

Klasifikasi Berbagai Jenis Jamur Layak Konsumsi dengan Metode Backpropagation

¹Ruth Hanseliani

Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Sanata Dharma Yogyakarta
Paingan, Maguwoharjo, Depok, Sleman, Daerah Istimewa Yogyakarta, Indonesia Telp., (0274)
883037, 883968 Fax. (0274) 886529
E-Mail: 155314103@student.usd.ac.id

²Cyprianus Kuntoro Adi

Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Sanata Dharma Yogyakarta
Paingan, Maguwoharjo, Depok, Sleman, Daerah Istimewa Yogyakarta, Indonesia Telp., (0274)
883037, 883968 Fax. (0274) 886529
E-Mail: kuntoroadi@usd.ac.id

ABSTRACT

Mushrooms are a potential source of vegetable food around us. But there are difficulties in recognizing different types of mushrooms worthy of consumption because they have similarities when viewed visually. This study aims to determine the accuracy of the backpropagation method in classifying various types of mushrooms suitable for consumption automatically. The steps taken begin with data capture, initial processing, feature extraction and classification. The data used are images of six types of mushrooms worthy of consumption including button mushrooms, ears, straws, portabella, shitake, and ash oysters. Total data used 222 mushroom data, each type consists of 37 image data. The data sharing is 216 data as training data and testing data, and 6 data are used for single data test. The initial processing stages are grayscale, adjustment, binaryzation, noise reduction and resizing. Feature extraction is done to obtain color characteristics (RGB and HSI) and texture characteristics (texture statistics and gray level co-occurrence matrices). The test results obtained an accuracy of 97%.

Keywords: HSI, texture statistics, GLCM, backpropagation, mushrooms

PENDAHULUAN

Jamur memiliki banyak keistimewaan antara lain kandungan gizi, kelezatan rasa, serta memiliki segudang manfaat dan khasiat. Selain itu pada saat ini, bisnis budidaya jamur serta pengolahannya juga memiliki daya tarik tersendiri. Hal itu terbukti dengan munculnya UKM kecil, menengah maupun besar yang menjadikan jamur sebagai bahan utama bisnis mereka. Namun hal tersebut belum diikuti oleh pengetahuan masyarakat luas akan khasiat jamur. Ditambah lagi masih ada masyarakat yang takut mengonsumsi jamur, bingung bagaimana cara mengolahnya, serta belum mengenali jenis-jenis jamur layak konsumsi [1].

Kesulitan klasifikasi berbagai jenis jamur layak konsumsi inilah yang merupakan permasalahan yang terjadi di masyarakat. Beberapa jenis jamur layak konsumsi memiliki bentuk yang hampir sama (mirip) dilihat secara visual dari warna, bentuk, tekstur dan ukurannya pada umumnya. Itu sebabnya diperlukan sistem komputer yang mampu melakukan klasifikasi berbagai jenis jamur layak konsumsi secara otomatis sebagai salah

satu alternatif teknologi dengan memanfaatkan ilmu teknologi informasi saat ini.

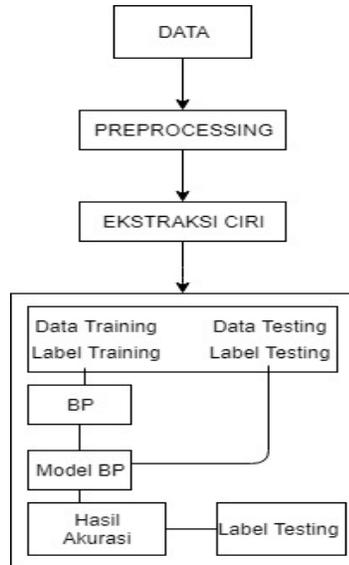
Ekstraksi ciri bertujuan untuk menajamkan perbedaan-perbedaan pola sehingga akan sangat memudahkan dalam pemisahan kategori kelas pada tahap klasifikasi. Hal yang tidak kalah penting adalah pemilihan metode ekstraksi ciri yang tepat sesuai dengan karakteristik data citra jamur. Tidak dapat dilakukan sembarang karena akan berkaitan dengan features analysis. Ekstraksi ciri yang digunakan dalam penelitian ini adalah metode rerata RGB dan metode HSI untuk ciri warna serta metode Statistik Tekstur dan GLCM untuk ciri tekstur.

Metode klasifikasi yang digunakan pada penelitian ini adalah jaringan syaraf tiruan *backpropagation*. Cara kerja jaringan syaraf tiruan sama dengan mengadopsi cara manusia belajar untuk mengendalikan sebuah pola informasi. Jaringan syaraf tiruan yang dibangun bertugas mengklasifikasikan berbagai jenis jamur layak konsumsi berdasarkan data berupa gambar. Hasil akurasi yang diperoleh dalam penelitian ini akan menjadi ukuran seberapa besar akurasi jaringan syaraf tiruan *backpropagation* dalam

mengklasifikasi berbagai jenis jamur layak konsumsi.

METODE PENELITIAN

Pada proses klasifikasi ini dilakukan dua tahap pembelajaran yakni tahap pelatihan dan pengujian. Proses pelatihan dilakukan untuk menemukan model klasifikasi yang akan digunakan dalam proses klasifikasi ini. Gambar 1 menampilkan alur kerja dari klasifikasi jenis jamur.



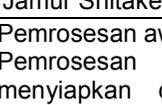
Gambar 1. Diagram blok Gambaran Umum Klasifikasi Jenis Jamur

Blok diagram diatas terdiri dari beberapa modul yaitu :

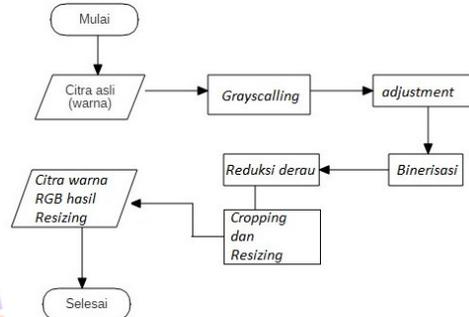
- a. Data
 Bagian ini bertujuan untuk mengambil (*capture*) citra berbagai jenis jamur layak konsumsi. Proses pengambilan data citra menggunakan kamera, kemudian obyek (jamur) diletakkan pada sebuah kotak kosong. Citra jamur yang telah dicapture kemudian disimpan dalam bentuk file dengan format *.jpg. Data citra jamur yang digunakan dalam penelitian ini adalah jamur yang masih dalam keadaan baik (lengkap) serta dalam keadaan segar.

Tabel 1. Data Gambar Jamur Layak Konsumsi

	
Jamur Kancing	Jamur Kuping
	
	Jamur Portabella

Jamur Merang	
Jamur Shitake	
Jamur Tiram Abu	

- b. Pemrosesan awal
 Pemrosesan awal merupakan tahap menyiapkan data sebelum dilakukan ekstraksi ciri. Pada Gambar 3 menampilkan alur pemrosesan awal klasifikasi jenis jamur.



Gambar 2. Alur Pemrosesan awal Klasifikasi Jenis Jamur

Adapun tahapan dalam pemrosesan awal klasifikasi jenis jamur yakni dimulai dari melakukan *grayscale*, penghilangan bayangan (*adjustment*), binarisasi, reduksi derau, *cropping* serta *resizing*.

- Proses *grayscale* mengubah citra berwarna yang memiliki 3 komponen warna R, G, dan B menjadi 1 komponen warna dengan memberikan sebuah nilai bobot kepada 3 komponen R, G, dan B kemudian dijumlahkan untuk memperoleh intensitas warna abu-abu [2]. Persamaan yang umum dipakai untuk mengubah citra berwarna menjadi citra berskala keabuan menggunakan standar NTSC (*National Television System Committee*) yang dapat dinyatakan sebagai berikut :

$$g(x, y) = 0.299 \times R(x, y) + 0.587 \times G(x, y) + 0.144 \times B(x, y) \dots(1)$$

Dengan :
 $g(x, y)$ = citra yang akan dikonversi menjadi citra berskala keabuan.
 $R(x, y)$ = nilai piksel pada lokasi (x, y) untuk komponen R.
 $G(x, y)$ = nilai piksel pada lokasi (x, y) untuk komponen G.
 $B(x, y)$ = nilai piksel pada lokasi (x, y) untuk komponen B

- Proses penghilangan bayangan yang ada disekitar objek termasuk dalam proses transformasi intensitas. Metode yang digunakan didasarkan pada manipulasi langsung atas piksel-piksel pada suatu citra. Adapun domain spasial mengacu pada bidang citra itu sendiri. Proses-proses domain spasial dinotasikan dengan :

$$g(x, y) = T [f(x, y)] \dots\dots\dots (2)$$

Dimana $f(x,y)$ adalah citra masukan, $g(x,y)$ adalah citra keluaran (hasil dari proses), dan T adalah sebuah operator pada f , yang didefinisikan pada lingkungan tertentu disekitar titik (x,y) .

- Proses binerisasi merupakan sebuah proses untuk mengubah citra grayscale menjadi citra hitam putih yang memiliki 1 keping yakni dengan nilai 0 dan 1. Pada dasarnya citra dipisahkan menjadi 2 bagian yakni obyek dari citra dan *background*. Dalam mengonversi citra abu menjadi citra hitam putih sekaligus dilakukan proses pemisahan antara obyek dengan *background* maka dilakukan proses pengecekan nilai setiap piksel terhadap nilai ambang atau sering disebut juga dengan pengambangan intensitas atau *thresholding*. Nilai ambang ditentukan dengan terlebih dahulu melihat histogram citra dan dipilih nilai sebagai ambang pada bagian lembah. Secara sistematis, pengambangan intensitas dinyatakan melalui persamaan :

$$g(x, y) = \begin{cases} 1, & \text{jika } f(x, y) > T \\ 0, & \text{jika } f(x, y) \leq T \end{cases} \dots\dots\dots(3)$$

- Proses reduksi derau penghilangan derau) ini dilakukan pada citra yang memiliki derau yang dalam skala kecil disekitar objek Median filter adalah sebuah noise removal filter yang menerapkan sebuah algoritma smoothing yang teratur dengan tujuan menghilangkan noise pada data citra dengan tetap menjaga sisi pada obyek di citra. Pemilihan median filter yang digunakan untuk menghilangkan noise dengan tetap menjaga obyek yang ada pada citra [3].

- Proses cropping atau pemotongan gambar adalah mengambil bagian obyek jamur saja tanpa mengikutsertakan bagian *background* dari data citra. Alur cropping yakni mengetahui lebih dahulu titik teratas, titik terbawah, titik ter kiri serta titik terkanan dari obyek data citra tersebut yang telah diubah menjadi citra biner. Lalu nilai koordinat dari titik-titik tersebut disimpan, untuk selanjutnya digunakan dalam cropping citra warna dari data tersebut.

- Mengubah ukuran citra (*resizing image*) artinya adalah mengubah besarnya ukuran citra digital dalam piksel. Ada kalanya ukurannya berubah menjadi lebih kecil dari file aslinya dan ada kalanya sebaliknya (Rendra, 2014). Resizing dalam hal memperkecil ukuran citra dilakukan untuk mengurangi jumlah piksel pada citra. Pada proses resizing untuk memperbesar ukuran citra dilakukan dengan menambah jumlah piksel pada citra.

c. Ekstraksi Ciri

Pada bagian ini bertujuan untuk mendapatkan atau memperoleh ciri dari setiap data citra jamur layak konsumsi. Pada penelitian ini digunakan ciri warna dan ciri tekstur. Ciri warna yakni dari warna RGB dan warna HIS (*hue, saturation* dan *intensity*). Dalam mengekstrak ciri tekstur citra yang digunakan yakni citra abu (*grayscale*). Adapun ciri tekstur yakni menggunakan metode Statistik Tekstur dan *GLCM*. Metode statistik tekstur terdiri dari *mean, standard deviasi, smoothness (roughness), skewness, uniformity*, serta *entropy*. Metode *GLCM* yakni *mean, variance, energy, entropy, contrast, dissimilarity, homogeneity*, dan *correlation*.

• Metode rerata R, G, B dan RGB untuk ciri warna

Ciri warna dengan menghitung rerata nilai keping *Red*, keping *Green*, keping *Blue*, serta rerata dari RGB. Langkah awal yakni memisahkan keping *Red*, *Green*, dan *Blue* terlebih dahulu. Hasil pemisahan itu akan digunakan untuk menghitung rerata dari tiap keping *Red*, *Green*, *Blue* yang ada pada objek citra. Rerata keping *Red* (keping merah) dihitung dengan lebih dahulu

menjumlahkan seluruh nilai-nilai intensitas yang terdapat dalam keping *Red* kemudian hasil penjumlahan tersebut dibagi dengan total jumlah piksel dari objek tersebut. Total jumlah piksel objek dihitung dengan cara mengalikan baris dan kolom dari objek tersebut. Cara demikian juga dilakukan pada keping *Green* dan *Blue*. Untuk rerata *R*, rerata *G*, rerata *B* dengan rumus :

$$Mean = \frac{1}{M \times N} \sum_{i=1}^M \sum_{j=1}^N P_{ij} \dots\dots(4)$$

Untuk rerata RGB dengan rumus:

$$Mean\ RGB = \frac{MeanR + MeanG + MeanB}{M \times N} \dots(5)$$

• **Metode HSI untuk ciri warna**

Ciri warna HSI (*Hue, Saturation, and Intensity*) memisahkan komponen intensitas dari informasi pembawa warna (hue dan saturasi) pada sebuah citra warna. Hasilnya, model HIS merupakan perangkat ideal dalam pengembangan algoritma-algoritma pemrosesan citra berbasis deskripsi-deskripsi yang alamiah untuk manusia [5]. Adapun rumus untuk memperoleh komponen *H* dari tiap piksel RGB didapatkan dengan persamaan :

$$H = \begin{cases} 0, & \text{jika } B \leq G \\ 360 - \theta, & \text{jika } B > G \end{cases} \dots\dots(6)$$

dengan:

$$\theta = \cos^{-1} \left\{ \frac{\frac{1}{2} [(R-G) + (R-B)]}{\sqrt{\frac{1}{4} [(R-G)^2 + (R-G)(G-B)]^2}} \right\} \dots\dots(7)$$

Komponen saturasi diberikan oleh :

$$S = 1 - \frac{3}{(R+G+B)} [\min(R, G, B)] \dots\dots(8)$$

$$I = \frac{1}{3} (R + G + B) \dots\dots(9)$$

Keterangan rumus :

H = nilai *Hue*

S = nilai *Saturation*

I = nilai *Intensity*

• **Metode Statistik Tekstur**

Rekaman citra terhadap objek-objek real di alam tidak selalu menunjukkan region-region citra dengan intensitas atau warna yang uniform atau

homogeny. Misalnya citra permukaan lantai kayu parket tidaklah uniform, tetapi mengandung variasi intensitas warna yang membentuk pola berulang. Analisis tekstur dan segmentasi citra berdasarkan tekstur selalu mengacu pada dua hal: (1) analisis tingkat kekasaran permukaan serta (2) analisis struktur pola dan orientasinya. Dua hal ini telah menjadi dasar bagi pengembangan teori tentang analisis tekstur citra. Pendekatan analisis statistik (*statistical approach*) telah dikembangkan untuk mengetahui kekasaran permukaan tekstur, sedangkan untuk mengetahui bentuk atau pola tekstur dapat dilakukan melalui pendekatan terstruktur [6].

a) *Mean* untuk menghitung rata-rata intensitas area citra.

$$\mu = \sum_{i=0}^{L-1} i \cdot p(i) \dots\dots(10)$$

b) *Standard deviation* untuk menghitung rata-rata kontras area citra.

$$\sigma = \sqrt{M_2} \dots\dots(11)$$

c) *Smoothness* atau *Roughness* untuk menghitung kehalusan atau kekasaran permukaan (tinggi rendahnya variasi intensitas atau kontras) region. *R=0*, intensitas permukaan region rata atau homogeny (kontras=0). *R* mendekati 1, permukaan region kasar ataupun memiliki variasi intensitas atau kontras tinggi.

$$R = \frac{\sigma^2}{1 + \sigma^2} \dots\dots(12)$$

d) *Skewness* untuk mengukur tingkat simetri histogram atau kecondongan distribusi intensitas terhadap posisi nilai rata-rata (μ). *Skewness 0*, distribusi intensitas $<\mu$ sama besar dengan distribusi intensitas $>\mu$. *Skewness positif*, histogram lebih condong melebar ke kanan (intensitas $>\mu$ lebih banyak). *Skewness negatif*, histogram lebih condong melebar ke kiri (intensitas $<\mu$ lebih banyak).

$$M_3 = \sum_{i=0}^{L-1} (i - \mu)^3 p(i) \dots\dots(13)$$

e) *Uniformity* untuk mengukur keseragaman nilai intensitas. *U* maksimum (*U=1*) bila semua piksel dalam area memiliki nilai intensitas sama.

$$U = \sum_{i=0}^{L-1} p^2(i) \dots\dots\dots(14)$$

- f) Entropy untuk Mengukur tingkat keacakan intensitas. E maksimum bila piksel-piksel dalam area memiliki nilai sangat acak. E=0 bila semua piksel itu memiliki nilai intensitas sama.

$$E = - \sum_{i=0}^{L-1} p(i) \log_2 p(i) \dots\dots\dots(15)$$

• **Metode Gray Level Co-occurrence Matrices untuk ciri tekstur**

Matriks co-occurrence (CM) merupakan satu dari sejumlah metode analisis tekstur. *Gray Level Co-occurrence Matrices* (GLCM) yang dikembangkan oleh Robert Haralick merupakan metode yang paling sering digunakan. Robert Haralick telah mendefinisikan beberapa karakteristik / fitur tekstur citra yang dihitung dari matriks co-occurrence (CM) dengan menyatakan matriks probabilitas $P(i, j) = CM(i, j) / \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} CM(i, j)$. Diantara fitur tersebut adalah *mean* (μ_x, μ_y), *variance*, *homogeneity*, *dissimilarity*, *contrast*, *entropy*, dan *energy*[6].

- a. *Mean* merupakan rata-rata distribusi probabilitas $P(i, j)$ yang dapat dihitung berdasarkan sampel x dan sampel y, dimana $\mu_x \approx \mu_y$

$$\mu_x = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} P(i, j) \dots\dots\dots(16)$$

- b. *Variance* menentukan sebaran nilai atau simpangan terhadap nilai rata-rata data (tekstur). Semakin kecil nilai variansi maka tekstur semakin homogeny, dan sebaliknya. Variansi dapat dihitung berdasarkan sampel x dan sampel y, dimana $\sigma_x^2 \approx \sigma_y^2$

$$\sigma_x^2 = \sum_{i=0}^{G-1} (i - \mu_x)^2 \sum_{j=0}^{G-1} P(i, j) \dots\dots\dots(17)$$

- c. *Uniformity of Energy* adalah ukuran homogenitas lokal dan merupakan kebalikan dari *entropi*. Fitur ini digunakan untuk melihat tingkat keseragaman tekstur. Semakin tinggi nilai energi, semakin tinggi tingkat homogenitas tekstur. Nilai energi ada pada kisaran [0, 1], dimana 1 menggambarkan area yang homogen.

$$Energy = \sum_{j=0}^{G-1} \sum_{i=0}^{G-1} (P(i, j))^2 \dots\dots\dots(18)$$

- d. Entropy adalah ukuran tingkat keacakan (*randomness*) permukaan tekstur akibat gangguan spasial atau frekuensi. Nilai entropi dapat memberi informasi fitur permukaan tekstur yang kasar atau halus. Semakin nilai entropi mendekati satu maka tingkat kekasaran tekstur semakin tinggi, sebaliknya nilai entropi mendekati nol maka permukaan tekstur semakin halus.

$$Entropy = - \sum_{j=0}^{G-1} \sum_{i=0}^{G-1} P(i, j) \log(P(i, j)) \dots\dots\dots(19)$$

- e. Contrast adalah variasi nilai intensitas lokal dalam matriks co-occurrence. Jika piksel tetangga memiliki nilai intensitas yang mirip atau berdekatan, maka kontras tekstur sangat rendah. Nilai kontras tinggi menunjukkan tekstur dengan variasi intensitas yang tinggi, untuk nilai kontras rendah menunjukkan tekstur yang halus atau lembut.

$$Contrast = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} (i - j)^2 P(i, j) \dots\dots\dots(20)$$

- f. Dissimilarity (ketidakmiripan) adalah ukuran yang mendefinisikan variasi tingkat intensitas pasangan piksel dalam citra. Mirip dengan kontras yang dihitung dengan jarak Euclidean, dissimilarity dihitung dengan jarak city block.

$$Dissimilarity = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} |i - j| P(i, j) \dots\dots\dots(21)$$

- g. Homogeneity mengukur tingkat homogenitas perulangan struktur tekstur, dimana bobot nilainya merupakan invers dari kontras (pers 2...). Namun, dalam MATLAB homogeneity merupakan invers dari dissimilaritas (pers 2...). Tingkat homogeneity tekstur sangat tinggi bila nilai matriks co-occurrence terkonsentrasi sepanjang diagonal matriks. Ini menunjukkan bahwa ada banyak piksel dengan perulangan pasangan nilai intensitas yang sama sebagai pembentuk struktur tekstur. Homogeneity memiliki kisaran nilai [0 1]. Untuk nilai homogeneity sama dengan 1 menunjukkan bahwa tekstur

memiliki struktur perulangan yang ideal, sedang bila nilainya rendah menunjukkan bahwa elemen tekstur memiliki variasi yang tinggi dan tersebar secara merata dalam area tekstur.

$$\text{Homogeneity} = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{P(i,j)}{1+(i-j)^2} \dots (22)$$

$$\text{Homogeneity} = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{P(i,j)}{1+|i-j|} \dots (23)$$

- h. Correlation adalah ukuran keterhubungan linear tingkat keabuan satu piksel relative terhadap piksel lainnya pada posisi tertentu.

$$\text{Correlation} = \sum_{i=0}^{G-1} \sum_{j=0}^{G-1} \frac{(i-\mu_x)(j-\mu_y)P(i,j)}{\sigma_x \sigma_y} \dots (24)$$

d. Klasifikasi

Pada bagian ini adalah tahap pengenalan. Data testing diujikan ke data training menggunakan metode backpropagation. *Backpropagation* merupakan algoritma yang terawasi dan biasanya digunakan oleh perceptron dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyiannya. *Backpropagation* melatih jaringan untuk mendapatkan keseimbangan antara kemampuan jaringan untuk mengenali pola yang digunakan selama pelatihan serta kemampuan jaringan untuk memberikan respon yang benar terhadap pola masukan yang serupa (tapi tidak sama) dengan pola yang dipakai selama pelatihan.

- **Pelatihan Standar Backpropagation**
Menurut Siang (2009), pelatihan *backpropagation* meliputi 3 fase. Fase pertama adalah fase maju. Pola masukan dihitung maju mulai dari layar masukan hingga layar keluaran menggunakan fungsi aktivasi yang ditentukan. Fase kedua adalah fase mundur. Selisih antara keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan tersebut dipropagasikan mundur, dimulai dari garis yang berhubungan langsung dengan unit-unit di layar keluaran. Fase ketiga adalah modifikasi bobot untuk menurunkan kesalahan yang terjadi [7].

❖ **Fase I : Propagasi Maju**

Selama propagasi maju, sinyal masukan (=xi) dipropagasikan ke layar tersembunyi menggunakan fungsi aktivasi yang ditentukan. Keluaran dari setiap unit layar tersembunyi (=zj) tersebut selanjutnya dipropagasikan maju lagi ke layar tersembunyi di atasnya menggunakan fungsi aktivasi yang ditentukan. Demikian seterusnya hingga menghasilkan keluaran jaringan (=yk).

Berikutnya, keluaran jaringan (=yk) dibandingkan dengan target yang harus dicapai (=tk). Selisih tk-yk adalah kesalahan yang terjadi. Jika kesalahan ini lebih kecil dari batas toleransi yang ditentukan, maka iterasi dihentikan. Akan tetapi apabila kesalahan masih lebih besar dari batas toleransinya, maka bobot setiap garis dalam jaringan akan dimodifikasi untuk mengurangi kesalahan yang terjadi.

❖ **Fase II : Propagasi Mundur**

Berdasarkan kesalahan tk-yk, dihitung faktor (k = 1,2,...,m) yang dipakai untuk mendistribusikan kesalahan di unit yk ke semua unit tersembunyi yang terhubung langsung dengan yk. Dk juga dipakai untuk mengubah bobot garis yang berhubungan langsung dengan unit keluaran.

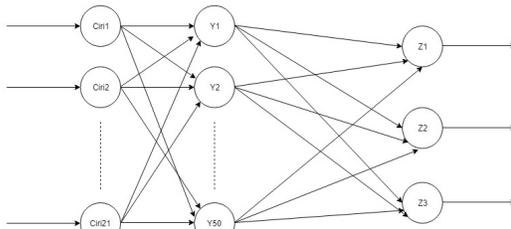
Dengan cara yang sama, dihitung faktor dj disetiap unit dilayar tersembunyi sebagai dasar perubahan bobot semua garis yang berasal dari unit tersembunyi di layar di bawahnya. Demikian seterusnya hingga semua faktor d di unit tersembunyi yang berhubungan langsung dengan unit masukan dihitung.

❖ **Fase III : Perubahan Bobot**

Setelah semua faktor d dihitung, bobot semua garis dimodifikasi bersamaan. Perubahan bobot suatu garis didasarkan atas faktor d neuron di layar atasnya. Sebagai contoh, perubahan bobot garis yang menuju ke layar keluaran didasarkan atas dk yang ada di unit keluaran.

Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi. Umumnya kondisi penghentian yang sering dipakai adalah jumlah iterasi atau kesalahan. Iterasi akan dihentikan jika jumlah iterasi yang dilakukan sudah melebihi jumlah maksimum iterasi yang

ditetapkan, atau jika kesalahan yang terjadi sudah lebih kecil dari batas toleransi yang diijinkan.



Gambar 3. Arsitektur Jaringan Klasifikasi Jenis Jamur

Setelah proses ekstraksi ciri selesai dilakukan, maka selanjutnya adalah tahap klasifikasi. Dalam tahap klasifikasi, langkah awal yang dilakukan adalah membagi data menjadi bagian ciri dan target. Kemudian langkah berikutnya adalah membagi data menjadi data training dan data testing. Digunakan 3-fold yakni 2/3 dari data sebagai data training, serta 1/3 dari data digunakan sebagai data testing. Baik untuk ciri maupun target, dilakukan pembagian training dan testing.



Gambar 4. Pembagian data training dan data testing.

Tidak lupa juga diikuti dengan pemberian label training dan label testing. Untuk pemberian label training dan label testing, dilakukan pengubahan kelas menjadi kelas biner. Adapun pembagian kelasnya yakni sebagai berikut :

Tabel 2. Kelas angka dan kelas biner

Kelas dalam angka	Jenis Jamur	Kelas dalam biner
1	Kancing	111
2	Kuping	110
3	Merang	101
4	Portabella	100
5	Shitake	011
6	Tiram Abu	010

Pada penelitian ini klasifikasi dilakukan sebanyak 50 iterasi. Jumlah neuron

yang divariasi dalam penelitian ini adalah 50 neuron dengan kelipatan 5. Dilakukan juga penghitungan hasil klasifikasi untuk setiap jenis fungsi training dan fungsi aktivasinya. Adapun fungsi training yang diujikan dalam penelitian ini adalah *trainlm*, *traindxd*, *traincgb*, *trainbfg*, *trainrp*, *trainscg*, *traincgb*, *traincgp*, *trainoss*. Sedangkan fungsi aktivasi yang juga diujikan dalam penelitian ini adalah *logsig*, *tansig* serta *purelin*.

Ketika dilakukan pengujian dengan berbagai fungsi training dan fungsi aktivasi serta variasi jumlah neuron dan jumlah layer tersembunyi, maka diperoleh hasilnya adalah fungsi training yang digunakan *trainlm* serta fungsi aktivasi yang dipilih yakni *logsig*. Saat proses klasifikasi yakni digunakan sejumlah 3 model atau 3 set yang telah dibuat diawal. Masing-masing model atau set dapat memberikan akurasi yang berbeda-beda. Selanjutnya untuk memperoleh akurasi totalnya yakni dilakukan penjumlahan dari akurasi ketiga model atau set tersebut kemudian dibagi 3 untuk mendapatkan rerata akurasinya. Kemudian hasilnya dibandingkan dengan label testing. Hasil dari rerata ketiga akurasi tersebut menjadi akurasi terbaik.

PEMBAHASAN

1. Hasil Pemrosesan awal

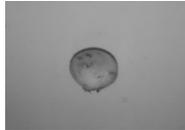
Ketika tahap pengambilan data telah selesai dilakukan, maka tahap selanjutnya yang dilakukan adalah pemrosesan awal atau *preprocessing*. Tahap *preprocessing* dilakukan untuk mempersiapkan data agar siap untuk diolah pada tahap berikutnya (ekstraksi ciri). Adapun tahapan *preprocessing* dalam penelitian ini yakni *grayscale*, penghilangan bayangan, binarisasi, reduksi derau, cropping serta resizing dengan hasil sebagai berikut :

➤ Citra Warna



Ketika citra berhasil dicapture dan disimpan dalam file berbentuk *.jpg, maka hasil yang diperoleh dari proses pengambilan data jika ditampilkan pada gambar diatas.

➤ Grayscale



Gambar diatas adalah contoh hasil dari proses grayscale yakni mengonversi citra warna menjadi citra berskala abu.

➤ Penghilangan bayangan



Gambar diatas merupakan citra hitam putih jamur Kancing. Jelas terlihat bahwa disekitar obyek terdapat banyak bayangan (derau dalam jumlah banyak).

➤ Binarisasi



Hasil dari citra abu yang telah dibersihkan dari bayangan kemudian dibinarisasi untuk menjadi citra biner. Hasilnya setelah dibinarisasi ini digunakan untuk proses berikutnya yakni cropping.

➤ Reduksi derau



Proses reduksi derau dibutuhkan untuk menghilangkan derau-derau (yang cenderung lebih kecil) yang masih tersisa disekitar obyek pada citra.

➤ Cropping



Proses pemotongan gambar dilakukan dengan terlebih dahulu mengetahui koordinat titik teratas, titik terbawah, titik ter kiri dan titik terkanan dari sebuah obyek pada citra jamur.

➤ Resizing



Proses resizing menghasilkan citra dengan ukuran yang berbeda dari ukuran aslinya. Pada proses ini, citra diresize sesuai dengan ukuran yang sudah ditentukan. Ukuran yang digunakan yakni 897x865 piksel.

2. Ekstraksi ciri

Hasil ekstraksi ciri metode rerata Red, Green, Blue dan RGB pada Tabel 4 yang menampilkan masing-masing ciri warna RGB setiap jenis jamur. Setiap ciri baik itu rerata Red, rerata Green, rerata Blue serta rerata RGB merepresentasikan obyek citra jamur dari setiap jenis jamur.

Tabel 3. Contoh hasil ekstraksi ciri warna RGB

Jenis jamur	Atribut			
	Rerata Red	Rerata Green	Rerata Blue	Rerata RGB
Kancing	133	122	110	122
Kuping	102	94	93	96

Tabel 4. Contoh hasil ekstraksi ciri warna HSI

Jenis jamur	Atribut		
	Hue	Saturation	Intensity
Kancing	0.209391	0.106604	0.476992
Kuping	0.294265	0.064746	0.377153

Tabel 5. Contoh hasil ekstraksi ciri Statistik Tekstur

Jenis jamur	Atribut		
	1	2	3
Kancing	124	24.064	0.998276
Kuping	96	61.824	0.999738

Jenis jamur	Atribut		
	4	5	6
Kancing	-9559.92	0.014373	6.3869
Kuping	139026	0.018063	6.3633

Keterangan :

- 1 = Mean
- 2 = Standard deviation
- 3 = Smoothness atau Roughness
- 4 = Skewness
- 5 = Uniformity
- 6 = Entropy

Table 6. Contoh hasil ekstraksi ciri GLCM

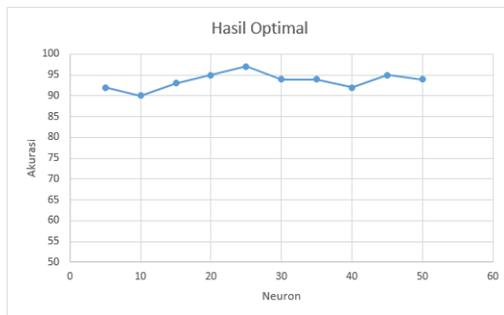
Jenis jamur	Atribut		
	Dissimilarity	contrast	Homogeneity
Kancing	1.38535	5.733934	0.597552
Kuping	1.743124	13.41421	0.617368

Jenis jamur	Atribut		
	energy	entropy	mean X
Kancing	0.003406	2.772548	123.9711
Kuping	0.007564	2.758092	96.03294

Jenis jamur	Atribut	
	varianceX	correlation
Kancing	578.7261	0.995576
Kuping	3819.699	0.998678

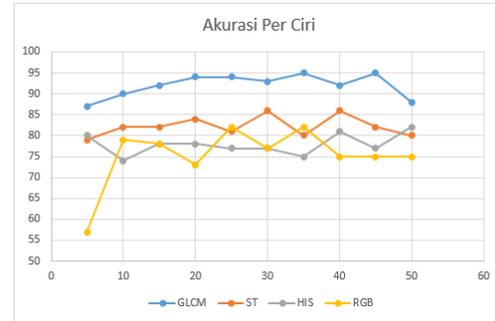
3. Klasifikasi

Beberapa uji coba dilakukan pada penelitian ini. Diantaranya adalah pemilihan fungsi training, pemilihan fungsi aktivasi, perbandingan ketika dilakukan resizing atau tidak, mengetahui ciri yang paling optimal, serta mengetahui akurasi terbaik dengan arsitektur optimal. Hasil yang diperoleh antara lain :



Gambar 5. Grafik Perbandingan akurasi per ciri

Setelah dilakukan percobaan untuk mengetahui ciri mana yang secara individual memiliki akurasi tertinggi, maka berdasarkan grafik perbandingan, ciri yang paling baik akurasinya adalah ciri tekstur GLCM yaitu dengan akurasi 95%.



Gambar 6. Grafik Hasil Akurasi Optimal

Berdasarkan hasil yang diperoleh dapat dilihat pada Gambar 6 yakni grafik hasil akurasi optimal atau terbaik. Hasil terbaik yakni 97% dengan input yakni 21 ciri, lalu akurasi optimal terdapat pada neuron ke 25 dengan jumlah layer tersembunyi yakni 1. Fungsi trainingnya adalah trainlm, fungsi aktivasinya adalah logsig serta citra yang digunakan yakni hasil resize yang sudah dilakukan penyamaan ukuran citra.

KESIMPULAN

Dalam hasil penelitian tentang klasifikasi berbagai jenis jamur layak konsumsi menggunakan metode backpropagation dapat disimpulkan beberapa hal sebagai berikut :

- 1) Metode *backpropagation* mampu mengklasifikasikan berbagai jenis jamur layak konsumsi dengan akurasi terbaik yakni 97% dengan menggunakan ciri warna dan ciri tekstur dengan total jumlah ciri yakni 21 ciri, jumlah layer tersembunyi yakni 1 serta berada pada neuron ke 25. Data tidak dinormalisasi serta sudah disamakan ukurannya (resizing). Fungsi training : trainlm, fungsi aktivasi : sigmoid biner (logsig).
- 2) Ciri yang paling berpengaruh dalam mengklasifikasikan berbagai jenis jamur layak konsumsi yakni ciri tekstur dengan metode GLCM dengan akurasi sebesar 95%. kesimpulan yang diambil berdasarkan penelitian yang telah dilakukan. Berisi pernyataan singkat tentang hasil yang disajikan dari pembahasan. Saran dapat dituliskan pada bagian paling akhir.

DAFTAR PUSTAKA

- [1] Jaelani. 2008. *Jamur Berkhasiat Obat*. Pustaka Obor Populer: Jakarta.
- [2] Kadir, A. & Susanto, A. 2013. *Teori dan Aplikasi Pengolahan Citra*. Penerbit ANDI. Yogyakarta
- [3] Irviantina, S. & Pardosi, I. 2016. Salt and Pepper Noise Removal dengan Spatial Median Filter dan Adaptive Noise

- Reduction. JSM STMIK Mikroskil Vol 17, No. 2. STMIK Mikroskil Medan.
- [4] Rendra Pranadipa dan kawan-kawan. Pengenalan Angka Pada Plat Nomor Dengan Metode Template Matching. Informatika, Universitas Brawijaya, Malang.
- [5] Sianipar, RH. 2018. *Dasar Pemrosesan Citra Digital dengan Matlab*. Penerbit ANDI. Yogyakarta.
- [6] Madenda, Sarifuddin. 2015. *Pengolahan Citra & Video Digital*. Penerbit Erlangga. Jakarta.
- [7] Siang, J.J. 2005. *Jaringan Saraf Tiruan dan Pemrogramannya Menggunakan MATLAB*. Penerbit ANDI. Yogyakarta.

