

IDENTIFIKASI JENIS PENYAKIT PADA KAKAO DENGAN PENGOLAHAN CITRA DIGITAL DAN K-NEAREST NEIGHBOR

CACAO DISEASE IDENTIFICATION USING DIGITAL IMAGE PROCESSING AND K-NEAREST NEIGHBOR

^[1]Gede Hari Yogiswara ^[2]Rita Magdalena, IR., M.T. ^[3]Hilman Fauzi T.S.P., S.T., M.T.

^{1,2,3}Prodi S1 Teknik Telekomunikasi, Fakultas Teknik Elektro, Universitas Telkom

^[1]hariyogiswara@gmail.com ^[2]ritamagdalena@telkomuniversity.ac.id

^[3]hilmanfauzitsp@telkomuniversity.ac.id

Abstrak

Indonesia merupakan negara penghasil dan pengekspor kakao terbesar ke-3 dunia setelah Pantai Gading dan Ghana. Kualitas biji kakao dipengaruhi baik atau buruk dari buahnya itu sendiri. Salah satu parameter dari kualitas buah kakao ada tidaknya penyakit yang dihasilkan oleh hama ataupun patogen yang menyerang tanaman kakao. Namun pengecekan penyakit pada kakao masih dilakukan manual oleh manusia dengan tingkat ketelitian yang relatif masih rendah.

Agar proses pengecekan penyakit pada kakao bisa lebih efisien, maka diperlukan suatu sistem yang dapat melakukan identifikasi penyakit secara otomatis. Pada tugas akhir ini penulis akan merancang sistem identifikasi jenis penyakit kakao otomatis berbasis pengolahan citra digital. Cara kerja dari sistem ini adalah dengan membandingkan sampel kakao yang akan diteliti dengan referensi yang ada pada database. Pada penelitian ini hanya mengidentifikasi buah normal dan buah yang terserang busuk buah dan helopeltis. Sistem yang dirancang pada tugas akhir ini menggunakan metode *Principal Component Analysis* (PCA), klasifikasi *K-Nearest Neighbor* (KNN) dan diimplementasikan menggunakan aplikasi Matlab R2009a.

Berdasarkan hasil pengujian secara keseluruhan, dapat disimpulkan bahwa sistem dapat mengidentifikasi jenis penyakit pada kakao. Hasil akurasi tertinggi sebesar 86,67% diperoleh dengan kombinasi dari 3 parameter yaitu, menggunakan ukuran citra normalisasi 128 x 64 piksel, tanpa membatasi jumlah PC yang digunakan dan dengan metode *distance cityblock* dimana nilai $k = 1$. Rata-rata waktu komputasi yang dibutuhkan untuk proses *pre-processing* adalah 0,31284 detik, sedangkan rata-rata waktu komputasi yang dibutuhkan untuk proses ekstraksi ciri adalah 0,03598 detik dan rata-rata waktu yang dibutuhkan untuk proses klasifikasi adalah 0,07774 detik.

Kata kunci: Buah kakao, Identifikasi jenis penyakit kakao, PCA, k-NN, MATLAB.

Abstract

Indonesia is the 3rd largest country producer and exporter of cocoa in the world after Côte d'Ivoire and Ghana. The quality of cocoa beans are influenced from the fruit itself. One of the parameters of the quality of cocoa pods presence or absence of disease generated by pests or pathogens that attack cocoa plants. But checking the disease in cocoa still done manually by humans with accuracy level is still relatively low.

In order for the process of checking the disease in cocoa can be more efficient, we need a system that can automatically identify the disease. In this final task the author will design a system to identification of diseases of cocoa based automatic digital image processing. The workings of this system is to compare samples of cacao to be examined with reference to the existing database. In this study only identifies normal fruit and the type of black pod disease and helopeltis. The design of the system in this final task is using *Principal Component Analysis* (PCA) methode, *K-Nearest Neighbor* (KNN) classification and implemented using Matlab R2009a applications.

Based on the whole test result, it can be concluded that the system can identify the type of disease in cocoa. Results of the highest accuracy is 86.67% was obtained with a combination of three parameters that is, using the normalization image 128 x 64 pixels, without limiting the number of PCs that are used and the method of distance Cityblock where the value of $k = 1$. The average computation time needed for the pre-processing is 0.31284 seconds, while the average computation time required for the process of feature extraction is 0.03598 seconds and the average time required for classification process is 0.07774 seconds.

Keywords: Cocoa fruit, Cocoa Disease Identification type, PCA, k-NN, MATLAB.

1. Pendahuluan

Indonesia merupakan negara penghasil kakao (*Theobroma cacao*, L.) terbesar nomor tiga di dunia, setelah Pantai Gading dan Ghana [1]. Masalah yang paling sering dialami oleh petani kakao adanya penyakit-penyakit

yang menyerang tanaman kakao. Hal ini dapat mengakibatkan menurunnya kualitas kakao yang dihasilkan. Besarnya kerugian sangat berbeda antarkebun, bervariasi antara 26 % dan 50 % [4].

Selama ini, pengecekan pada tanaman kakao masih dilakukan secara manual dengan hanya mengandalkan tenaga ahli. Cara ini tentunya membutuhkan waktu yang lama mengingat luasnya lahan perkebunan kakao yang harus dievaluasi.

Pada tugas akhir ini, penulis ingin menciptakan suatu sistem identifikasi jenis penyakit pada kakao otomatis. Metode ekstraksi ciri akan dilakukan dengan menggunakan metode *Principal Component Analysis* (PCA) sebagaimana pada paper Deteksi Cacat Daun Teh *Camellia Sinensis* dengan Pengolahan Citra Digital dan JST Learning Vector Quantization [6]. Pada paper tersebut didapatkan akurasi sebesar 85,38% dan waktu komputasi sebesar 4,477 detik. Metode klasifikasi akan menggunakan k-NN sebagaimana pada paper Klasifikasi Buah Belimbing Berdasarkan Citra Red-Green-Blue Menggunakan KNN dan LDA [9]. Pada paper tersebut didapatkan akurasi sebesar 91%. Dengan melihat hasil kedua paper tersebut maka penulis merancang sistem identifikasi jenis penyakit pada kakao menggunakan prinsip pengolahan citra digital metode *Principal Component Analysis* (PCA) dan metode klasifikasi *k-Nearest Neighbor* (k-NN).

2. Dasar Teori dan Perancangan

2.1 Kakao

Kakao merupakan tumbuhan berwujud pohon dimana biji dari tumbuhan ini digunakan sebagai produk olahan yang dikenal sebagai coklat. Habitat asli tanaman kakao adalah hutan tropis dengan naungan pohon-pohon yang tinggi, curah hujan tinggi, suhu sepanjang tahun relatif sama, serta kelembaban tinggi yang relatif tetap. Warna buah kakao sangat beragam, tetapi pada dasarnya hanya ada dua macam warna. Buah yang ketika muda berwarna hijau atau hijau agak putih jika sudah masak akan berwarna kuning. Sementara itu, buah yang ketika muda berwarna merah, setelah masak berwarna jingga [5].

2.1.1 Busuk Buah

Penyakit ini disebabkan oleh jamur *P. palmivora* yang dapat menyerang buah muda sampai masak. Buah yang terserang nampak bercak bercak coklat kehitaman, biasanya dimulai dari pangkal, tengah atau ujung buah. Apabila keadaan kebun lembab, maka bercak tersebut akan meluas dengan cepat ke seluruh permukaan buah, sehingga menjadi busuk, kehitaman dan apabila ditekan dengan jari terasa lembek dan basah. Penyebaran penyakit dibantu oleh keadaan lingkungan yang lembab terutama pada musim hujan. Buah yang membusuk pada pohon juga mendorong terjadinya infeksi pada buah lain dan menjalar kebagian batang/cabang. Patogen ini disebarkan oleh angin dan air hujan melalui spora. Pada saat tidak ada buah, jamur dapat bertahan di dalam tanah. Penyakit ini akan berkembang dengan cepat pada daerah yang mempunyai curah hujan tinggi, kelembaban udara dan tanah yang tinggi terutama pada pertanaman kakao dengan tajuk rapat [2].



Gambar 1. Gejala Busuk Buah

2.1.2 Helopeltis

Helopeltis merupakan penyakit buah kakao yang disebabkan oleh serangan kepik *Helopeltis* spp. yang menyerang buah kakao dan pucuk/ranting muda. Serangan pada buah tua tidak terlalu merugikan, tetapi sebaliknya pada buah muda. Selain kakao, hama ini juga memakan banyak tanaman lain, diantaranya: teh, jambu biji, jambu mete, lamtoro, apokat, mangga, dadap, ubi jalar, dll. Buah muda yang terserang mengering lalu rontok, tetapi jika tumbuh terus, permukaan kulit buah retak dan terjadi perubahan bentuk. Serangan pada buah tua, tampak penuh bercak-bercak cekung berwarna coklat kehitaman, kulitnya mengeras dan retak. Serangan pada pucuk atau ranting menyebabkan pucuk layu dan mati, ranting mengering dan meranggas [2].



Gambar 2. Buah yang Terserang *Helopeltis*

2.2 Citra Digital

Citra digital merupakan citra yang telah direpresentasikan secara numerik yang awalnya berupa fungsi malar (kontinyu) menjadi nilai – nilai diskrit melalui proses sampling (digitalisasi). Pada umumnya citra digital berbentuk persegi panjang, dan dimensi ukurannya dinyatakan sebagai tinggi x lebar (atau lebar x panjang). Persilangan antara baris dan kolom tertentu pada citra digital disebut dengan piksel [3].

2.2.1 Citra RGB

Warna – warna yang diterima oleh mata manusia merupakan hasil dari kombinasi cahaya dengan panjang gelombang yang berbeda – beda. Penelitian memperlihatkan bahwa kombinasi warna yang memberikan rentang warna yang paling lebar adalah red (R), green (G), dan blue (B). Ketiga warna tersebut dinamakan warna pokok (primaries), dan sering disebut sebagai warna dasar RGB [3]. Dari pencampuran warna pokok dengan perbandingan yang berbeda – beda akan dapat diperoleh kombinasi warna – warna lain. Citra RGB disimpan dalam Matlab dengan array berukuran $m \times n \times 3$ yang mendefinisikan warna merah, hijau, dan biru untuk setiap pixelnya. Warna pada setiap pixel ditentukan dari kombinasi merah, hijau, dan biru [10].

2.3 K-Nearest Neighbor

Prinsip kerja *K-Nearest Neighbor (K-NN)* adalah mencari jarak terdekat antara data yang dievaluasi dengan K tetangga (neighbor) terdekatnya dalam data pelatihan. Mirip dengan teknik klastering, mengelompokkan suatu data baru berdasarkan jarak data baru itu ke beberapa data/tetangga terdekat [7].

- Euclidean distance* adalah matrik yang paling sering digunakan untuk menghitung kesamaan dua vektor. *Euclidean Distance* menghitung akar dari kuadrat perbedaan dua vector [7].

$$\text{Euclidean Distance} = \sqrt{\sum_{i=1}^p (x_i - y_i)^2} \quad (1)$$

- Cityblock distance* juga disebut sebagai *Manhattan distance/ Boxcar distance/ Absolute value distance*. *Cityblock distance* menghitung nilai perbedaan absolut dari dua vector [7].

$$\text{Cityblock distance} = \sum_{i=1}^p |x_i - y_i| \quad (2)$$

- Dalam *Cosine distance*, titik-titik dianggap sebagai vektor, dan dilakukan pengukuran terhadap sudut antara dua vektor tersebut. Untuk memperoleh jarak dua vektor x_s dan x_t , memakai rumus sebagai berikut [7].

$$\text{Cosine distance} = 1 - \frac{x_s \cdot x_t}{|x_s| |x_t|} \quad (3)$$

- Dalam *Correlation distance*, titik-titik dianggap sebagai barisan nilai, jarak antar nilai x_s dan x_t memakai rumus sebagai berikut [7].

$$\text{Correlation distance} = 1 - \frac{(x_s - \bar{x}_s)(x_t - \bar{x}_t)'}{\sqrt{(x_s - \bar{x}_s)(x_s - \bar{x}_s)'} \sqrt{(x_t - \bar{x}_t)(x_t - \bar{x}_t)'}} \quad (4)$$

2.4 Gambaran Umum Sistem

Secara garis besar, rancang bangun sistem ini terdiri dari beberapa proses, yaitu:

1. Akusisi citra
2. *Preprocessing*
3. Ekstraksi ciri PCA
4. Klasifikasi
5. Perhitungan akurasi dan waktu komputasi sistem

2.5 Akusisi Citra

Akusisi citra merupakan proses pengambilan citra yang akan digunakan pada sistem. Pada perancangan sistem ini dilakukan pengambilan citra dengan menggunakan *smartphone* yang selanjutnya disimpan dalam format *.jpg dengan ukuran 768 x 1024 pixel. Citra yang telah diambil akan dijadikan sebagai data uji dan data latih untuk melakukan identifikasi jenis penyakit pada kakao. Total jumlah citra yang digunakan pada proses latih adalah sebanyak 90 citra, terdiri dari 30 citra buah kakao kondisi normal, 30 citra buah kakao kondisi busuk buah dan 30 citra buah kakao kondisi helopeltis. Sedangkan total jumlah citra yang digunakan pada proses uji adalah sebanyak 30 citra, terdiri dari 10 citra buah kakao kondisi normal, 10 citra buah kakao kondisi busuk buah dan 10 citra buah kakao kondisi helopeltis.

2.6 Preprocessing

Proses *per-processing* merupakan tahapan sebelum dilakukan pemrosesan selanjutnya terhadap citra digital. Tujuan dari *pre-processing* adalah meningkatkan kualitas dari data masukan serta menghilangkan informasi yang tidak diperlukan dalam proses selanjutnya yaitu ekstraksi ciri. Pada *pre-processing* akan dilakukan *cropping*, *sharpening* dan normalisasi citra. Citra awal akan melalui proses *cropping* untuk menghilangkan bagian latar yang tidak digunakan. Citra hasil *cropping* akan dinormalisasi untuk mengurangi jumlah data pada proses ekstraksi ciri.

2.7 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) diperkenalkan oleh Pearson (1901) dan Hotelling (1933). Maksud dari penggunaan PCA sendiri adalah untuk menangkap variasi total di dalam kumpulan data yang dilatih, dalam hal ini data berupa citra dan untuk direpresentasikan variasi ini dengan variabel yang lebih sedikit. Suatu citra yang direpresentasikan dengan variabel yang lebih sedikit akan lebih mudah untuk ditangani dan dimengerti daripada jika direpresentasikan dengan raw pixel yang banyak dari citra tersebut [6].

Langkah-langkah pembentukan PCA :

1. Sebelum PCA dapat dilakukan, dilakukan *lexicographical ordering* untuk setiap citra yang akan dilatihkan dimana kolom yang satu diletakkan di samping kolom yang lain sehingga membentuk matriks yang merupakan matriks kolom. Matriks-matriks kolom tersebut disusun sedemikian rupa sehingga membentuk suatu matriks U dengan orde $m \times n$ dimana m adalah banyaknya jumlah piksel citra ($w \times h$) dan n adalah banyaknya citra. Matriks inilah yang akan digunakan sebagai masukan bagi PCA [6].
2. Menghitung rata-rata (mean) setiap baris dari matriks besar tersebut [6].

$$\bar{\mu} = \frac{1}{n} \sum_{j=1}^n \mu_j \quad (5)$$

$\bar{\mu}$ = vektor rata-rata
 μ_j = jumlah matriks

3. Mengurangi semua nilai di setiap baris matriks besar dengan rata-rata tersebut (data normal) [6].

$$X_{norm} = X - \bar{\mu} \quad (6)$$

4. Membuat matriks kovarian [6].

$$S = \frac{1}{n} X_{norm}^T X_{norm} \quad (7)$$

5. Mencari nilai eigen (S) dan vektor eigen (V) dari matriks kovarian. Selanjutnya, vektor eigen V dikorelasikan terhadap data normal sehingga dihasilkan vektor eigen U. Vektor eigen U ini kemudian diurutkan secara descending berdasarkan nilai eigen [6].
6. Memilih jumlah PC yang diinginkan. Principal Component (PC) adalah matriks pembawa ciri data. PC yang dihasilkan dalam PCA merupakan eigen vector yang terurut berdasarkan eigen value [6].

2.8 Klasifikasi

Setelah melalui ekstraksi ciri maka selanjutnya akan dilakukan klasifikasi pada citra dengan menggunakan metode *k - Nearest Neighbor*. Metode *k - Nearest Neighbor* menggunakan kombinasi nilai $k=1,3,5$ dan empat jenis *distance* yaitu *euclidean*, *cityblock*, *cosine*, *correlation*. Pada rancang bangun sistem ini, citra buah kakao akan diklasifikasikan menjadi tiga kelas yaitu normal, busuk buah dan helopeltis. Untuk citra latih, data hasil ekstraksi ciri akan disimpan dalam *database*. Pada proses pengujian, citra hasil ekstraksi ciri yang didapat akan dicocokkan dengan data yang telah disimpan pada *database* untuk kemudian dilakukan pengklasifikasian.

2.9 Perhitungan Akurasi dan Waktu Komputasi Sistem

Akurasi merupakan ukuran ketepatan sistem dalam mengenali input yang diberikan sehingga menghasilkan keluaran yang benar. Secara sistematis dapat didefinisikan dengan rumus berikut ini :

$$\text{Akurasi} = \frac{\text{Jumlah Benar}}{\text{Jumlah Data}} \times 100\% \quad (8)$$

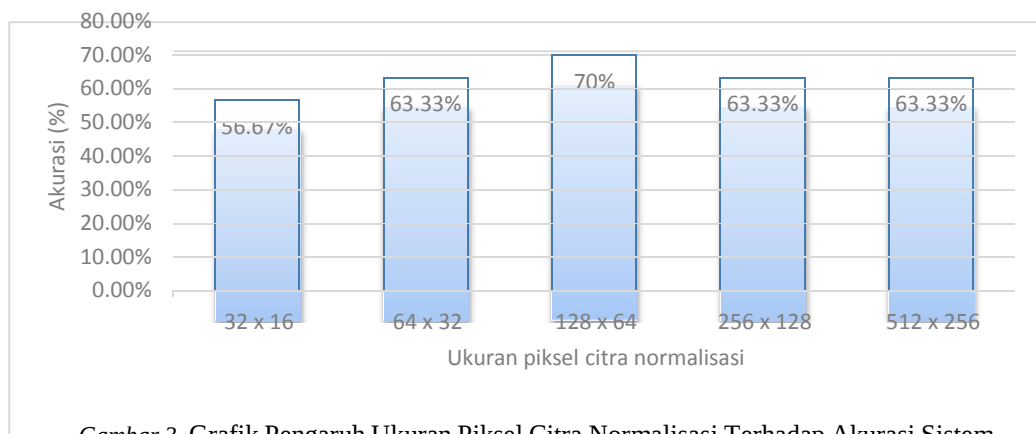
Waktu komputasi adalah waktu yang dibutuhkan sistem untuk melakukan suatu proses. Cara mendapatkan waktu komputasi pada sistem ini adalah dengan menggunakan toolbox yang telah disediakan oleh Matlab.

3. Pembahasan

3.1 Hasil Pengujian Pengaruh Ukuran Piksel Citra Normalisasi Terhadap Akurasi Sistem

Pengujian ini bertujuan untuk mengetahui apakah ukuran piksel citra normalisasi berpengaruh terhadap akurasi dari sistem. Ukuran piksel yang digunakan adalah 32x16 piksel, 64x32 piksel, 128x64 piksel, 256x128 piksel dan 512x256 piksel. Akurasi yang paling optimal didapatkan saat menggunakan ukuran citra normalisasi 128x64 piksel yaitu sebesar 70%. Saat menggunakan ukuran citra normalisasi 64x32 piksel, 256x128 piksel dan 512x256 piksel didapatkan akurasi yang lebih kecil yaitu sebesar 63,33%. Akurasi yang paling kecil didapat pada saat menggunakan ukuran citra normalisasi 32x16 piksel yaitu sebesar 56,57%. Dalam hal ini dapat dikatakan bahwa saat menggunakan ukuran citra normalisasi yang lebih besar dari 128x64 piksel akan memiliki variasi data yang semakin banyak sehingga penentuan batasan antar kelas makin kabur. Sedangkan saat

menggunakan ukuran citra normalisasi yang lebih kecil dari 128x64 piksel akan memiliki variasi data yang lebih sedikit sehingga penentuan batasan antar kelas makin susah.

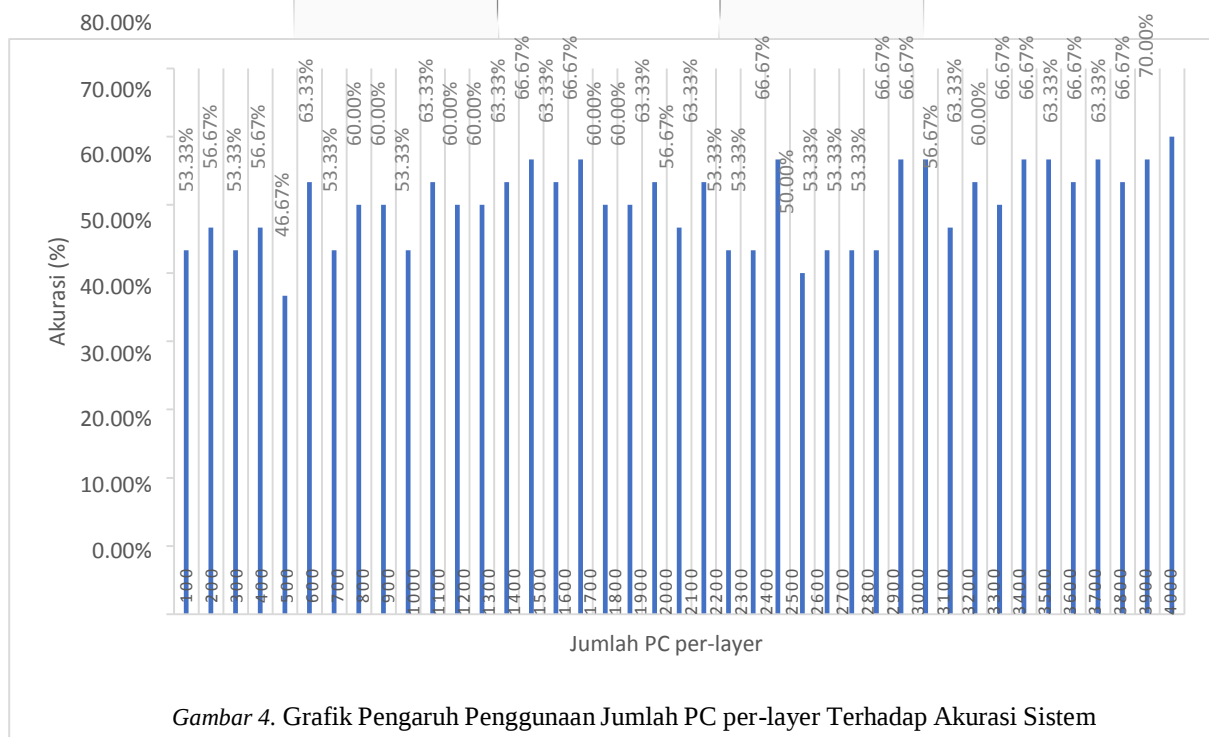


Gambar 3. Grafik Pengaruh Ukuran Piksel Citra Normalisasi Terhadap Akurasi Sistem

3.2 Hasil Pengujian Pengaruh Penggunaan Jumlah PC per-layer Terhadap Akurasi Sistem

Berdasarkan pengujian sebelumnya yaitu pengaruh ukuran citra normalisasi didapatkan akurasi yang paling optimal yaitu sebesar 70% dimana pengujian dilakukan saat menggunakan citra normalisasi berukuran 128x64 piksel. Maka pada pengujian kali ini akan menggunakan citra normalisasi berukuran 128x64 piksel sebagai acuan.

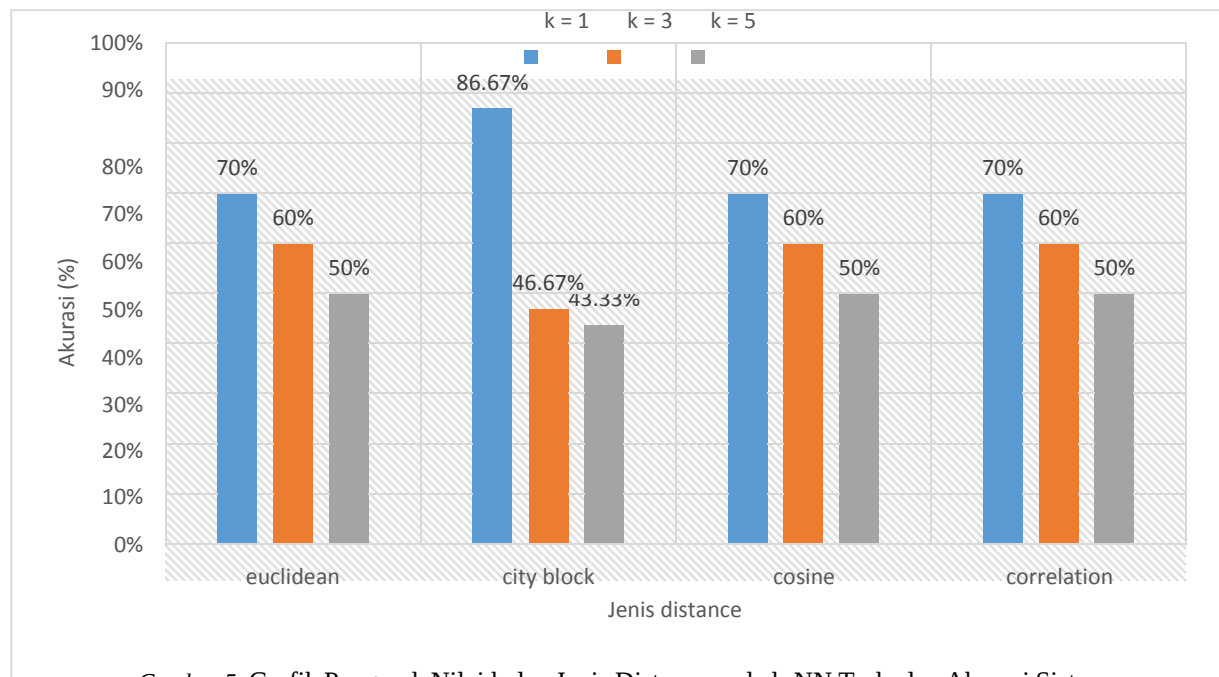
Parameter yang akan digunakan pada pengujian ini adalah menggunakan jumlah PC per-layer *default* dan dengan mengubah jumlah PC yang akan digunakan. Jumlah PC per-layer *default* adalah 4069. Akurasi yang didapatkan saat kondisi penggunaan jumlah PC *default* yaitu sebesar 70%. Sedangkan akurasi lebih kecil saat penggunaan jumlah PC per-layer yang dibatasi. Dapat disimpulkan bahwa penggunaan jumlah PC yang dibatasi akan mengurangi kandungan informasi dari data itu sendiri sehingga saat proses klasifikasi akan ada beberapa informasi yang hilang atau tidak sesuai antara data saat pelatihan dan pengujian.



Gambar 4. Grafik Pengaruh Penggunaan Jumlah PC per-layer Terhadap Akurasi Sistem

3.3 Hasil Pengujian Pengaruh Nilai k dan Jenis Distance pada k-NN Terhadap Akurasi Sistem

Pada pengujian klasifikasi K-NN digunakan metode *distance euclidean*, *city block*, *cosine* dan *correlation* dan nilai $k = 1, 3, \text{ dan } 5$. Berdasarkan gambar 4.3 diatas, dapat diketahui bahwa jenis metode *distance* yang menghasilkan akurasi tertinggi adalah metode *Cityblock* dengan $k = 1$ yang memberikan akurasi sebesar 86,67%. Berdasarkan hal tersebut maka dapat dikatakan bahwa semakin tinggi nilai k yang diberikan akan membuat akurasi sitem menjadi semakin kecil. Pada umumnya semakin tinggi nilai k yang diberikan akan semakin mengurangi efek *noise* pada proses klasifikasi, tetapi hal tersebut juga akan membuat batasan antar setiap klasifikasi akan menjadi semakin kabur. Nilai $k = 1$ berarti terdapat 1 vektor berdekatan yang digunakan sebagai pembandingan. Dengan menggunakan nilai tersebut sudah dapat mewakili vektor ciri dari berbagai kelas.



Gambar 5. Grafik Pengaruh Nilai k dan Jenis Distance pada k-NN Terhadap Akurasi Sistem

3.4 Analisis Waktu Komputasi

Pengujian waktu komputasi sistem dilakukan dengan menggunakan 5 kali percobaan dan beberapa parameter yang sebelumnya sudah di uji dan menghasilkan akurasi paling optimal yaitu menggunakan ukuran citra normalisasi 128 x 64 piksel, tanpa pembatasan jumlah PC dan menggunakan *distance cityblock* dengan nilai $k = 1$. Rata-rata waktu yang dibutuhkan untuk proses *pre-processing* adalah 0,31284 detik, sedangkan rata-rata waktu komputasi yang dibutuhkan untuk proses ekstraksi ciri adalah 0,03598 detik dan rata-rata waktu yang dibutuhkan untuk proses klasifikasi adalah 0,07774 detik. Proses ekstraksi ciri dan klasifikasi membutuhkan waktu yang paling cepat dibandingkan dengan proses *pre-processing* karena tahapan pada *pre-processing* lebih panjang dibandingkan dengan proses ekstraksi ciri dan klasifikasi.

4. Kesimpulan dan Saran

4.1 Kesimpulan

Berdasarkan hasil analisis terhadap pengujian yang dilakukan pada sistem, maka dapat diambil kesimpulan sebagai berikut:

1. Sistem identifikasi jenis penyakit pada buah kakao menggunakan metode ekstraksi ciri PCA dan metode klasifikasi k-NN dapat diimplementasikan dengan akurasi tertinggi sebesar 86,67%.
2. Ukuran citra normalisasi sangat berpengaruh terhadap akurasi sistem dan yang paling ideal untuk digunakan adalah 128 x 64 piksel.

3. Pembatasan jumlah PC yang digunakan dapat mempengaruhi akurasi sistem, yang paling optimal didapatkan saat kondisi penggunaan jumlah PC *default* yaitu sebesar 70%, sedangkan akurasi menurun saat penggunaan jumlah PC yang dibatasi.
4. Pada proses klasifikasi dengan k-NN, jenis metode *distance* yang menghasilkan akurasi tertinggi adalah metode *Cityblock* dengan $k = 1$ yang memberikan akurasi sebesar 86,67%.
5. Rata-rata waktu komputasi yang dibutuhkan untuk proses *pre-processing* adalah 0,31284 detik, sedangkan rata-rata waktu komputasi yang dibutuhkan untuk proses ekstraksi ciri adalah 0,03598 detik dan rata-rata waktu yang dibutuhkan untuk proses klasifikasi adalah 0,07774 detik.

4.2 Saran

Berikut adalah beberapa saran untuk pengembangan Tugas Akhir atau penelitian selanjutnya :

1. Merperbanyak citra latih dengan menambahkan beberapa jenis penyakit lainnya yang menyerang buah maupun bagian lainnya pada tanaman kakao.
2. Menggunakan metode ekstraksi ciri lain dan metode klasifikasi selain k-NN untuk meningkatkan akurasi sistem.
3. Dapat dibuat sistem identifikasi secara *real time* dan digunakan pada smartphone.

Daftar Pustaka:

- [1] Yenita Afriyeni, Nasril Nasir, Periadnadi, Jumjunidang, "Jenis-Jenis Jamur pada Pembusukan Buah Kakao (*Theobroma cacao*, L.) di Sumatera Barat", Jurnal Biologi Universitas Andalas, 2013.
- [2] Direktorat Perlindungan Perkebunan, Direktorat Jenderal Bina Produksi Perkebunan Departemen Pertanian, "Musuh Alami, Hama dan Penyakit Tanaman Kakao edisi kedua", Proyek Pengendalian Hama Terpadu Perkebunan Rakyat, 2002.
- [3] Rinaldi Munir, "Pengolahan Citra Digital dengan Pendekatan Algoritmik", Bandung: Informatika, 2004.
- [4] Akhmad Fauzan, Lahmuddin Lubis, Mukhtar Iskandar Pinem, "Keparahan Penyakit Busuk Buah Kakao (*Phytophthora palmivora* Butl.) pada Beberapa Perkebunan Kakao Rakyat yang Berbeda Naungan di Kabupaten Langkat", Jurnal Online Agroekoteknologi Vol.1 No.3, 2013.
- [5] Puslitbang Perkebunan, "Budidaya dan Pasca Panen Kakao", Pusat Penelitian dan Pengembangan Perkebunan, 2010.
- [6] Puji Lestari, Bambang Hidayat, Eko Susatio, "Deteksi Cacat Daun Teh *Camellia Sinensis* dengan Pengolahan Citra Digital dan JST Learning Vector Quantization," IT Telkom Journal on ICT, 2012.
- [7] Eko Prasetyo, "Data Mining Konsep dan Aplikasi Menggunakan Matlab", Penerbit Andi, 2012.
- [8] Darma Putra, Pengolahan Citra Digital.: Penerbit Andi, 2010.
- [9] Retno Nugroho Whidhiasih, Nursinta Adi Wahanani, Supriyanto, "Klasifikasi Buah Belimbing Berdasarkan Citra Red-Green-Blue Menggunakan KNN dan LDA," Jurnal Penelitian Ilmu Komputer, System Embedded & Logic Universitas Pakuan Bogor, 2013.
- [10] Yulia Saraswati, "Sistem Klasifikasi Jenis Dan Kematangan Buah Tomat Berdasarkan Bentuk Dan Ukuran Serta Warna Permukaan Kulit Buah Berbasis Pengolahan Citra", Tugas Akhir, Bandung: Institut Teknologi Telkom, 2009.