

# IMPLEMENTASI TRANSFORMASI *HAAR WAVELET* UNTUK DETEKSI CITRA JERUK NIPIS YANG BUSUK

Lidya Andriani Sunjoyo<sup>1</sup>  
lidya.andriani@ti.ukdw.ac.id

R. Gunawan Santosa<sup>2</sup>  
gunawan@staff.ukdw.ac.id

Kristian Adi Nugraha<sup>3</sup>  
adinugraha@staff.ukdw.ac.id

## *Abstract*

*Lime is a fruit that has been widely cultivated and used in Indonesia. Many products use this fruit in the production process. The process of sorting fruit is undeniably a very substantial early process. It is necessary for large-scale be aware of this in term of result and time required for the sorting process. Pattern Recognition is a discipline that focuses on classifying or picturing an object based on characteristics or main attribute of the object. In this research, the author implements Haar Wavelet Transformation method by characteristic extraction based on colour and texture , performs classification using K-Nearest Neighbor (k-NN) to detect indication of rotten lime and the grade of k on k-NN so the accuracy of the result could be acquired. Based on analysis result, Haar Wavelet Transformation method is able to be implemented to detect the indication of rotten lime and most optimal accuracy level of this system reaches the number of 85 percent.*

**Keywords:** *Pattern Recognition, k-NN, Haar Wavelet*

## 1. Pendahuluan

Saat ini sudah banyak produk kemasan maupun produk instan yang memanfaatkan buah jeruk nipis. Produk-produk tersebut tidak sedikit yang menggunakan buah jeruk nipis (*Citrus Aurantifolia*) untuk diolah sebagai bahan pokok atau tambahan pada minuman, makanan dan produk kemasan lainnya. Jeruk nipis juga sangat mudah ditemui di Indonesia karena gampang untuk dibudidayakan. Dalam proses pemanfaatannya tentu perlu dilakukan pemilihan buah yang baik. Perusahaan tentunya juga perlu memberikan kualitas yang baik untuk hasil produksinya. Oleh karena itu, proses awal pemilihan buah di pabrik menjadi sangat penting.

Perusahaan yang berskala besar tentunya memerlukan waktu dan tenaga yang lebih dalam penyortirannya. Sehingga diperlukan usaha untuk menggunakan teknologi dalam otomatisasi penyortiran jeruk yang segar dan jeruk yang busuk. Otomatisasi ini dilakukan dengan menggunakan masukan pemrosesan citra digital dan mengolahnya (Low, 1991). Penelitian ini mencoba untuk menerapkan transformasi Haar wavelet dan metode klasifikasi k-NN untuk mengklasifikasikan jeruk yang segar dan jeruk yang busuk berdasarkan citra dari jeruk tersebut. Dan juga akan dilakukan penelitian seberapa akurat transformasi Haar wavelet dan metode klasifikasi k-NN dalam mendeteksi citra jeruk nipis yang busuk.

Dari permasalahan diatas, penulis akan membuat sistem implementasi Transformasi Haar Wavelet untuk deteksi busuk pada citra jeruk nipis. Akan dilakukan ekstraksi ciri berdasarkan tekstur dan warna dan kemudian diklasifikasi menggunakan metode k-Nearest Neighbor. Penelitian yang dilakukan ini akan menerapkan metode Transformasi Haar Wavelet untuk mendeteksi citra jeruk nipis yang busuk dan melihat tingkat akurasinya.

Beberapa penelitian seperti Listyaningrum (2007) dalam penelitiannya berjudul Analisis Tekstur menggunakan Metode Transformasi Paket Wavelet membandingkan analisis tekstur menggunakan metode Haar Wavelet, Coiflet dan Daubechies 8. Tidak dilakukan klasifikasi dalam penelitian ini, hanya menghitung jarak euclidean saja. Hasil dari penelitian ini didapat bahwa tiap metode memiliki kelebihan dan kekurangannya untuk tiap

---

<sup>1</sup> Universitas Kristen Duta Wacana, Fakultas Kristen Duta Wacana, Teknik Informatika

<sup>2</sup> Universitas Kristen Duta Wacana, Fakultas Kristen Duta Wacana, Teknik Informatika

<sup>3</sup> Universitas Kristen Duta Wacana, Fakultas Kristen Duta Wacana, Teknik Informatika

jenis tekstur. Namun secara umum metode *Haar Wavelet* memiliki kesalahan paling sedikit dibandingkan yang lainnya.

Prihartono (2011) dalam penelitiannya yang berjudul Identifikasi Iris Mata Menggunakan Alihragam Wavelet Haar melakukan pengenalan iris mata menggunakan metode *Haar Wavelet* untuk mengetahui identitas pemiliknya.

Sedangkan Leidiyana (2013) meneliti Penerapan Algoritma K-Nearest Neighbor untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bermotor.

Dari beberapa penelitian yang telah dilakukan oleh para peneliti tersebut, maka penelitian ini merupakan gabungan dua tahap proses yaitu ekstraksi fitur jeruk nipis dengan menggunakan Haar Wavelet dan klasifikasi jeruk nipis busuk atau segar dengan metode K-Nearest Neighbor. Sehingga diharapkan dapat menghasilkan sistem yang dapat mengenali citra jeruk nipis yang busuk dengan tepat.

## 2. Landasan Teori

### 2.1. k-Nearest Neighbors (k-NN)

Algoritma k-NN adalah suatu metode yang menggunakan algoritma *supervised*. Tujuan dari algoritma k-NN adalah untuk mengklasifikasi objek baru berdasarkan atribut dan *training samples*.

Algoritma k-NN menggunakan klasifikasi ketetanggaan sebagai nilai prediksi dari sampel uji yang baru, misalnya pada penelitian Algoritma k-Nearest Neighbor dalam Klasifikasi Data Hasil Produksi Kelapa Sawit pada PT. Minamas . (Krisandi, Helmi, & Prihandono, 2013)

Jarak yang digunakan adalah jarak *Cosine Similarity Measure*, yang didefinisikan sebagai berikut .

$$\text{Similarity}(d_j, q) = \frac{d_j \cdot q}{|d_j| \cdot |q|} = \frac{\sum_{i=1}^t w_{ij} \cdot w_{iq}}{\sqrt{\sum_{i=1}^t w_{ij}^2 \sum_{i=1}^t w_{iq}^2}} \quad [1]$$

Keterangan:

$d_j$  = ciri data *learning*

$q$  = ciri data *query*

$w_{ij}$  = vektor ciri ke- $i$  data *learning*

$w_{iq}$  = vektor ciri ke- $i$  data *query*

Pada fase pembelajaran, algoritma ini hanya melakukan penyimpanan vektor-vektor fitur dan klasifikasi dari data pembelajaran. Sedangkan pada fase klasifikasi dilakukan langkah sebagai berikut:

- Fitur-fitur yang sama dihitung untuk data *test* (yang klasifikasinya tidak diketahui).
- Jarak dari vektor yang baru ini terhadap seluruh vektor data pembelajaran dihitung
- Sejumlah  $k$  buah yang paling dekat diambil
- Titik yang baru klasifikasinya diprediksikan termasuk pada klasifikasi terbanyak dari titik-titik tersebut (Whidhiasih, Wahanani & Supriyanto , 2013)

### 2.2. Grayscale

Gambar *grayscale* adalah gambar yang hanya terdiri dari beberapa tingkat warna dari putih hingga hitam. Gambar *grayscale* 8 bit memiliki 256 tingkat warna abu-abu mulai dari putih hingga hitam. Pada dasarnya proses *grayscale* dilakukan dengan meratakan nilai piksel dari 3 nilai RGB menjadi 1 nilai. Untuk memperoleh hasil yang lebih baik, nilai piksel tidak langsung dibagi menjadi 3 melainkan terdapat persentasi dari masing-masing nilai. Salah satu persentasi yang sering digunakan adalah 29,9% dari warna merah (*Red*), 58,7% dari warna hijau (*Green*), dan 11,4% dari warna biru (*Blue*). Nilai piksel didapat dari jumlah persentasi 3 nilai tersebut. (Nurhasanah, 2012)

Untuk proses *grayscale* digunakan formula sebagai berikut.

$$grayscale = (R \times 0.21) + (G \times 0.75) + (B \times 0.07) \quad [2]$$

Keterangan:

*grayscale* = intensitas warna keabuan piksel

R = intensitas warna piksel merah

G = intensitas warna piksel hijau

B = intensitas warna piksel biru

Derajat keabuan memiliki rentang nilai dari  $I_{min}$  sampai  $I_{max}$ , atau  $I_{min} < f < I_{max}$ , selang  $(I_{min}, I_{max})$  disebut skala keabuan. Biasanya selang  $(I_{min}, I_{max})$  sering digeser untuk alasan-alasan praktis menjadi selang  $[0, L]$ , yang dalam hal ini nilai intensitas 0 menyatakan hitam, nilai intensitas L menyatakan putih, sedangkan nilai intensitas antara 0 sampai L bergeser dari hitam ke putih. Sebagai contoh citra *grayscale* dengan 256 level artinya mempunyai skala abu dari 0 sampai 255 atau  $[0, 255]$ , yang dalam hal ini intensitas 0 menyatakan hitam, intensitas 255 menyatakan putih, dan nilai antara 0 sampai 255 menyatakan warna keabuan yang terletak antara hitam dan putih. (Yulianto & Hatta, 2011)

### 2.3. Transformasi Haar Wavelet

*Haar* adalah *wavelet* yang paling tua dan sederhana, diperkenalkan oleh Alfred Haar pada tahun 1909. Beberapa matriks yang digunakan untuk transformasi Haar Wavelet adalah seperti di bawah ini (Arora, Brar, & Kumar, 2014)

Matriks *lowpass* Haar  $H_0^{(n)}$  dan *highpass* Haar  $H_1^{(n)}$  :

$$H_0^{(n)} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ddots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \in R^{n \times n} \quad [3]$$

$$H_1^{(n)} = \frac{1}{2} \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & \dots & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & \dots & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & \dots & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & \dots & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & \dots & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \ddots & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & 1 & -1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \in R^{n \times n} \quad [4]$$

Matrik untuk melakukan *down sampling*  $\downarrow 2$   $D^{(n)}$  adalah:

$$D^{(n)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix} \in R^{\frac{n}{2} \times n} \quad [5]$$

Hasil proses *down sampling* dengan ekspresi  $D^{(n)}H_0^{(n)}$  (*filter lowpass*) dan  $D^{(n)}H_1^{(n)}$  (*filter highpass*) :

$$T^{(n)} = \frac{1}{2} \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & & & & \vdots \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \in R^{\frac{n}{2} \times n} \quad [6]$$

$$B^{(n)} = \frac{1}{2} \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \dots & & & & \vdots \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \in R^{\frac{n}{2} \times n} \quad [7]$$

Berikut adalah langkah-langkah metode Transformasi *Haar Wavelet*

1. *Transpose* matriks *input*
2. Mengkalikan filter *low* dan *high* yang sudah digabung dengan matriks *input*
3. Melakukan proses *transpose* matriks hasil dari langkah 2
4. Mengkalikan filter *low* dan *high* yang sudah digabung dengan hasil *transpose* langkah 3
5. Menampilkan *output* yang terdiri dari 4 subbidang yaitu LL, HL, LH, LL
6. Jika lebih dari 1 level, ulangi lagi langkah 1-5 dengan input matriks subbidang LL dari level sebelumnya, begitu seterusnya.

Matriks *input* dari metode Transformasi *Haar Wavelet* tersebut adalah matriks citra yang sudah berupa *grayscale*. Setelah proses 1 sampai dengan 5, maka akan didapat matriks fitur (karakteristik) dari citra tersebut. Jadi dengan kata lain Transformasi *Haar Wavelet* mengambil fitur dari matriks citra yang akan diolah.

## 2.4. Fitur Energi Subbidang

Fitur adalah karakteristik unik dari suatu objek. Pada citra fitur merupakan ciri yang dimiliki oleh citra tersebut yang menjadi pembeda antar citra satu dan citra yang lainnya. (Dharma, 2010)

Berikut adalah formula perhitungan energi .

$$E_{\text{subband\_scale}} = \frac{\sum_{x,y} (d_{x,y}^{\text{subbidang}})^2}{n} \quad [8]$$

Keterangan:

$d_{x,y}^{\text{subbidang}}$  = nilai piksel suatu subbidang pada titik koordinat (x,y)

n = jumlah piksel pada setiap subbidang

## 2.5. Akurasi Sistem

Tingkat keakuratan sistem dapat dihitung berdasarkan jumlah perbandingan jumlah data uji yang dapat diklasifikasi dengan benar dengan jumlah seluruh data uji. Dapat ditulis dengan persamaan:

$$\text{Acc}(T) = \frac{T_{\text{pos}}}{n} \quad [9]$$

Keterangan:

$T_{\text{pos}}$  = jumlah data uji yang dapat diklasifikasi dengan benar

n = jumlah seluruh data uji

### 3. Metodologi penelitian

Metodologi penelitian yang dilakukan pada penelitian ini adalah membuat sistem kemudian menguji cobakan sistem tersebut dengan masukan citra jeruk nipis yang akan menghasilkan keluaran jenis jeruk yang segar atau busuk. Setelah sistem yang dibuat selesai, maka dilakukan penelitian sesuai dengan urutan-urutan berikut yang ada pada sistem.

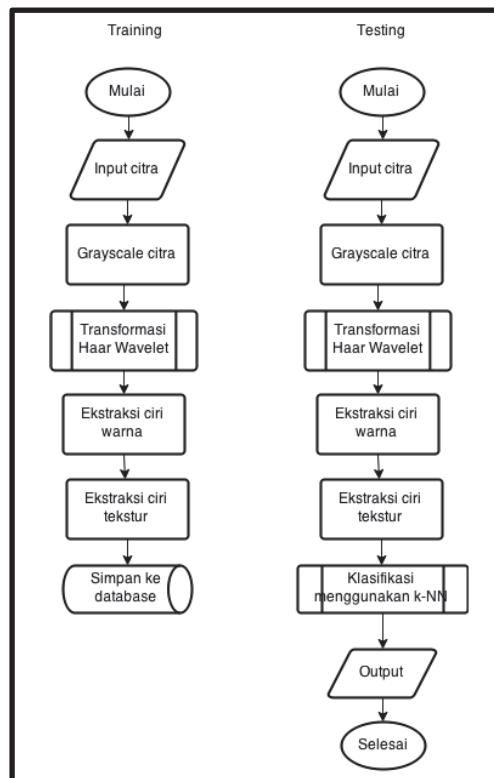
Penelitian yang dilakukan pada sistem untuk mendeteksi citra jeruk nipis yang busuk memiliki beberapa langkah yang ada pada sistem yaitu : mengubah citra menjadi *grayscale*, transformasi *Haar Wavelet*, ekstraksi ciri, dan perhitungan jarak menggunakan *Cosine Similarity Measure* dan kemudian diklasifikasi dengan *k-NN*. Melakukan pelatihan dengan menyimpan ciri beserta hasil klasifikasi ke dalam basis data dan melakukan pengujian dengan memberikan input citra jeruk nipis yang baru, kemudian melihat hasil klasifikasi sistem dan menghitung tingkat keakuratan/keberhasilannya. Citra pelatihan berjumlah 30 citra yaitu 15 citra jeruk nipis busuk dan 15 citra jeruk nipis segar. Sedangkan citra untuk pengujian sebanyak 20 foto yaitu 10 citra jeruk nipis busuk dan 10 citra jeruk nipis segar. Berikut adalah contoh citra jeruk nipis busuk (Gambar 1), citra jeruk nipis segar (Gambar 2) serta diagram alir sistem utama.



Gambar 1. Citra Jeruk Nipis Busuk



Gambar 2. Citra Jeruk Nipis Segar



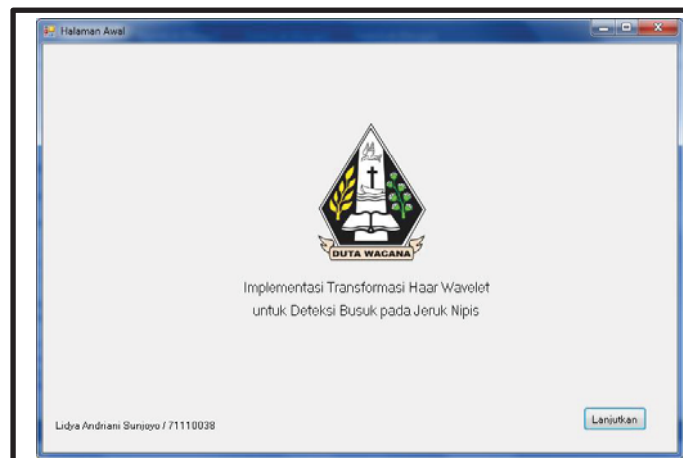
Gambar 3. Diagram Alir Utama Sistem

## 4. Hasil dan Pembahasan

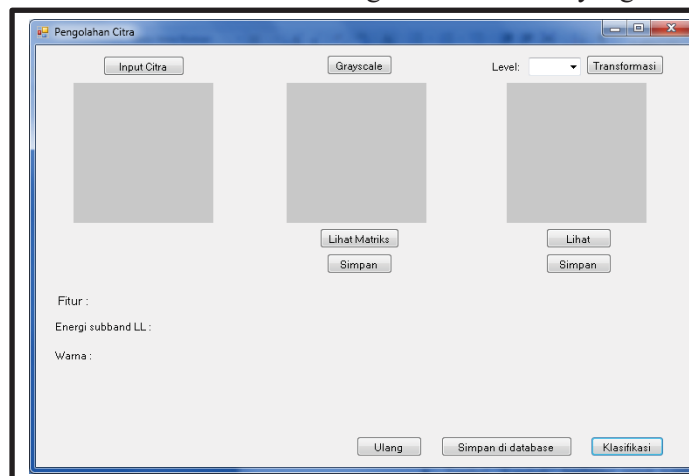
### 4.1. Implementasi Sistem

Sistem diimplementasikan menggunakan bahasa pemrograman visual basic. Sistem yang dibuat memiliki beberapa fitur, yaitu : mengubah citra warna menjadi *grayscale*, melakukan transformasi *Haar Wavelet* sesuai level yang diinginkan, menghitung dan menampilkan ekstraksi ciri warna dan tekstur (dihitung dari energi), menyimpan data ke basis data menyimpan citra hasil *grayscale* dan transformasi, menampilkan matriks citra, *reset* program, melakukan klasifikasi menggunakan metode k-NN sesuai dengan k yang diinputkan, dan menampilkan hasilnya.

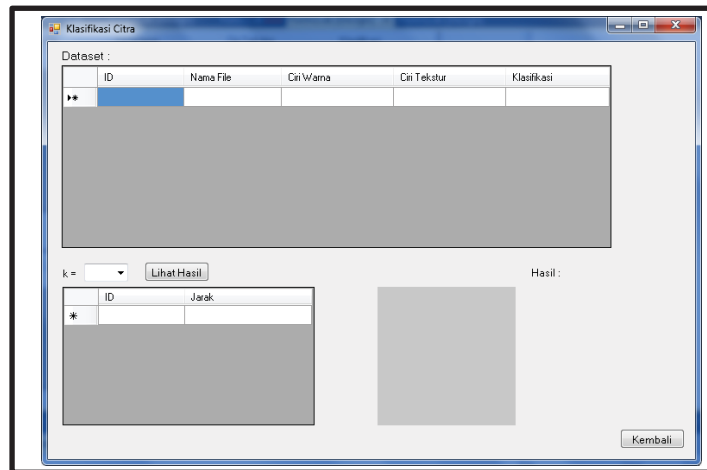
Pada halaman awal program (Gambar 4) hanya terdapat logo UKDW, judul program, keterangan nama pembuat dan tombol “Lanjutkan” yang akan menuju pada halaman pengolahan citra (Gambar 5). Halaman pengolahan citra ini adalah halaman sebelum melakukan klasifikasi. Semua proses yang ada dihalaman ini berhubungan dengan citra, seperti input, *grayscale*, transformasi, simpan citra, lihat matriks citra, ekstraksi ciri dan menyimpan di basis data. Apabila memilih menu ulang, program akan direset, apabila memilih menu simpan maka pengguna akan diminta untuk menginputkan kelas citra dan apabila memilih menu klasifikasi, maka akan menuju pada halaman Klasifikasi (Gambar 6), dimana proses klasifikasi dilakukan untuk mendapatkan hasil akhir. Terdapat 2 tabel pada halaman klasifikasi. Tabel yang berada diatas adalah tabel untuk menampilkan data dari basis data yang sudah terisi dari fase pelatihan, sedangkan tabel yang dibawah untuk menampilkan data yang terambil sebanyak k dalam proses klasifikasi k-NN.



Gambar 4. Halaman Awal Program atau Sistem yang dibuat



Gambar 5. Halaman Pengolahan Citra



Gambar 6. Halaman Klasifikasi Citra

#### 4.2. Pengujian Sistem

Pada tahap ini akan dilakukan beberapa pengujian yang akan digunakan untuk menguji tingkat keakuratan sistem. Percobaan akan dilakukan dengan mengubah level transformasi *Haar Wavelet* dan nilai *k* (tetangga) dalam klasifikasi *k*-NN. Nantinya level transformasi akan dikombinasikan dengan nilai *k* untuk dilihat tingkat keakuratannya. Pada percobaan ini digunakan nilai 2,4 dan 6 untuk level transformasi dan 5, 7, 9, 11, 13,15 untuk nilai *k*.

Tabel 1.

Tabel Hasil Pengujian Sistem

Level	k	Klasifikasi Benar	Klasifikasi Salah	Akurasi (%)
2	5	16	4	80
2	7	16	4	80
2	9	16	4	80
2	11	16	4	80
2	13	16	4	80
2	15	16	4	80
4	5	17	3	85
4	7	17	3	85
4	9	17	3	85
4	11	17	3	85
4	13	16	4	80
4	15	17	3	85
6	5	17	3	85
6	7	17	3	85
6	9	17	3	85
6	11	17	3	85
6	13	16	4	80
6	15	16	4	80

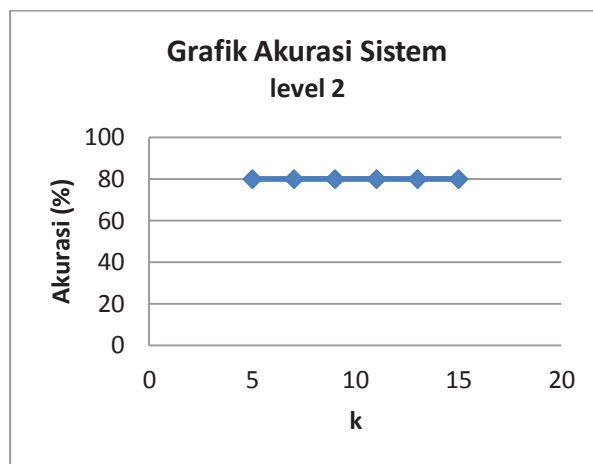
#### 4.3. Analisis Hasil Pengujian

Setelah melakukan pengujian sistem, akan dilihat bagaimana hasil penerapan Transformasi *Haar Wavelet* dan bagaimana tingkat akurasi sistem. Dari fase pelatihan, dapat dilihat perbedaan ciri citra buah jeruk nipis busuk dan yang segar. Terdapat perbedaan angka energi/ciri tekstur dan warna yang cenderung lebih besar pada citra jeruk segar dibanding yang busuk. Perbedaan ini memang tidak terlalu jauh, ada juga ciri citra busuk dan segar yang

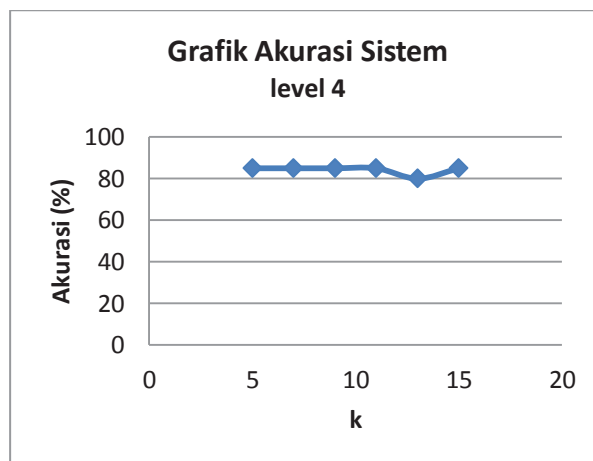
hampir sama. Pada pelatihan dengan citra yang ditransformasi level 2, energi citra jeruk busuk berkisar antara angka 36 - 70 dan warna berkisar antara 29 - 50. Sedangkan energi citra jeruk segar berkisar antara 66 sampai 100 dan warna berkisar antara 49 sampai 62. Pada level 4, ciri citra jeruk yang busuk teksturnya antara 34 - 70 dan warna 29 - 50, sedangkan yang segar ciri teksturnya 50 - 100 dan warna 48 - 61. Dan pada level 6, ciri citra jeruk yang busuk teksturnya antara 15 - 63 dan warna 21 - 47, sedangkan yang segar ciri teksturnya 53 - 100 dan warna 43 - 60.

Pada pengujian dengan level transformasi 2 dengan nilai  $k = 5, 7, 9, 11, 13,$  dan  $15$  ternyata didapat hasil akurasi sistem yang sama yaitu  $80\%$ . Terdapat 4 kesalahan pengenalan dari 20 citra uji, yaitu 4 citra jeruk busuk yang salah dikenali sebagai citra jeruk segar. %. Grafik akurasi sistem level 2 dapat dilihat pada Gambar 7.

Selanjutnya pengujian dengan citra yang ditransformasi level 4, baik citra pelatihan maupun pengujian. Hasil yang sama didapat ketika pengujian dengan  $k = 5, 7, 9, 11$  dan  $15$ . Terdapat 3 kesalahan pengenalan dan pada pengujian ini akurasi sistem mencapai  $85\%$ . Sedangkan pada pengujian  $k=13$  terdapat 4 kesalahan. Sehingga akurasi sistem pada pengujian ini adalah  $80\%$ . Grafik akurasi sistem level 4 dapat dilihat pada Gambar 8.



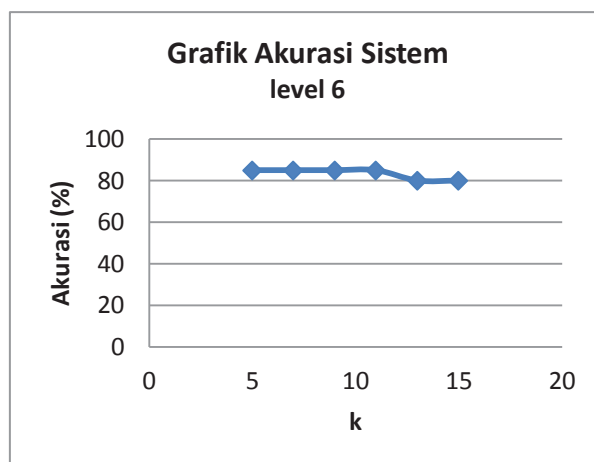
Gambar 7. Grafik Akurasi Sistem untuk Citra Transformasi Level 2



Gambar 8. Grafik Akurasi Sistem untuk Citra Transformasi Level 4

Pengujian terakhir adalah pengujian dengan citra yang ditransformasi level 6 baik citra pelatihan maupun pengujian. Pada klasifikasi dengan  $k=5$  ternyata tidak ada perbedaan hasil klasifikasi dengan pengujian sebelumnya. Hasil yang sama didapat pada pengujian dengan  $k = 7, 9,$  dan  $11$ . Pada semua pengujian ini akurasi sistem mencapai  $85\%$  karena ada 3 kesalahan klasifikasi. Sedangkan pada pengujian dengan  $k = 13$  dan  $15$  ada perbedaan hasil klasifikasi. Terdapat 4 kesalahan citra kelas busuk yang salah klasifikasi dan akurasi sistemnya adalah  $80\%$ . %. Grafik akurasi sistem level 6 dapat dilihat pada Gambar 9.





Gambar 9. Grafik Akurasi Sistem untuk Citra Transformasi Level 6

Dari seluruh hasil pengujian, dapat dilihat Transformasi *Haar Wavelet* dapat diimplementasikan untuk deteksi citra jeruk nipis yang busuk dengan baik. Perubahan level menjadi lebih tinggi dapat berpengaruh dengan hasil klasifikasi yang semakin mendekati kenyataan. Dilihat dari akurasi sistem pada level 2 yang seluruhnya hanya 80% dan meningkat menjadi 85% pada level 4 dan 6. Sedangkan pemilihan k tidak didasari dari semakin besar k maka semakin baik klasifikasi. Pada pengujian, hasil yang baik didapat ketika  $k = 5, 7, 9$ , dan 11. Pada  $k = 13$ , dari percobaan pada level 4 dan 6 justru didapat penurunan akurasi sistem. Sedangkan pada  $k = 15$  hasilnya berubah-ubah yaitu 80% di level 4 dan 85% di level 6.

## 5. Kesimpulan

- Berdasarkan hasil implementasi dan analisis sistem, maka dapat disimpulkan bahwa:
- Untuk mendeteksi citra jeruk nipis yang busuk dilakukan transformasi terhadap citra sesuai level yang diinginkan, diekstraksi ciri warna dan teksturnya, dihitung jarak kedekatan citra uji dengan tiap citra pelatihan, dan diklasifikasi menggunakan k-NN sehingga bisa didapatkan hasil pengenalan citra jeruk busuk atau segar. Dengan demikian, metode *Haar Wavelet* dapat diimplementasi untuk deteksi citra jeruk nipis.
  - Tingkat keakuratan penerapan metode Transformasi *Haar Wavelet* dalam mendeteksi citra jeruk nipis yang busuk yang paling baik adalah 85%, yaitu didapat ketika level transformasi *Haar Wavelet* 4 dengan nilai k dalam klasifikasi k-NN 5, 7, 9, 11, 15 dan level 6 dengan nilai k 5, 7, 9, dan 11.

## Daftar Pustaka

- Arora, S., S. Brar, Y., & Kumar, S. (2014). Haar Wavelet Transform for Solution of Image Retrieval. *International Journal of Advanced Computer and Mathematical Sciences*, 1-5.
- Dharma, P. (2010). *Pengolahan Citra Digital*. Yogyakarta: Andi.
- Krisandi, N., Helmi, & Prihandono, B. (2013). Algoritma k-Nearest Neighbor dalam Klasifikasi Data Hasil Produksi Kelapa Sawit pada PT. Minamas. *Buletin Ilmiah Math. Stat. dan Terapannya*, 34.
- Leidiyana, H. (2013). Penerapan Algoritma K-Nearest Neighbor untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bermotor. *Jurnal Penelitian Ilmu Komputer, System Embedded & Logic*, 65-76.
- Listyaningrum, R., Santoso, I., & Isnanto, R. R. (2007). Analisis Tekstur Menggunakan Metode Transformasi Paket Wavelet. Semarang.
- Low, A. (1991). *Introductory Computer Vision and Image Processing*. Singapore: McGraw-Hill Book Co.
- Nurhasanah. (2012). Pendeteksian Tepi Citra CT Scan dengan Menggunakan Laplacian of Gaussian (LOG). *POSITRON*, 17 - 22.
- Prihartono, T. D., Isnanto, R. R., & Santoso, I. (2011). Identifikasi Iris Mata Menggunakan Alihragam Wavelet Haar. *TRANSMISI*, 71-75.
- Whidhiasih, R. N., Wahanani, N. A., & Supriyanto. (2013). Klasifikasi Buah Belimbing Berdasarkan Citra Red-Green-Blue Menggunakan KNN dan LDA. *Jurnal Penelitian Ilmu Komputer, System Embedded & Logic*, 29-35.
- Yulianto, A., & Hatta, A. M. (2011). Rancang Bangun Spektrometer Menggunakan Prisma Dan Webcam. 1-8.