

Klasifikasi Daun Dengan Perbaikan Fitur Citra Menggunakan Metode *K-Nearest Neighbor*

Febri Liantoni

Jurusan Teknik Informatika, Institut Teknologi Adhi Tama, Surabaya, Indonesia
febri.liantoni@gmail.com

Diterima 18 September 2015

Disetujui 23 Desember 2015

Abstract—Plants are the most important part in life on earth as oxygen supplier to breathe, groceries, fuel, medicine and more. Plants can be classified based on its leaves shape. Classification process is required well data extraction feature, so it needs fixing feature process at pre-processing level. Combining median filter and image erosion is used for fixing feature process. Whereas for feature extraction is used invariant moment method. In this research, it is used leaves classification based on leaves edge shape. K-Nearest Neighbor Method (KNN) is used for leaves classification process. KNN method is chosen because this method is known rapid in training data, effective for large training data, simple and easy to learn. Testing the result of leaves classification from image which is on dataset has been built to get accuracy value about 86,67%.

Index Terms—Classification, Median Filter, Invariant Moment, K-Nearest Neighbor.

I. Pendahuluan

Ilmu tentang tumbuhan mengalami kemajuan yang pesat, bidang pengetahuan yang sebelumnya hanya merupakan cabang ilmu tumbuhan saja, sekarang telah menjadi bidang ilmu yang berdiri sendiri. Salah satunya adalah Morfologi Tumbuhan yang mempelajari bentuk dan susunan tubuh tumbuhan. Bentuk tepi daun bias digunakan untuk acuan klasifikasi daun. Tumbuhan berguna sebagai penyedia oksigen untuk bernafas, sebagai bahan makanan, bahan bakar, obat-obatan, kosmetik dan lebih banyak lagi. Proses klasifikasi tumbuhan dapat dilakukan dengan cara mengidentifikasi citra bentuk daun dari tumbuhan itu sendiri. Cara pengambilan

gambar daun dari tumbuhan tersebut, maka dapat dilakukan langkah-langkah pengenalan pola daun dengan cara mengenali karakteristik struktural daun seperti bentuk dan tekstur daun tersebut. [1], [2].

Proses klasifikasi tumbuhan dapat dilakukan dengan cara mengidentifikasi gambar bentuk daun dari tumbuhan. Dengan cara tersebut maka dapat dilakukan langkah-langkah pengenalan pola daun dengan mengenali karakteristik struktural daun seperti bentuk dan tekstur sebuah daun. Metode untuk melakukan pemrosesan terhadap citra masukan dengan pemanfaatan teknik pengolahan citra digital dilakukan untuk menganalisa karakteristik struktural daun.

Perkembangan teknologi untuk teknik pengolahan citra juga berkembang pesat. Berbagai teknik dikembangkan untuk mempermudah pekerjaan manusia, baik sebagai pengolah citra, analis citra maupun pengguna citra untuk berbagai tujuan dan keperluan. Seringkali citra yang digunakan tidak dalam kondisi yang ideal untuk dikaji dikarenakan banyaknya gangguan, dapat berupa bayangan, foto atau gambar kabur, kurang jelasnya kenampakan obyek sehingga dapat menimbulkan masalah dan mempengaruhi hasil interpolasi serta akan mempengaruhi analisa dan perencanaan yang akan dilakukan, maka diperlukan berbagai teknik pengolahan citra untuk memperoleh citra yang ideal.

Teknik pengolahan citra digital ini dilakukan pada tahapan pra-proses citra sampai didapatkan bentuk tepian dan ciri struktural dari masing-masing daun. Metode yang digunakan pada ekstraksi fitur ini adalah pengenalan fitur morfologi digital [3]. Proses ekstraksi fitur

yang digunakan menggunakan metode *Moment Invariant*. Setelah itu dilakukan ekstraksi fitur dari citra daun tersebut sehingga didapatkan informasi struktural daun yang kemudian digunakan sebagai data pengelompokkan.

Sistem pengenalan dan pengklasifikasian daun tumbuhan yang otomatis sangat berguna karena dapat mendukung pengklasifikasian tumbuhan dengan cepat. Daun merupakan salah satu ciri tumbuhan yang unik dan mudah diamati dan cukup representatif sehingga bisa dijadikan obyek untuk ekstraksi fitur tumbuhan. Ekstraksi fitur obyek yang tepat sangat mempengaruhi baik buruknya hasil klasifikasi tumbuhan. Beberapa penelitian tentang pengelompokkan bentuk tumbuhan, umumnya menggunakan metode jaringan saraf tiruan seperti yang dilakukan Stephen dkk tahun 2011, Kadir dkk tahun 2011, Husin dkk tahun 2012, Chaki dkk tahun 2011 [1],[2],[3],[4]. Metode jaringan saraf tiruan banyak digunakan karena metode ini dikenal lebih cepat secara substansial. Akan tetapi penentuan jumlah hidden layer yang digunakan berpengaruh pada hasil, serta dibutuhkan parameter jumlah epoch yang besar sehingga membutuhkan komputasi yang lebih tinggi. Pada tahun 2012, Arunpriya melakukan penelitian pengenalan tumbuhan menggunakan algoritma *Support Vector Machine* (SVM) [5]. Metode SVM yang bersifat *linear classifier* dan secara teoritik hanya dikembangkan untuk permasalahan dua kelas[6].

Pada penelitian ini klasifikasi akan dilakukan untuk citra daun dengan metode *K-Nearest Neighbor* (KNN). Proses klasifikasi berdasarkan fitur bentuk tepi daun. Metode KNN melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang objek terdekat. Sebelum tahapan klasifikasi terlebih dahulu dilakukan tahapan pra proses citra dan ekstraksi fitur citra tepi daun agar didapatkan nilai masukan yang tepat untuk tahapan klasifikasi spesies daun berdasarkan citra daun.

II. Metode

A. Ekstraksi Fitur Invariant Moment

Ekstraksi fitur yang digunakan adalah *moment invariant*. Proses ini dilakukan untuk menghasilkan nilai-nilai fitur berupa vektor dari citra biner. Fitur yang digunakan yaitu seven moment invariant yang akan menghasilkan tujuh nilai pada vektor fitur.

Proses pengenalan sebuah obyek di dalam sebuah citra setelah proses segmentasi, sering terbentur pada permasalahan posisi obyek, rotasi sumbu obyek, dan perubahan skala dari obyek. Posisi obyek yang bergeser atau berputar maupun ukurannya yang lebih kecil atau lebih besar daripada dapat menyebabkan kesalahan dalam pengenalan atau identifikasi obyek.

Pada penggunaan perhitungan nilai 2 (dua) dimensi momen sample gambar $M \times M$ dari fungsi kontinu $f(x,y), (x,y=0, \dots, M-1)$ didapatkan Persamaan 1.

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{M-1} (x)^p \cdot (y)^q \cdot f(x,y) \quad (1)$$

Momen dapat menggambarkan suatu obyek dalam hal area, posisi, orientasi dan parameter terdefinisi lainnya. Dengan mendapatkan sejumlah informasi momen, baik momen tingkat ke nol (m_{00}) dan ke satu (m_{10}) dan (m_{01}) atau momen sentral, dan momen pada tingkat ≥ 2 atau moment invariant dari sebuah obyek, maka obyek tersebut dapat diidentifikasi sekalipun telah mengami pergeseran (translasi), perputaran (rotasi) maupun perubahan skala.

Dari moment $f(x,y)$ akan ditranslasikan dengan nilai (a,b) sehingga didapatkan perhitungan baru seperti Persamaan 2.

$$\mu_{pq} = \sum_x \sum_y (x+a)^p \cdot (y+b)^q \cdot f(x,y) \quad (2)$$

Dari central moment utama yaitu m_{pq} atau μ_{pq} dikomputasi melalui proses substitusi terhadap nilai $a = -x$ dan nilai $b = -y$ maka akan didapatkan perhitungan pada Persamaan 3.

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p \cdot (y - \bar{y})^q \cdot f(x,y) \quad (3)$$

$$\text{Untuk nilai } \bar{x} = \frac{m_{10}}{m_{00}} \text{ dan } \bar{y} = \frac{m_{01}}{m_{00}}$$

Ketika proses normalisasi maka nilai penskalaan yang digunakan dalam perhitungan berubah, ditunjukkan pada Persamaan 4.

$$\eta_{pq} = \mu_{pq} / \mu_{pq}^y \quad (4)$$

Untuk nilai $\gamma = [(p + q)/2] + 1$

Kemudian dari proses tersebut maka didapatkan nilai *seven moment invariant* dengan perhitungan pada Persamaan 5.

$$\begin{aligned} Hu_1 &= \eta_{20} + \eta_{02} \\ Hu_2 &= (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2 \\ Hu_3 &= (\eta_{20} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2 \\ Hu_4 &= (\eta_{20} + \eta_{12})^2 + (3\eta_{21} + \eta_{03})^2 \\ Hu_5 &= (\eta_{20} + 3\eta_{12})(\eta_{30} + \eta_{12})(\eta_{20} + \eta_{12})^2 - 3(\eta_{12} + \eta_{03})^2 + (3\eta_{21} + \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \\ Hu_6 &= (\eta_{20} - \eta_{02})(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \\ Hu_7 &= (3\eta_{21} - \eta_{30})(\eta_{30} + \eta_{12})(\eta_{20} + \eta_{12})^2 - 3(\eta_{12} + \eta_{03})^2 + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2] \end{aligned} \quad (5)$$

B. K-Nearest Neighbor (KNN)

Algoritma K-Nearest Neighbor (KNN) adalah sebuah metode untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Data pembelajaran diproyeksikan ke ruang berdimensi banyak, dimana masing – masing dimensi merepresentasikan fitur dari data.

Algoritma KNN termasuk metode yang menggunakan algoritma *supervised* [7][8][9]. Perbedaan antara *supervised learning* dengan *unsupervised learning* adalah pada *supervised learning* bertujuan untuk menemukan pola baru dalam data dengan menghubungkan pola data yang sudah ada dengan data yang baru. Sedangkan pada *unsupervised learning*, data belum memiliki pola apapun, dan tujuan *unsupervised learning* untuk menemukan pola dalam sebuah data [7][8][9]. Tujuan dari algoritma KNN adalah untuk mengklasifikasi objek baru berdasarkan atribut dan training samples [8][9]. Dimana hasil dari sampel uji yang baru diklasifikasikan

berdasarkan mayoritas dari kategori pada KNN. Pada proses pengklasifikasian, algoritma ini tidak menggunakan model apapun untuk dicocokkan dan hanya berdasarkan pada memori.

Algoritma KNN menggunakan klasifikasi ketetangaan sebagai nilai prediksi dari contoh data uji yang baru [7][8][9]. Jarak yang digunakan adalah jarak *Euclidean Distance*. Jarak Euclidean adalah jarak yang paling umum digunakan pada data numeric. Algoritma KNN merupakan algoritma yang menentukan nilai jarak pada pengujian data testing dengan data training berdasarkan nilai terkecil dari nilai ketetangaan terdekat [10].

III. Perancangan sistem

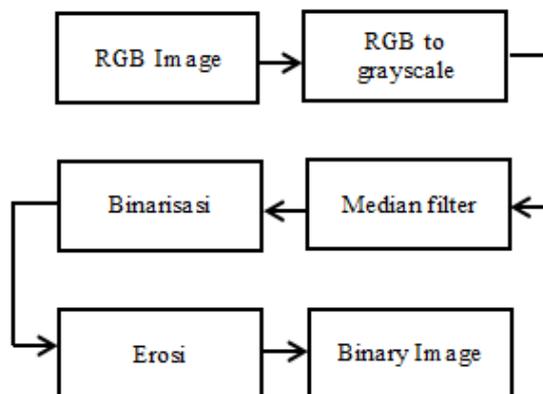
Pada penelitian ini data yang digunakan berupa gambar daun dengan latar belakang berwarna putih. Dataset yang digunakan yaitu berupa citra daun hijau yang terdiri atas 5 jenis (5 kelas). Data citra yang digunakan sebanyak 50 foto citra. Sistem klasifikasi citra daun menggunakan metode *K-Nearest Neighbor* terdiri dari beberapa proses meliputi praproses, ekstraksi fitur dan tahap klasifikasi. Contoh data yang digunakan seperti ditunjukkan pada Gambar 1.



Gambar 1. Contoh data daun

A. Praproses

Praproses dilakukan dengan tujuan untuk mengolah data masukan sehingga dapat digunakan untuk proses ekstraksi fitur. Terdapat beberapa langkah yang dilakukan untuk praproses yaitu perubahan citra pada ruang warna RGB ke ruang warna grayscale, melakukan median filter, pembuatan citra biner, dan melakukan perbaikan beberapa piksel gambar dengan metode erosi citra. Dari tahapan praproses akan didapatkan citra biner yang akan digunakan pada ekstraksi fitur. Tahapan yang dilakukan pada praproses ditunjukkan pada Gambar 2.



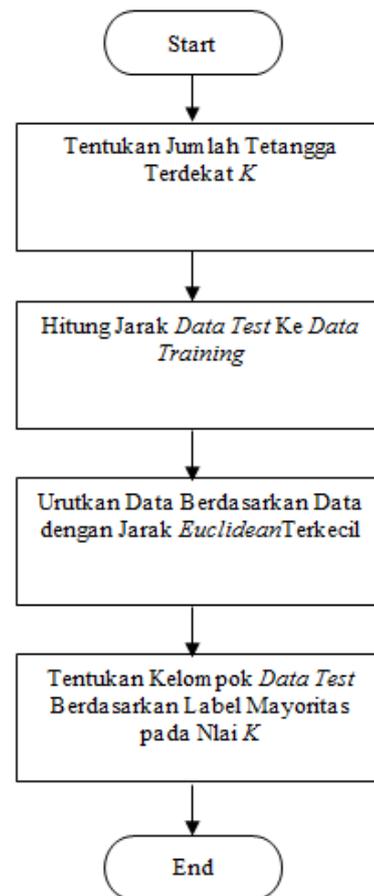
Gambar 2. Tahapan praproses

Tahap awal praproses dengan mengubah citra ke dalam *grayscale*. Pemrosesan ini dilakukan untuk mengubah domain piksel citra menjadi 8 bit skala abu-abu. Untuk konversi tersebut digunakan seperti ditunjukkan pada Persamaan 6.

$$gray = \frac{red \cdot 299 + green \cdot 587 + blue \cdot 114}{1000} \quad (6)$$

B. Klasifikasi *K-Nearest Neighbor (KNN)*.

Pada penelitian ini digunakan metode *K-Nearest Neighbor (KNN)*. Metode KNN dilakukan dengan membandingkan data uji dengan data *training*. Salah satu tahap awal yang harus dilakukan adalah menentukan jumlah nilai *K*. Tahapan algoritma KNN yang dilakukan ditunjukkan pada Gambar 3.



Gambar 3. Tahapan algoritma KNN

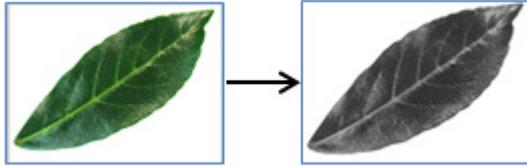
IV. Hasil dan pembahasan

Pengujian dilakukan pada tahap praproses dan tahap klasifikasi. Hasil dari praproses digunakan sebagai data pada metode klasifikasi. Pengujian ini bertujuan untuk mengetahui keberhasilan sistem dalam klasifikasi daun. Skenario uji coba yang dilakukan dengan cara data dipisahkan menjadi dua bagian yaitu 70% digunakan sebagai data latih dan 30% digunakan sebagai data uji. Data latih digunakan untuk ekstraksi fitur sedangkan data uji coba digunakan untuk menguji ketepatan sistem dalam melakukan klasifikasi daun. Hasil klasifikasi tersebut akan dicatat dan dibandingkan dengan klasifikasi yang sebenarnya. Dari hasil klasifikasi kemudian dihitung akurasi.

A. Hasil Praproses

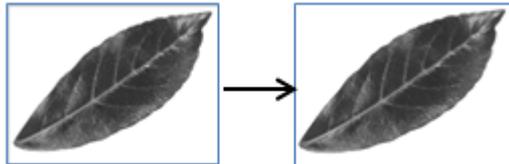
Pada praproses dilakukan pengolahan gambar untuk beberapa tahapan. Meliputi proses

grayscale, median filter, binarisasi dan erosi citra. Contoh citra hasil konversi dari RGB ke citra grayscale seperti ditunjukkan pada Gambar 4.



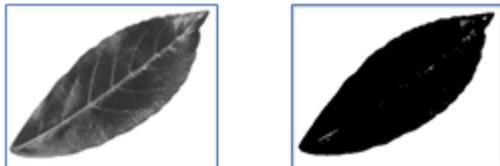
Gambar 4. Konversi RGB ke Grayscale

Citra grayscale hasil konversi kemudian dilakukan proses median filter dengan tujuan untuk menghilangkan noise dari citra. Contoh citra hasil perbaikan dengan median filter ditunjukkan pada Gambar 5.



Gambar 5. Perbaikan dengan Median Filter

Proses binarisasi citra grayscale dikonversi ke citra hitam putih. Citra hasil binarisasi ditunjukkan pada Gambar 6.



Gambar 6. Proses binarisasi citra

Tahapan selanjutnya dilakukan proses erosi. Proses ini bertujuan untuk memperbaiki citra hasil dari proses binarisasi. Proses ini dilakukan

dengan cara menghilangkan struktur daun sehingga didapatkan obyek yang jelas.. Citra hasil perbaikan dengan proses erosi ditunjukkan pada Gambar 7.



Gambar 7. Proses erosi citra

Hasil proses erosi citra dilakukan operasi reverse untuk mendapatkan citra dengan warna putih dengan latar belakang hitam. Secara keseluruhan praproses yang dilakukan ditunjukkan pada Gambar 8. Setelah tahapan praproses selesai kemudian dilakukan ekstraksi fitur daun menggunakan metode invariant moment.



Gambar 8. Hasil praproses

B. Hasil Ekstraksi Fitur

Tahap ekstraksi fitur telah berhasil menghasilkan vektor fitur yang berisi tujuh nilai dari invariant moment. Contoh nilai pada ekstraksi fitur yang dihasilkan oleh sistem ditunjukkan pada Tabel 1.

Tabel 1. Hasil data ekstraksi

Data Daun	Moment Invariant						
	hu1	hu2	hu3	hu4	hu5	hu6	hu7
daun1	636,2875715	377,5476012	9,558924343	8,495767867	0,076561086	5,214553338	-0,0001017151
daun2	607,7752928	342,1171992	9,475688886	8,199628769	0,072273792	4,788270122	-0,0006172914
daun3	463,357561	186,4047613	9,138979028	7,064831623	0,056762516	3,046526299	-0,0007697243
daun4	225,9960207	25,25512555	0,133270072	0,038015113	0,000002688	0,0058611	0,0000003073
daun5	223,6928708	24,32089636	0,105783019	0,034432638	0,000002077	0,005365624	0,0000000595
daun6	230,1434446	26,81762805	0,382964436	0,129518085	0,000028842	0,021187938	-0,0000004064
daun7	163,3980718	0,060128297	0,256191543	0,000221773	-0,000000000	-0,0000001880	0,0000000006

C. Hasil Klasifikasi

Pengujian klasifikasi dilakukan terhadap 15 data uji. Data uji akan diklasifikasikan terhadap data *training*. Proses klasifikasi yang dilakukan menggunakan metode *K-Nearest Neighbor* (KNN). Hasil pengujian klasifikasi daun berdasarkan nilai *moment invariant* ditunjukkan pada Tabel 2.

Tabel 2. Hasil pengujian klasifikasi dengan menggunakan metode *K-Nearest Neighbor* (KNN)

Daun	Kelas		Hasil	
	Sebenarnya	Sistem	Benar	Salah
daun1	1	1	1	0
daun2	1	1	1	0
daun3	1	1	1	0
daun4	2	4	0	1
daun5	2	2	1	0
daun6	2	2	1	0
daun7	3	3	1	0
daun8	3	3	1	0
daun9	3	3	1	0
daun10	4	4	1	0
daun11	4	4	1	0
daun12	4	2	0	1
daun13	5	5	1	0
daun14	5	5	1	0
daun15	5	5	1	0

Berdasarkan hasil pengujian terhadap 15 percobaan didapatkan 13 obyek yang terklasifikasi dengan benar dan 2 obyek yang salah terklasifikasi. Data yang tidak terklasifikasi dengan benar yaitu pada daun4 dan daun12. Dari hasil pengujian ini maka didapatkan akurasi sistem sebesar $13/15 = 86,67\%$.

V. Simpulan

Dari hasil pengujian yang telah dilakukan meliputi praproses, ekstraksi fitur dan klasifikasi didapatkan kesimpulan.

1) Penggunaan median filter dan erosi citra mampu memperbaiki citra daun, sehingga

dapat mempermudah proses ekstraksi fitur dan klasifikasi.

2) Pengujian klasifikasi dengan menggunakan metode *K-Nearest Neighbor* (KNN) didapatkan nilai akurasi sebesar 86,67%, hasil ini menunjukkan metode KNN mampu melakukan klasifikasi daun dengan baik.

SARAN

Saran penelitian berikutnya adalah perbaikan fitur citra untuk mendeteksi sidik jari, tulisan, tanda tangan atau wajah seseorang. Selain itu juga bisa dikembangkan ke arah *Optical Character Recognition* (OCR). Perbaikan fitur bisa dilakukan pada proses pembentukan segmen karakter yang dibedakan menjadi garis, kurva atau *loop*. OCR dapat digunakan untuk mengenali tulisan teks cetak, sehingga jika terjadi kehilangan data dapat mengembalikan dokumen yang hilang dengan mengenali dokumen secara otomatis.

DAFTAR PUSTAKA

- [1] Stephen G. W., Forrest S. B., Eric Y. Xu, Yu-Xuan W., Yi-F. C. and Qiao-Liang X., "A Leaf Recognition Algorithm for Plant Classification Using Probabilistic Neural Network", IEEE International Symposium, pp 11-16, July, 2007..
- [2] A. Kadir, Lukito E. N, Adhi N, "Leaf Classification Using Shape, Color, and Texture Features", International Journal of Computer Trends and Technology, July to Aug, 2011.
- [3] Z. Husin, A. Y. M. Shakaff, A. H. A. Aziz, R. S. M. Farook, M. N. Jaafar, U. Hashim, A. Harun, "Embedded Portable Device For Herb Leaves Recognition Using Image Processing Techniques And Neural Network Algorithm", Science Direct on Computers and Electronics in Agriculture, hal 18-29, 2012.
- [4] Chaki J, Parekh R, "Plant Leaf Recognition using Shape based Features and Neural Network classifiers", International Journal of Advanced Computer Science and Applications, vol 2, no 10, 2011.
- [5] ArunPriya C, Balasaravanan T, "An Efficient

Leaf Recognition Algorithm for Plant Classification Using Support Vector Machine”, Proceedings of the International Conference on Pattern Recognition, Informatics and Medical Engineering, hal 21-23, Maret, 2012.

- [6] Theodoridis, Sergios. and Koutroumbas, Kontantinos., “Pattern Recognition”, 2nd Edition, New York, USA: Academic Press, 2003
- [7] Wu X, Kumar V., “The Top Ten Algorithms in Data Mining”, New York: CRC Press; 2009.
- [8] Larose D., “Discovering Knowledge in Data”, USA: John Wiley’s and Son ; 2005.
- [9] Han J and Kamber M., “Data Mining: Concept and Techniques”. New York: Morgan Kaufmann Publisher ; 2006.
- [10] Goujon G, Chaoqun, Jianhong W., “Data Clustering : Theory, Algorithms, and Applications”, Virginia: ASA; 2007.