

Pengenalan Pola Tulang Daun Dengan Jaringan Syaraf Tiruan Backpropagation

Alvin Hanjaya Tandrian¹, Adhi Kusnadi²

Fakultas Teknik dan Informatika, Universitas Multimedia Nusantara, Gading Serpong, Indonesia
adhi.kusnadi@umn.ac.id

Diterima 30 Oktober 2018

Disetujui 21 Desember 2018

Abstract— The development of technology has affected many areas of life. Progress in the field of Computer Science can reach other aspect of science. This research apply the knowledge of Computer Science in Biological Science, the one is the morphology of leaf venation. Leaf venation is an important aspect in the process of identification. Therefore, in this research developed the system that classify the type of leaf venation. This application is used as means of research on the performance of pattern recognition on backpropagation neural network. The system designed using the Java programming and socket programming to transfer data from the mobile device into the computer. Data testing is implemented using Android to facilitate process of taking the picture. While in the process of training data for the optimal weight applied directly on the server computer by using Java Eclipse. In the stage of image processing is implemented by using the library of Canny edge detection. Data consisted of five categories of leaf vein pattern, with a sample of three leaves for each pattern. Training data using two of the three leaves for each pattern, with 10 images each leaf so that there are 20 images for each pattern, with a total of 100 images for all patterns. Data testing use 10 images from the third leaf to count the accuracy. The system managed to get the best accuracy by using an image size of 200 x 200 with 100 hidden node with the average accuracy of 76%.

Index Terms— *Android, Canny Edge Detection, Java, Neural Networks Backpropagation, Socket Programming*

I. PENDAHULUAN

Pada paper ini, diteliti pengenalan pola tulang daun dengan menggunakan jaringan syaraf tiruan backpropagation. Penelitian ini menerapkan salah satu Ilmu Komputer pada cabang Ilmu Biologi, yaitu morfologi daun. Menurut [1] ciri-ciri daun memegang peranan penting dalam proses identifikasi *species* dan taksonomi tumbuhan (ilmu yang mempelajari penamaan tumbuhan). Jenis dari tulang daun merupakan sebuah aspek penting dalam pengidentifikasian.

Penelitian yang dilakukan terinspirasi dari penelitian yang pernah dilakukan sebelumnya [2] [3]. Akan tetapi pada penelitian [2] tersebut hanya melakukan analisis tepi daun dan pada penelitian [3]

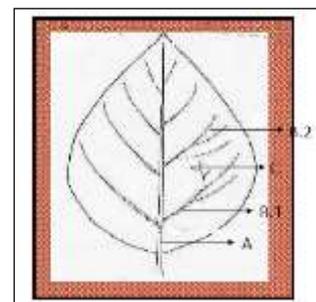
melakukan klasifikasi jenis bunga. Pada penelitian [4], identifikasi khusus untuk bunga kamboja, dan ada beberapa penelitian lain yang meneliti khusus pada jenis tanaman. Oleh karena itu pada penelitian ini dilakukan penelitian yang berbeda yaitu pengenalan pola daun yang dapat bermanfaat dapat mengenali jenis-jenis pohon secara umum.

Beberapa penelitian terkait mengenai identifikasi pola tulang daun ada yang dilakukan secara manual maupun berbeda dalam penggunaan algoritma antaranya [5] [6] [7]. Pada penelitian ini digunakan algoritma jaringan syaraf tiruan backpropagation, karena algoritma ini terbukti mampu mengenali berbagai macam objek pada penelitian pengenalan pada pengolahan citra. Berdasarkan hasil uji coba, sistem pengenalan pola daun menggunakan jaringan syaraf tiruan backpropagation telah berhasil dengan tingkat akurasi yang relative baik. Dengan adanya sistem ini akan mempermudah orang untuk mengetahui jenis tanaman berdasarkan pola daun

II. LANDASAN TEORI

A. Morfologi daun

Pada morfologi daun sebenarnya dapat dibedakan menjadi dua yaitu morfologi luar yang mempelajari dan mengamati bentuk fisik, maupun morfologi dalam, yang mempelajari bagian terdalam dari daun dengan menggunakan alat bantu mikroskopik [8].

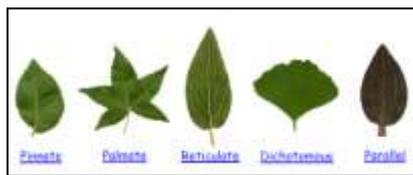


Gambar 1. Morfologi tulang daun

Pada Gambar 1 terlihat bahwa daun tersusun atas berbagai macam tulang daun. A menunjukkan ibu

tulang daun (*costa*), yang merupakan tulang utama yang langsung terhubung dengan pangkal daun. B merupakan tulang-tulang cabang (*nervus lateralis*), B.1 dan B.2 menunjukkan tingkatan cabangnya. C adalah urat-urat daun (*vena*) yang merupakan bagian terkecil dari struktural tulang daun[8].

Namun, ternyata berbagai tanaman tidak selalu mempunyai rupa tulang yang sama, walau pada dasarnya mempunyai tiga macam tulang dan fungsi yang sama pula. Maka dari itu muncul istilah “*Leaf venation*” atau pola tulang daun. Pola tulang daun dapat dibedakan menjadi lima macam [9] yaitu *pinnate*, *palmate*, *reticulate*, *dichotomous* dan *parallel*.



Gambar 2 Lima pola tulang daun [9]

B. Canny edge detection

Edge detection adalah salah satu metode *image processing* yang digunakan untuk mengambil suatu ciri khusus dari citra. Tujuan utama dari *edge detection* adalah mengurangi jumlah data pada *image*, namun tetap menjaga sifat struktural yang akan digunakan untuk pemrosesan gambar lebih lanjut [10]. Menurut [11] Algoritma *canny edge detection* terdiri dari lima langkah, yaitu *smoothing*, *finding gradients*, *non-maximum suppression*, *double thresholding*, dan *edge tracking by hysteresis*.

C. Jaringan saraf tiruan *backpropagation*

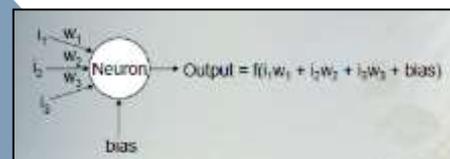
Jaringan saraf tiruan dipahami sebagai sistem pemrosesan informasi yang bekerja berdasarkan prinsip kerja dari sistem saraf biologis. Jaringan saraf tiruan dibentuk untuk memecahkan suatu masalah tertentu seperti pengenalan pola atau klasifikasi karena adanya proses pembelajaran [12].

Backpropagation neural network terdiri atas beberapa proses yang harus dilakukan. Fase pelatihan adalah tahap awal yang digunakan untuk melatih jaringan saraf tiruan. Data diberikan dengan *output* pola yang sudah ditentukan. Jaringan akan menyesuaikan bobot antara *output* dan pola yang telah ditentukan. Setelah mendapatkan bobot optimal dari jaringan dan dianggap telah cukup, maka fase *testing* dapat dimulai. Fase *testing* dilakukan dengan cara memasukkan data yang belum pernah dikenali oleh jaringan namun tetap mempunyai asal yang serupa dengan data *training*.

Berikut adalah parameter yang diperlukan dalam pembuatan jaringan saraf tiruan:

1. Inisialisasi *range* bobot (r): *range* bobot biasanya berkisar $[-r, r]$

2. Banyaknya *node* dalam *hidden layer*: Menentukan jumlah *node* dalam setiap lapisan tersembunyi. Memilih jumlah *node* dapat ditentukan dengan cara *trial and error*.
3. Banyaknya *epoch*: Satu *epoch* adalah satu putaran dalam proses *training* data. Memperbesar banyaknya *epoch* akan meningkatkan akurasi, namun akan memakan waktu yang cukup lama. Begitu juga sebaliknya, jumlah *epoch* yang terlalu sedikit akan mengurangi akurasi namun memiliki waktu pengerjaan yang cepat.
4. Penjumlahan nilai pada *node*: Penjumlahan *node* dari satu *layer* ke *layer* berikutnya digunakan dengan menjumlahkan *value node* dikalikan dengan bobotnya masing-masing lalu dijumlahkan dengan *bias*. *Bias* bernilai satu dan memiliki bobotnya sendiri. Nilai *bias* digunakan untuk menghindari *value* dari *node* agar tidak bernilai nol. Gambar 3 menunjukkan perhitungan *value* pada suatu *node*.



Gambar 3 Penjumlahan *node* [13]

5. Fungsi aktivasi: Dalam *backpropagation*, fungsi aktivasi yang dipakai harus memenuhi beberapa syarat yaitu: kontinu, terdiferensial dengan mudah dan merupakan fungsi yang tidak turun [14] Salah satu yang paling sering dipakai adalah fungsi *sigmoid* yang memiliki *range* (0,1) yang ditunjukkan pada rumus 1.

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

6. Besarnya *learning rate*: Adalah faktor pengali untuk kesalahan dalam perbaikan propagasi balik. Nilai *learning rate* yang rendah, akan menghasilkan pembelajaran yang lambat namun cenderung stabil. Nilai *learning rate* yang besar menghasilkan pembelajaran yang lebih cepat namun fase pembelajaran yang kurang stabil. Nilai *learning rate* berkisar antara 0.1 sampai 0.9.
7. *Critical error*: Merupakan batasan *error* yang diterima. Jika *error* yang keluar sama atau lebih kecil dari batasan *error* maka pelatihan dihentikan. Cara paling umum untuk menghitung *error* adalah menggunakan *Mean Square Error* (MSE).

$$MSE = (\text{target} - \text{output})^2 \quad (2)$$

8. Perhitungan *error*: Dalam fase *backpropagation* perlu dilakukan perhitungan *error* pada tiap *node* dengan menggunakan rumus perhitungan *error*

dengan cara mengkalikan turunan dari fungsi aktivasi dengan target yang diharapkan dikurangi dengan *output* dari *node* seperti yang terlihat pada Rumus 3.

$$\delta_j = f'(net_j)(target_j - output_j) \quad (3)$$

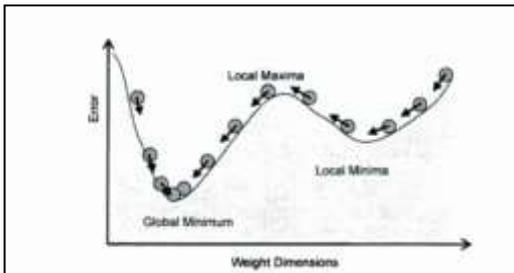
- Perubahan bobot: Perubahan bobot dilakukan setelah *error* diperoleh untuk menghitung bobot yang baru. Nilai perubahan bobot akan diperoleh dengan mengalikan *learning rate*, *error*, dan *value node* seperti terlihat pada rumus 4.

$$\Delta w_{ji} = \alpha \delta_j x_{ji} \quad (4)$$

Setelah diperoleh nilai perubahan bobot, maka akan dihitung bobot baru dengan menambahkan bobot lama dan bobot baru seperti pada Rumus 5.

$$w_{new} = w_{old} + \Delta w_{new} \quad (5)$$

- Momentum*: Merupakan parameter tambahan yang dapat dimasukkan kedalam perhitungan perubahan bobot *backpropagation*. *Momentum* digunakan untuk menghindari terjebak pada *local minimum* dan mengarahkan bergerak kedalam *global minimum* agar memperoleh *value error* yang lebih kecil seperti yang terlihat pada Gambar 4.



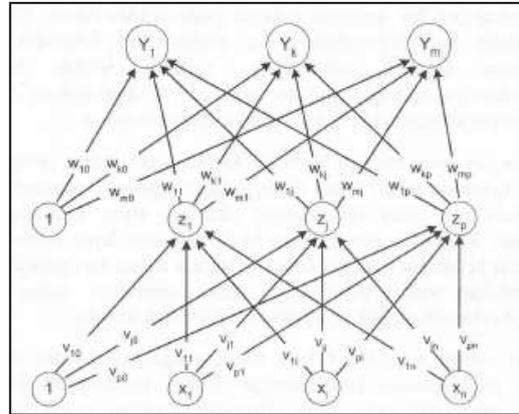
Gambar 4 Momentum [15]

Momentum dapat diterapkan dengan menambahkan perubahan bobot sebelumnya ($n-1$) dikalikan dengan *value momentum* yang telah ditetapkan. Rumus perubahan bobot menggunakan *momentum* dapat dilihat pada Rumus 6.

$$\Delta w_{ji}^n = \alpha \delta_j x_{ji} + \mu \Delta w_{ji}^{(n-1)} \quad (6)$$

Jaringan saraf tiruan memiliki 3 *layer* utama yaitu *input layer*, *hidden layer*, dan *output layer*. Gambar 5 adalah arsitektur dari jaringan saraf tiruan dengan n *input* (ditambah *bias*), *hidden layer* terdiri atas p *node* (ditambah *bias*), serta m *node output*

layer. V_{ji} merupakan bobot dari *input node* X ke unit tersembunyi Z . W_{jk} merupakan bobot dari *hidden layer* ke *output layer*.



Gambar 5 Arsitektur jaringan saraf tiruan [14]

D. Socket programming

Socket merupakan cara melakukan komunikasi antar program dalam suatu jaringan yang sama. Socket digunakan untuk mewakili konektivitas antar *client* dan *server*. Untuk membuat suatu *socket connection* dibutuhkan IP dan port yang digunakan. *Server* yang sedang berjalan menggunakan suatu *port* yang telah ditetapkan. *Server* menunggu *client* untuk membuat *connection*. *Client* yang mengetahui IP dan *port server* akan membuat permintaan koneksi. Setelah koneksi diterima *server* maka *socket* dapat digunakan sebagai media perpindahan data [16].

III. METODE DAN PERANCANGAN SISTEM

A. Metode

Dalam penelitian ini, menggunakan metodologi penelitian sebagai berikut:

- Studi literatur
Pada tahap ini, dilakukan pencarian teori-teori mendasar mengenai pola tulang daun. Selain mencari data untuk objek kajian penelitian, dilakukan pula pencarian berbagai paper dan penelitian mengenai *backpropagation* sebagai program utama yang melakukan *pattern recognition* dan *edge detection* sebagai *image processing* untuk memproses citra warna menjadi greyscale.
- Perancangan sistem
Tahap perancangan sistem mencakup pembuatan DFD untuk menganalisis aliran data, flowchart untuk penentuan alur program, dan perancangan desain antarmuka untuk merancang tampilan yang baik seperti posisi peletakan button, text, dan gambar untuk membuat suatu user interface.
- Pemrograman sistem

Pemrograman aplikasi menggunakan Android Studio sebagai client yang digunakan oleh user untuk mengirim image daun ke server. Software Eclipse Luna digunakan untuk merancang sistem jaringan saraf tiruan backpropagation dengan bahasa pemrograman Java sebagai server di laptop yang menerima masukan image dari client dan memberikan hasil output pola.

4. Pengujian dan survey

Proses pengujian dilakukan dengan menggunakan data daun ketiga pada sistem jaringan saraf tiruan yang sebelumnya sudah dilatih dengan data daun pertama dan kedua. Dihitung akurasi pengenalan pola masing-masing dan dihitung pula akurasi rata-ratanya. Survey dilakukan untuk mengetahui kemudahan dan manfaat dari aplikasi pengenalan pola tulang daun.

B. Perancang Sistem

Sistem dibagi ke dalam dua bagian yaitu sisi *server* dan *client*. Di sisi *client*, *user* dapat memilih gambar dengan menggunakan *camera* dari perangkat *smartphone*. Gambar dikirim ke *server*, *server* akan memproses *image* dengan *edge detection*, lalu melakukan perhitungan jaringan saraf tiruan untuk memperoleh *output* pola yang kemudian akan dikirim kembali ke *client*. *Client* yang menerima *output* pola akan menampilkan informasi *output* pola dengan *dialog alert*.

Program utama jaringan saraf tiruan Backpropagation terdiri atas tiga tahap inialisasi, feedforward, dan backpropagation. Inialisasi adalah bagian awal spesifikasi dalam jaringan saraf tiruan ditentukan. Proses inialisasi mencakup penentuan nilai input node, hidden node, output node, leraning rate dan momentum. Input node dimasukkan sesuai dengan image size yang digunakan, hidden node ditentukan dengan metode trial dan error dan output node adalah hasil yang dikeluarkan oleh jaringan saraf tiruan (dalam percobaan ini digunakan lima output node sesuai dengan jumlah pola yang ada). Digunakan learning rate sebesar 0.3 dan nilai momentum sebesar 0.9 [17].

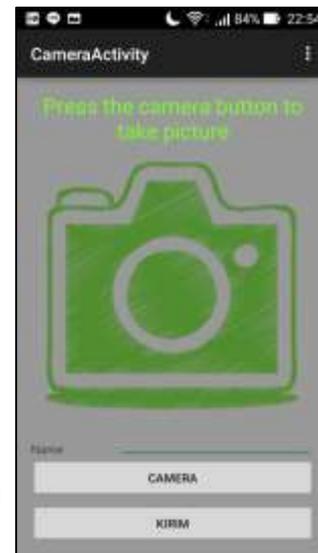
Feedforward adalah proses alur maju yang menghitung perubahan nilai dari input ke hidden layer, kemudian dari hidden layer untuk menunjukkan nilai yang dihasilkan pada output layer

IV. IMPLEMENTASI DAN UJI COBA

Proses uji coba dilakukan dengan melakukan percobaan menggunakan *image size* dan *hidden node* yang berbeda. Metode uji coba berdasarkan *trial and error* untuk mendapatkan suatu bobot yang optimal untuk mengenali pola tulang daun yang ada. Sistem dirancang dengan dua sistem utama sebagai *server* dan *client*. *User* menggunakan perangkat *smartphone*, sehingga *user* dapat secara langsung memfoto objek daun yang kemudian akan dikirim ke *server* komputer

untuk dilakukan proses pengenalan pola dengan menggunakan jaringan saraf tiruan, dan kemudian dikirim kembali hasil polanya ke perangkat *smartphone*.

Implementasi dengan menggunakan program berbasis android ditujukan untuk mempermudah *user* dalam proses pengambilan gambar yang lebih efisien (dapat langsung difoto atau menggunakan *gallery* dari *smartphone*) seperti yang ditunjukkan gambar 7.



Gambar 7 User interface image dari camera

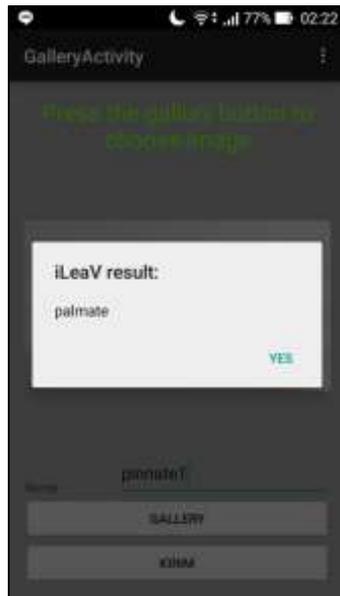
Gambar 8 adalah *user interface* saat *image* daun difoto, maka akan ditampilkan terlebih dahulu ke perangkat *smartphone*.



Gambar 8 User interface tampilan image

Gambar 9 adalah *user interface* saat *image* telah dikirim untuk proses pengenalan pola di server,

setelah hasil diperoleh maka akan ditampilkan dalam bentuk *alert dialog*.



Gambar 9 User interface hasil pola

Pertama-tama sistem diuji dengan logika sederhana (AND dan XOR) untuk membuktikan bahwa sistem jaringan saraf tiruan *backpropagation* dapat berjalan dengan baik. Selanjutnya pada sistem akan dilakukan uji coba dengan data daun yang menjadi fokus penelitian ini. Data yang diambil merupakan lima pola daun yang berbeda sebagai contoh dan sistem dapat dilatih untuk jenis pola pola yang lain. Tiap pola terdiri atas tiga helai daun yang difoto dan disimpan sebagai data. Daun pertama dan daun kedua (2/3) dari tiap pola akan digunakan sebagai *training set*, sedangkan data daun ketiga (1/3) akan digunakan sebagai *testing set* untuk akurasi pengenalan pola tersebut. Berikut adalah hasil rekapitulasi data daun dengan menggunakan berbagai *hidden node* dan *image size*:

Tabel 1 Rekapitulasi akurasi pengenalan pola

Image size	Hidden node	Epoch	rata-rata akurasi
100 X 100	10	5.044	38%
	20	1.939	56%
	50	624	44%
	100	222	60%
200 X 200	20	3.109	74%
	50	1.099	70%
	100	704	76%

Pada tabel, didapat tingkat akurasi tertinggi adalah 76% dan tingkat terendah adalah 38%. Ada beberapa faktor penyebab tingkat akurasi, salah satunya adalah pola daun yang tidak terlihat jelas, sehingga tidak

terbaca sebagai input pada jaringan syaraf tiruan. Tingkat akurasi ini masih memiliki tingkat yang belum maksimal sehingga perlu ditingkatkan dengan memperbaiki image preprocessing, sehingga diharapkan tingkat akurasi bertambah.

V. SIMPULAN

Berdasarkan penelitian yang dilakukan dengan menggunakan *image* daun ketiga dapat disimpulkan bahwa aplikasi pengenalan pola tulang daun dengan metode jaringan syaraf tiruan *backpropagation* telah berhasil dirancang dan dibangun. Sistem memperoleh akurasi rata-rata pengenalan yang paling baik sebesar 76%.

Image size memegang peranan penting dalam pengenalan pola. *Image size* yang terlalu kecil akan mengakibatkan data tidak memperoleh ciri khusus dari daun dan *image* yang terlalu besar akan memperoleh *noise* yang besar sehingga dapat mengganggu proses pengenalan pola. Dapat disimpulkan juga dengan bertambahnya *hidden node* maka nilai *epoch* akan semakin mengecil. Bobot optimal diperoleh dengan menggunakan *image size* 200 x 200, *hidden node* sebanyak 100, *learning rate* sebesar 0.3 dan *momentum* sebesar 0.9. Adapun akurasi masing-masing pola adalah *pinnate* (90%), *palmate* (90%), *reticulate* (100%), *parallel* (0%), dan *dichotomous* (100%).

Akurasi pola tulang daun *parallel* tidak dikenali sistem karena bentuk daun yang dipakai pada *training* dan *testing* tidak berhasil mendapatkan ciri khusus dari pola tulang daun dan adanya kemiripan dari bentuk pola luarnya dengan bentuk pola *pinnate*.

DAFTAR PUSTAKA

- [1] Zheng, Xiaodong & Wang, Xiaojie. (2010). Leaf Vein Extraction Based on Gray-scale Morphology. *International Journal of Image, Graphics and Signal Processing*. 2. 10.5815/ijgisp.2010.02.04.
- [2] Bowo, S. dkk. 2011. Analisis Deteksi Tepi Untuk Mengidentifikasi Pola Daun. skripsi. Universitas Diponegoro. Semarang
- [3] Kelvianto, Kevin. 2014. Rancang bangun aplikasi pengklasifikasian jenis bunga menggunakan jaringan syaraf tiruan metode *backpropagation* berbasis android. Universitas Multimedia Nusantara, Tangerang
- [4] Gustina, Sapriani, 2016 "Identifikasi Tanaman Kamboja Menggunakan Ekstraksi Ciri Citra Daun dan Jaringan Syaraf Tiruan". Annual Research Seminar: Computer Science and Information and Communications Technology_
- [5] Lee, K. B., dan Hong K. S. 2013. An Implementation of Leaf Recognition System using Leaf Vein and Shape. *International Journal of Bio-Science and Bio-Technology* Vol. 5, No. 2, April, 2013.
- [6] Larese, M.G., Namías, R., Craviotto, R.M., Arango, M.R., Gallo, C., & Granitto, P.M. (2014). Automatic classification of legumes using leaf vein image features. *Pattern Recognition*, 47, 158-168.
- [7] Struwe, L. 2016. Field identification of the 50 most common plant families in temperate regions (including agricultural, horticultural, and wild species). Rutgers University, New Brunswick, NJ, USA.

- [8] Warnita, dkk. Tanpa tahun. Bahan Ajar Botani. Tersedia dalam : faperta.unand.ac.id/deposit/BahanAjarBotani.pdf [diakses 17 Maret 2016]
- [9] Geneve, Robert. Tanpa tahun. PLS 220: Intoduction to plant identification. University of Kentucky: Department of Horticulture.
- [10] J. Canny, "A Computational Approach to Edge Detection," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986. doi: 10.1109/TPAMI.1986.4767851
- [11] Moueslund, Thomas B. 2009. Canny edge detection. Tersedia dalam: <http://www.cse.iitd.ernet.in/~pkalra/csl78>
- [12] Ginting, Eji Duanta. 2014. Perancangan Aplikasi Pengenalan Pola Daun Klasifikasi Jenis Tanaman Dengan Pemanfaatan Jaringan Saraf Tiruan Metode Probabilistik. Jurnal STMIK Budi Darma Medan.
- [13] K. Cannons and V. Cheung, 2002 "An Introduction to Neural Networks," Iowa State University, Ames.
- [14] Siang, J.J. 2005. Jaringan saraf tiruan & pemrogramnya dengan menggunakan matlab. Yogyakarta: Penerbit Andi
- [15] Priddy KL, Keller PE. Artificial neural networks: an introduction: SPIE Press 2005.
- [16] Aamir, Zeeshan. 2010. Developing a Communication Link between Agents and Cross Platform IDE.
- [17] Karmakar, S & Shrivastava, G & Kowar, Manoj. (2014). Impact of learning rate and momentum factor in the performance of back-propagation neural network to identify internal dynamics of chaotic motion. *Kuwait Journal of Science*. 41. 151-174.

