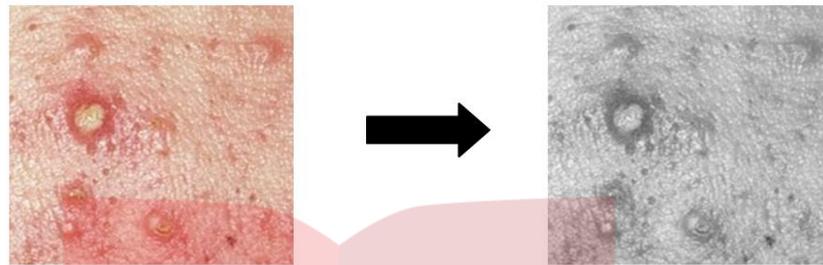
Gambar 3.3 Plot Ciri Warna

(a) *Acne*, (b) *Scabies*, (c) *Dermatitis*, (d) *Herpes*

c) Proses *Grayscale*

Proses *grayscale* adalah proses dimana citra *input* yang terdiri dari tiga *layer* warna RGB dirubah hanya menjadi satu *layer*, dimana *layer* tersebut menyatakan tingkat keabuan disetiap titik pada citra. Proses ini diperlukan karena pada proses ekstraksi ciri GLCM memerlukan citra *grayscale* sebagai *input*-nya. Proses perubahan ini dilakukan dengan melakukan penjumlahan komponen RGB yang telah diberikan bobot masing-masing. Persamaan matematisnya adalah :

$$gray_value = 0,2989R + 0,5870G + 0,1140B \tag{3.1}$$



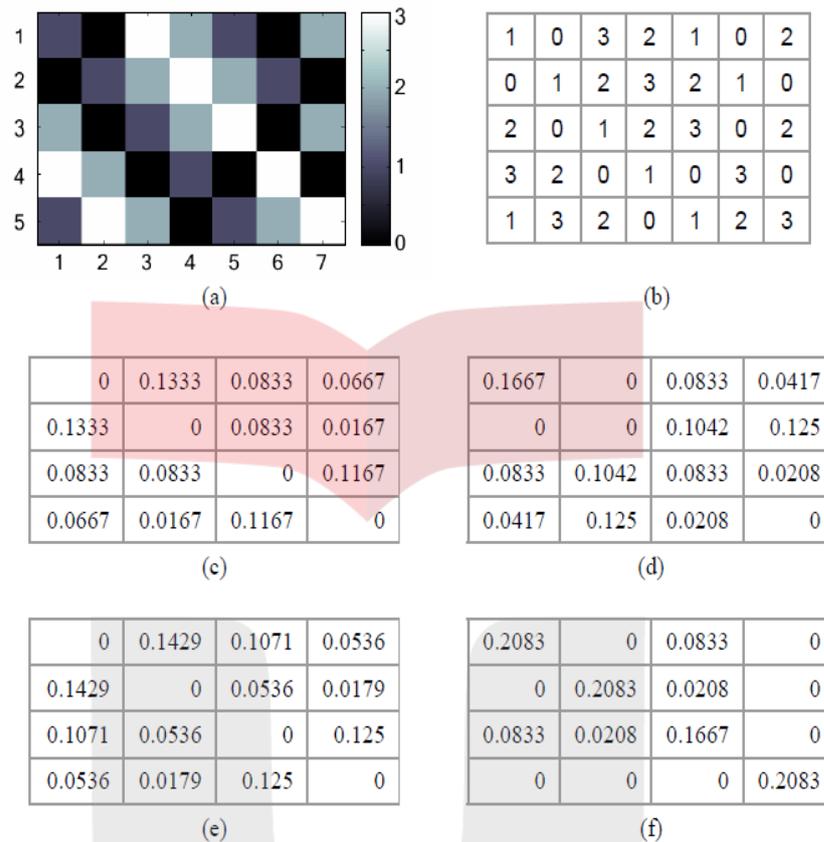
Gambar 3.4 Perubahan dari Citra RGB ke Citra *Grayscale*

d) Proses GLCM

Setelah citra diubah menjadi *grayscale*, citra akan diekstrak fitur teksturnya melalui metode GLCM. Metode ini dapat dibagi lagi menjadi dua tahap yang pertama adalah pembuatan matriks kookurensi dan yang kedua adalah penghitungan nilai properti dari matriks tersebut.

- Pembuatan matriks kookurensi

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 piksel dan orientasi dinyatakan dalam derajat. Orientasi dibentuk dalam empat arah sudut dengan interval sudut 45° , yaitu 0° , 45° , 90° , dan 135° . Sedangkan jarak antar piksel biasanya ditetapkan 1 piksel. Matriks kookurensi merupakan matriks bujursangkar dengan jumlah elemen sebanyak kuadrat jumlah level intensitas piksel pada citra. Setiap titik (p, q) pada matriks kookurensi berorientasi θ berisi peluang kejadian piksel bernilai p bertetangga dengan piksel bernilai q pada jarak d serta orientasi θ dan $(180-\theta)$.

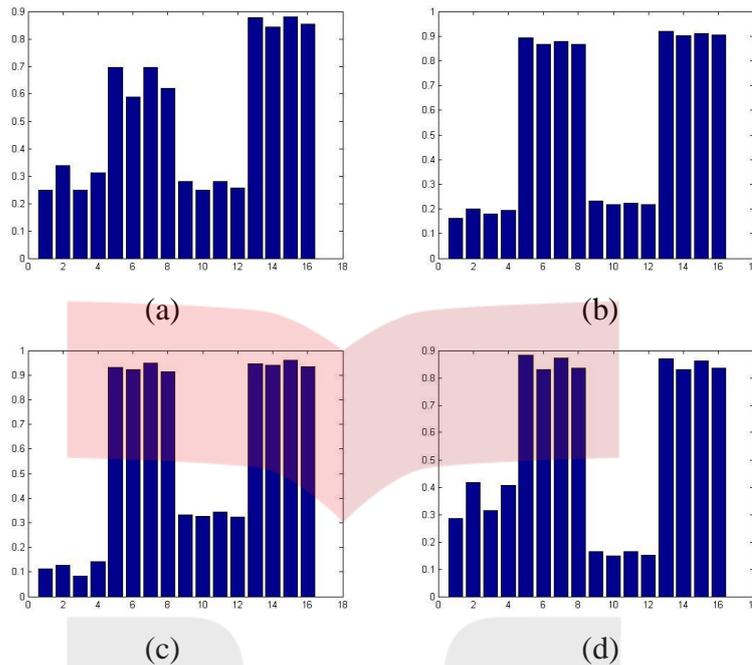


Gambar 3.5 Ilustrasi pembuatan matriks kookurensi^[15]

- (a) citra *input*
- (b) nilai intensitas citra *input*
- (c) matriks kookurensi 0^0
- (d) matriks kookurensi 45^0
- (e) matriks kookurensi 90^0
- (f) matriks kookurensi 135^0

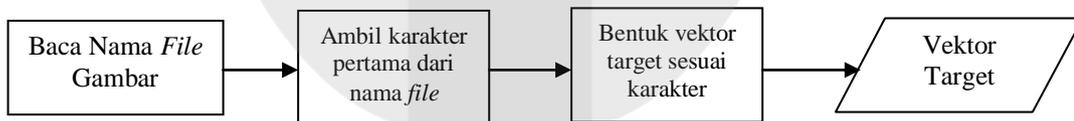
- Penghitungan nilai properti GLCM

Setelah didapat matriks kookurensi untuk masing-masing orientasi sudut, akan dihitung nilai properti dari matriks tersebut. Properti GLCM yang digunakan antara lain : *contrast*, *correlation*, *homogeneity*, dan *energy*. Jadi akan didapat 16 variabel dari proses ini berupa empat variabel untuk orientasi sudut 0^0 , empat variabel untuk orientasi sudut 45^0 , empat variabel untuk orientasi sudut 90^0 , dan empat variabel untuk orientasi sudut 135^0 . Variabel ini akan dibentuk vektor berukuran 16×1 , dimana vektor ini merupakan *output* dari proses ekstraksi ciri dengan metode GLCM yang merepresentasikan fitur tekstur dari citra *input*.



Gambar 3.6 Plot Ciri Tekstur
 (a) *Acne*, (b) *Scabies*, (c) *Dermatitis*, (d) *Herpes*

3.2.2 Proses Pembentukan Vektor Target



Gambar 3.7 Diagram Alir Proses Pembentukan Vektor Target

JST *Backpropagation* termasuk ke dalam metode *supervised learning*. Oleh karena itu diperlukan vektor-vektor target dalam proses pelatihan JST *backpropagation*. Dalam sistem ini vektor target dibentuk dari nama *file* citra yang di-*input*. Dalam sistem ini semua nama *file* data uji diberi nama dengan menggunakan beberapa aturan untuk memudahkan pembentukan vektor target. Karakter pertama dari nama *file* menandakan kelas dari citra tersebut. Untuk citra *acne* menggunakan karakter awal “1”, citra *scabies* menggunakan karakter awal “2”, citra *dermatitis* menggunakan karakter awal “3”, dan untuk citra *herpes* menggunakan karakter awal “4”. Pada proses ini akan dibaca karakter awal dari nama *file* data uji dimana akan dilakukan pembentukan vektor target sesuai

karakter awal ini. Untuk karakter awal “1” akan digunakan [1;0;0;0] sebagai vektor target, untuk karakter awal “2” akan digunakan [0;1;0;0] sebagai vektor target, untuk karakter awal “3” akan digunakan [0;0;1;0] sebagai vektor target, dan untuk karakter awal “4” akan digunakan [0;0;0;1] sebagai vektor target.

3.2.3 Proses *Training JST Backpropagation*

Proses *training* ini merupakan proses dimana akan dibentuk sebuah jaringan yang mampu mengenali pola dari masing-masing penyakit kulit. Awal dari proses ini akan pembentukan sebuah JST dengan parameter yang telah ditentukan oleh *user*. Parameter ini antara lain *learning rate* (α), *error target*, maksimum *epoch*, fungsi *training*, fungsi aktivasi, jumlah *hidden layer* dan jumlah *neuron* tiap-tiap *layer*. Parameter tersebut akan diinisialisasi dengan nilai sebagai berikut :

- $\alpha = 0,01$
- *error target* = 0,001
- maksimum *epoch* = 5000
- jumlah *neuron* pada *input layer* = 25
- jumlah *neuron* pada *output layer* = 4

Untuk fungsi *training*, fungsi aktivasi, jumlah *hidden layer*, dan jumlah *neuron* tiap *hidden layer* tidak ditentukan secara pasti nilainya sebab parameter-parameter ini akan diuji nantinya untuk mengetahui pengaruh dari parameter tersebut pada akurasi sistem. Untuk nilai *error* akan digunakan metode *msereg* (*mean squared error with generalization*). Metode ini merupakan turunan dari metode *mse* (*mean square error*). Metode ini dipilih untuk menambah kemampuan generalisasi sistem. Sistem dengan generalisasi yang baik mampu mengenali data uji lebih baik. Persamaan matematis dari *msereg* adalah :

$$msereg = \gamma mse + (1 - \gamma) msw \tag{3.2}$$

$$mse = \frac{1}{N} \sum_{i=1}^N (e_i)^2 \tag{3.3}$$

$$msw = \frac{1}{n} \sum_{j=1}^n (w_j)^2 \tag{3.4}$$

Dimana :

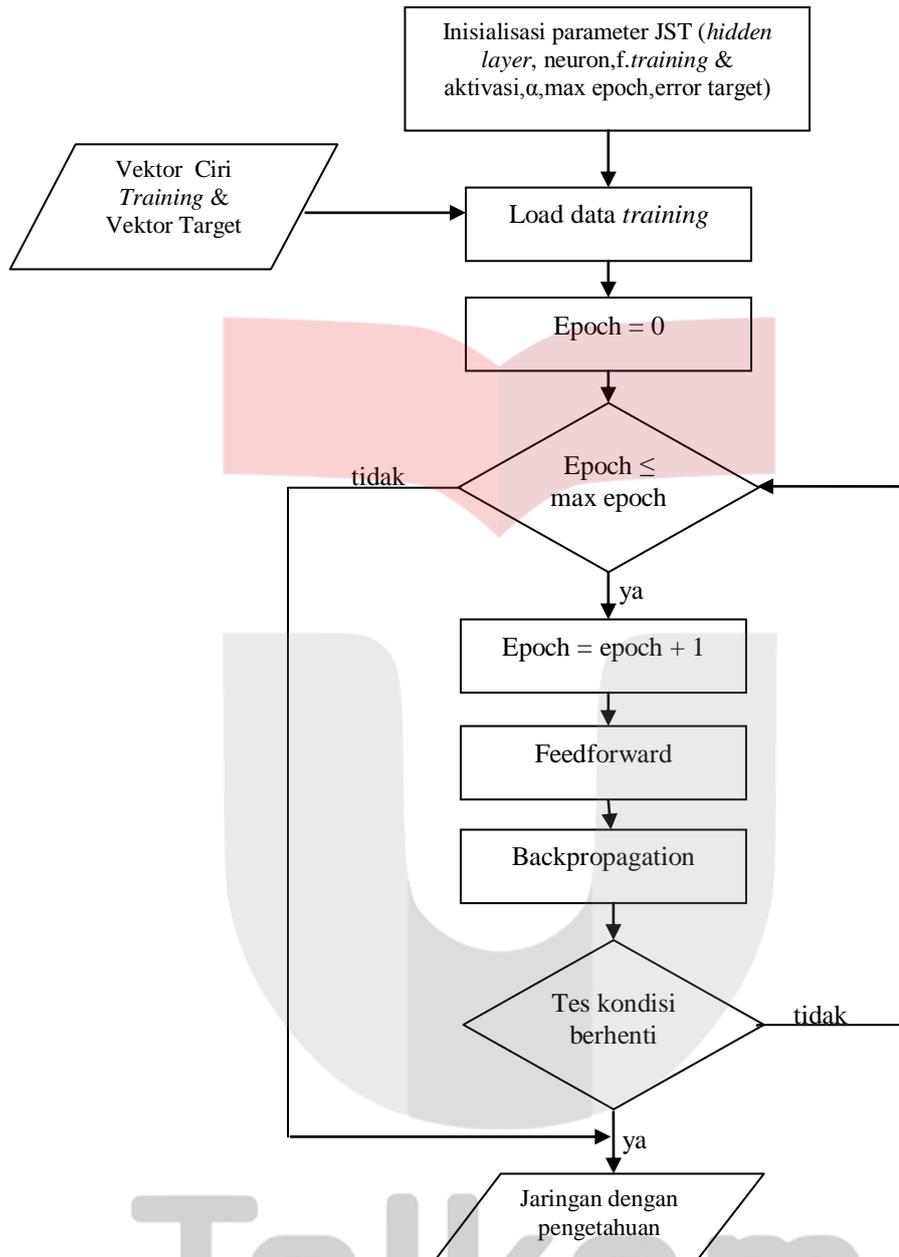
mse adalah *mean square error*

msw adalah *mean squared weight*

γ adalah *performance ratio*, dimana sistem ini akan menggunakan $\gamma = 0,5$

Setelah parameter tersebut diinisialisasi, maka proses selanjutnya adalah *me-load data training*. *Data training* adalah data yang berisi vektor-vektor ciri uji beserta vektor-vektor target. Selanjutnya *training* akan dimulai. Dalam *training backpropagation* terdapat dua langkah utama yaitu *feedforward* dan *backpropagation*. Masing-masing langkah ini telah dijelaskan pada bab II. *Training* akan berhenti apabila maksimum epoch tercapai atau kondisi berhenti sudah terpenuhi. Setelah *training* berhenti akan didapatkan jaringan yang sudah mempunyai pengetahuan. Selanjutnya jaringan ini akan diuji peformansinya pada proses klasifikasi.

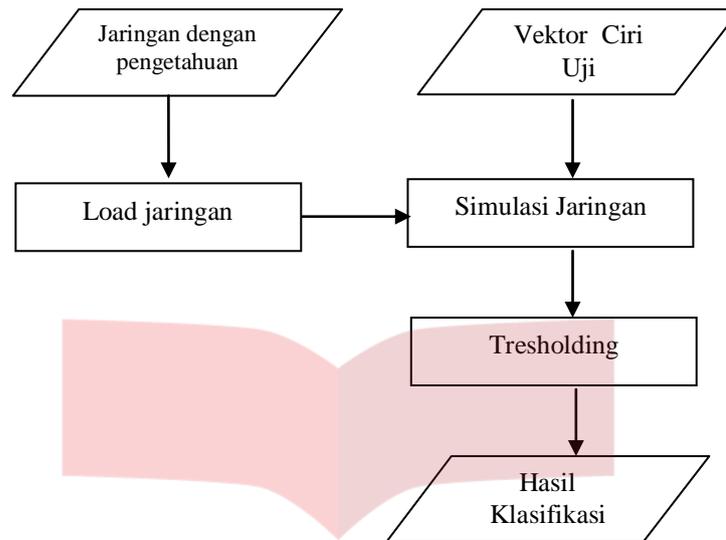




Gambar 3.8 Diagram Alir Proses *Training JST Backpropagation*

3.2.4 Proses Klasifikasi

Proses klasifikasi merupakan proses terakhir dalam sistem pendeteksian ini. Klasifikasi hanya dapat dilakukan bila proses *training* telah dilakukan. Proses ini terdiri tiga proses-proses kecil yang terdapat di dalamnya. Proses-proses itu antara lain *load* jaringan, simulasi jaringan, dan *thresholding*.



Gambar 3.9 Diagram Alir Proses Klasifikasi

a) Load jaringan

Pada proses ini jaringan hasil *training* akan diambil, dimana jaringan ini akan diuji performanya dalam mengklasifikasi data uji baru.

b) Simulasi jaringan

Setelah proses load jaringan selesai maka proses selanjutnya adalah simulasi jaringan. Simulasi jaringan dilakukan dengan meng-*input* vektor ciri data uji. *Output* dari proses ini nantinya akan dihasilkan sebuah vektor. Vektor ini akan diolah pada proses *thresholding* untuk mendapatkan hasil klasifikasi berupa jenis penyakit kulit.

c) *Thresholding*

Vektor hasil dari simulasi akan diberikan nilai *threshold*, dimana fungsi dari nilai *threshold* ini adalah untuk memetakan nilai pada vektor hasil simulasi ke nilai 0 atau 1. Bila nilai-nilai elemen pada vektor lebih dari nilai *threshold* ini maka nilai elemen tersebut akan dipetakan ke 1. Sebaliknya bila nilai elemen pada vektor kurang dari atau sama dengan nilai *threshold* maka elemen tersebut akan dipetakan ke 0. Secara matematis ditulis :

$$f(x) = \begin{cases} 0, & x \leq a \\ 1, & x > a \end{cases} \quad (3.5)$$

Dimana :

a merupakan nilai *threshold*

x merupakan nilai elemen pada vektor

Sistem ini akan menggunakan nilai *threshold* sebesar 0,5. Setelah proses *threshold* akan didapatkan vektor, dimana vektor ini akan menunjukkan penyakit kulit yang terdeteksi oleh sistem. Vektor dengan nilai [1;0;0;0] akan dideteksi sebagai penyakit *acne*, vektor [0;1;0;0] akan dideteksi sebagai *scabies*, vektor [0;0;1;0] akan dideteksi sebagai *dermatitis*, dan vektor [0;0;0;1] akan dideteksi sebagai *herpes*. Bila *thresholding* menghasilkan vektor selain yang disebutkan tadi, artinya sistem tidak bisa mengenali citra tersebut.

3.3 Peformansi Sistem

Untuk mengukur peformasi dari sistem akan digunakan dua parameter yaitu tingkat akurasi dan waktu komputasi.

3.3.1 Tingkat Akurasi

Tingkat akurasi menunjukkan kemampuan sistem dalam mengenali data uji dengan benar. Tingkat akurasi adalah persentase perbandingan antara jumlah data uji yang mampu dikenali dengan benar dengan jumlah data uji keseluruhan. Secara matematis dapat ditulis :

$$Akurasi = \frac{Jumlah\ data\ uji\ yang\ dikenali\ dengan\ benar}{Jumlah\ data\ uji\ keseluruhan} \times 100\% \quad (3.6)$$

Rentang nilai dari tingkat akurasi adalah 0% sampai dengan 100%. Nilai 0% menyatakan bahwa tidak ada data uji yang dikenali dengan benar dan nilai 100 % menyatakan bahwa semua data uji dapat dikenali dengan benar oleh sistem.

3.3.2 Waktu Proses

Waktu proses adalah waktu yang dibutuhkan oleh sistem untuk melakukan pendeteksian terhadap sebuah data uji. Waktu proses dihitung mulai dari *input* citra sampai dengan waktu keluarnya *output*.