

## Deteksi Jumlah Jeruk Menggunakan Metode Tranformasi Hough

Elva Amalia<sup>1</sup>, Andi Nurul Izzah<sup>2</sup>, Tsabita Syalza Billa<sup>3</sup>, Wanda Hamidah<sup>4</sup>,  
Andi Baso Kaswar<sup>5</sup>, Marwan Ramdhany Edy<sup>5</sup>

<sup>1</sup>elvaamaliaaa@gmail.com, <sup>2</sup>a.nurulizzah281@gmail.com, <sup>3</sup>tzabitasalsa@gmail.com, <sup>4</sup>wandahmdh26@gmail.com,  
<sup>5</sup>a.baso.kaswar@unm.ac.id, <sup>6</sup>marwanre@unm.ac.id  
<sup>123456</sup>Universitas Negeri Makassar

**Received :** 25 Apr 2023  
**Accepted :** 16 May 2023  
**Published :** 23 May 2023

### Abstract

**Abstract:** A common problem experienced by fruit traders is in calculating the number of citrus fruits. The activity of calculating manually takes a long time let alone sales on a large scale. From this problem, the author proposes the Hough Transform method for citrus fruit detection and calculation systems. In this study, it describes the system process into several stages, namely: Image input, grayscale image conversion, applying a median filter, stretching contrast, lowering brightness, and hough detection. The total dataset used in this study amounted to 70 orange image datasets. For the imagery used, the Hough Transformation algorithm is very instrumental in detecting circles on citrus fruit objects. Therefore, it is necessary to have imagery data with a uniform background to improve the clarity of the findings of circular object identification. In addition to the background factor of the object, the sensitivity value also affects the accuracy when identifying circles. Based on the results and discussion of the trial, the system can detect and calculate oranges quite well with an accuracy of up to 95.4%.

**Keywords:** Orange Image Detection, Image Intensity, Hough Transformation

### Abstrak

Permasalahan umum yang dialami oleh pedagang buah yaitu dalam menghitung jumlah buah jeruk. Kegiatan menghitung secara manual membutuhkan waktu yang lama apalagi penjualan dalam skala besar. Dari permasalahan tersebut penulis mengusulkan metode Hough Transform untuk sistem pendeteksi dan perhitungan buah jeruk. Pada penelitian ini menguraikan proses sistem menjadi beberapa tahapan, yaitu: Input citra, konversi citra grayscale, mengaplikasikan median filter, meregangkan kontras, menurunkan brightness, dan deteksi *hough transformation*. Total dataset yang digunakan pada penelitian ini berjumlah 70 dataset citra jeruk. Untuk citra yang digunakan, algoritma Transformasi Hough sangat berperan dalam mendeteksi lingkaran pada objek buah jeruk. Oleh karena itu, diperlukan data citra dengan latar belakang yang seragam guna meningkatkan kejelasan temuan identifikasi objek melingkar. Selain faktor *background* objek, nilai sensitivity juga mempengaruhi ketepatan saat mengidentifikasi lingkaran. Berdasarkan hasil dan pembahasan uji coba, sistem dapat mendeteksi dan menghitung jeruk cukup baik dengan akurasi mencapai 95,4%.

**Kata kunci:** Deteksi Citra Jeruk, Intensitas Citra, Transformasi Hough

*This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license*



## 1. Pendahuluan

Indonesia merupakan negara agraris penghasil bahan pangan dalam jumlah besar seperti sayuran dan buah-buahan. Salah satu jenis buah yang dihasilkan oleh Indonesia adalah buah jeruk. Buah jeruk merupakan buah yang paling banyak dikonsumsi oleh masyarakat karena mempunyai rasa yang menyegarkan. Tidak hanya itu, kandungan vitamin C pada buah jeruk cukup tinggi dan sangat baik bagi tubuh. Menurut survei yang dilakukan oleh SUSENAS, konsumsi buah jeruk masyarakat Indonesia mencapai 2,90 kg/tahun. Penggunaan buah jeruk sangat luas, baik sebagai bahan pangan langsung maupun sebagai bahan baku industri. Sehingga produksi buah jeruk sangat penting untuk memenuhi kebutuhan pasar.

Para petani buah jeruk membutuhkan manajemen produksi tiap waktu panennya. Hal ini tentu dinilai sangat penting untuk memperkirakan hasil buah di kebun jeruk. Perhitungan frekuensi hasil panen yang akurat dapat membantu petani untuk meningkatkan kualitas panen. Namun, hal ini masih dilakukan dengan cara manual yang tentu memerlukan banyak waktu serta tenaga kerja. Jika produksi buah jeruk menerapkan inovasi teknologi, hal ini dapat meningkatkan frekuensi dalam produksi buah jeruk sehingga memudahkan pekerjaan petani dan memenuhi kebutuhan pasar.

Seiring berkembangnya teknologi, banyak masalah yang dapat diselesaikan dengan pendekatan teknologi terutama menggunakan machine learning. Salah satu implementasinya yaitu terhadap bidang computer vision yakni, mendeteksi buah jeruk. Deteksi buah jeruk merupakan salah satu tugas menantang dalam machine learning dan computer vision karena buah jeruk memiliki berbagai ukuran dan tahapan. Permasalahan dalam menghitung jumlah buah jeruk yaitu kegiatan penting yang dilakukan untuk para pedagang dalam proses penjualan. Kegiatan menghitung secara manual membutuhkan waktu yang lama apalagi penjualan dalam skala besar. Tidak hanya membantu para pedagang, tetapi juga membantu petani dalam menentukan jumlah jeruk hanya dengan menggunakan citra yang diperoleh menggunakan kamera lalu diproses oleh komputer. Pemrosesan citra oleh komputer dengan melakukan perbaikan pencahayaan, perengangan kontras, dan deteksi objek.

Pengenalan objek pada citra digital, seperti deteksi bentuk dan deteksi angka merupakan teknologi yang banyak digunakan saat ini. Keakuratan

pendeteksian jumlah objek saat ini masih belum maksimal. Akurasi yang rendah disebabkan oleh posisi buah yang bertumpuk satu sama lain. Ada beberapa metode yang dapat digunakan untuk mendeteksi objek dalam *computer vision*, salah satunya adalah metode Hough Transform.

Beberapa penelitian sebelumnya mengenai pendeteksian buah, Hagai Raja Sinulingga[1] melakukan deteksi dan perhitungan jumlah jeruk matang dan non matang berdasarkan citra digital dengan memanfaatkan fitur deteksi tepi dan perbedaan warna (*color space*) antara jeruk dan lingkungannya. Citra yang digunakan merupakan citra buah jeruk berwarna oranye, dimana jika digunakan citra inputan buah jeruk berwarna hijau akan mempengaruhi jumlah deteksi buah jeruk pada sistem ini. Hal ini dikarenakan sistem memanfaatkan fitur perbedaan warna.

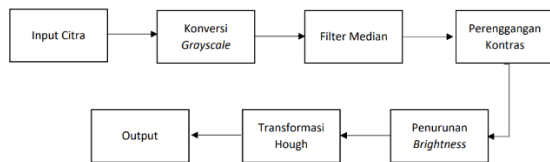
Penelitian terkait lainnya yang dilakukan oleh M Choirul Rahmadan[2], deteksi jumlah buah jeruk keprok menggunakan circular hough transform. Beberapa tahapan yang dilakukan yaitu preprocessing perbaikan kontras menggunakan metode CLAHE untuk menajamkan citra yang memerlukan parameter clipfactor, deteksi tepi menggunakan metode Canny, dan deteksi objek menggunakan metode CHT. Hasil penelitian ini hanya mendapatkan nilai akurasi 50%. K. Dhanalakshmi[3] juga melakukan penelitian yaitu penghitungan otomatis buah menggunakan circular hough transform untuk mengenali jeruk dan menghitung jumlah jeruk berdasarkan citra yang diinput. Hasil uji penelitian ini buah jeruk yang memiliki bentuk dan ukuran yang berbeda berdasarkan radius atau jari-jari lingkaran dapat diidentifikasi menggunakan metode CHT. Permasalahan yang dihadapi hanya pada citra buah jeruk yang diambil dari jarak jauh sehingga citra cenderung memiliki ukuran yang kecil. Selain itu, beberapa rentangnya tumpang tindih. Hal ini menyebabkan proses deteksi menjadi sulit.

Selanjutnya Xiao Changyi[4], melakukan deteksi apel dari citra pohon apel berdasarkan BP *neural network* dan *hough transform*. Sebelum dilakukan proses deteksi apel dilakukan proses training citra apel dan background images. Proses ini dilakukan untuk meningkatkan akurasi karena pengambilan citra pada sistem deteksi ini dilakukan langsung di pohon sehingga banyak tantangannya seperti apel yang tertutup oleh ranting, apel yang saling tumpang tindih dan sebagainya. Sistem ini mencapai koefisien korelasi R sebesar 0,985. Tetapi, sistem ini membutuhkan data yang besar karena dilakukan proses training dan juga algoritma yang

diusulkan sangat kompleks sehingga membutuhkan komputasi yang besar. Hasanah[5] sistem penghitung jeruk matang pada kebun berdasarkan hue, saturation dan chrominance-red menggunakan algoritme watershed berbasis raspberry pi. Namun, proses pengambilan citra dilakukan di ruang terbuka sehingga untuk meningkatkan akurasi sistem diperlukan tahapan operasi piksel yang dapat meningkatkan kualitas citra.

Berdasarkan penelitian sebelumnya, maka penulis membuat sistem yang dapat mendeteksi dan menghitung jumlah jeruk secara otomatis menggunakan metode circular hough transform dengan menambahkan beberapa tahapan untuk mendapatkan akurasi yang maksimal. Sistem ini bekerja dengan cara menginput citra yang telah diambil lalu dilakukan proses konversi ke grayscale kemudian dilakukan filter median untuk menghilangkan derau yang ada pada citra setelah itu dilakukan peregangkan kontras, menurunkan cahaya agar tepi pada jeruk lebih tajam dan dilakukan algoritma Hough Transform. Output sistem ini yaitu jumlah jeruk yang berhasil dideteksi oleh sistem

## 2. Metode



Gambar 1. Segmentasi Citra

Pada penelitian ini kami menguraikan proses sistem menjadi beberapa tahapan, yaitu: Input citra, konversi citra grayscale, mengaplikasikan median filter, meregangkan kontras, menurunkan brightness, dan deteksi *hough transformation*. Penjelasan mengenai keseluruhan proses akan dijelaskan di bawah ini.

### 2.1. Input Citra



Gambar 2. Citra Jeruk

Citra merupakan salah satu bentuk informasi yang diperlukan manusia selain teks, suara dan video [6]. Pada tahap pertama dalam penelitian ini ialah tahap penginputan citra. Pada tahap ini pengguna akan memasukkan citra jeruk yang akan dideteksi menggunakan sistem. Dataset yang digunakan penulis adalah dataset pribadi yang diambil sendiri menggunakan kamera penulis.

### 2.2. Konversi Citra Grayscale

Sesuai dengan namanya, citra jenis ini menangani gradasi warna hitam dan putih, yang tentu saja menghasilkan efek warna abu-abu. Pada jenis gambar ini, warna dinyatakan dengan intensitas. Dalam hal ini, intensitas berkisar antara 0 sampai dengan 255. Nilai 0 menyatakan hitam dan nilai 255 menyatakan putih. Dalam mengolah citra diperlukan untuk mengkonversi citra berwarna kedalam bentuk citra berskala keabuan (*grayscale*), mengingat banyak pemrosesan citra yang bekerja pada skala keabuan. Secara umum citra berwarna dapat dikonversikan ke citra *grayscale* melalui persamaan 1 berikut.

$$I = a \times R + b \times G + c \times B, a + b + c = 1 \quad (1)$$

R menyatakan nilai komponen merah, G menyatakan nilai komponen hijau, dan B menyatakan nilai komponen biru. Misalnya, sebuah piksel mempunyai channel R, G, B.



Gambar 3. Citra Grayscale

Setelah citra dimasukkan, tahap selanjutnya adalah mengkonversi citra yang awalnya RGB menjadi citra *grayscale*. Tahap ini dilakukan untuk menyederhanakan model citra sehingga bisa diproses melalui tahap berikutnya.

### 2.3. Filter Median

Dalam pengolahan citra, filter median sering digunakan. Median merupakan nilai tengah dari

kumpulan data. Nilai yang lebih baik digunakan untuk suatu piksel ditentukan oleh nilai median dari setiap piksel dan kedelapan piksel tetangga pada 8-ketetanggaan [7]. Secara matematis ditulis dengan persamaan 2 berikut.

$$g(y, x) = \text{median}(f(y - 1, x - 1), f(y - 1, x), f(y - 1, x + 1), f(y, x - 1), f(y, x), f(y, x + 1), f(y + 1, x - 1), f(y + 1, x), f(y + 1, x + 1)) \quad (2)$$

1	2	2
4	5	4
7	7	7

1, 2, 2, 4, 4, 5, 7, 7, 7

Gambar 4. Ilustrasi Operasi Filter Median

Pada gambar 4 di atas, terlihat bahwa untuk mendapatkan nilai median perlu dilakukan pengurutan (*sorting*) terlebih dulu sebelum mengambil nilai tengah atau median. Operasi filter median dapat diterapkan pada pengolahan citra yang dapat dilihat pada gambar 5 berikut.



Gambar 5. Citra Jeruk setelah Pengaplikasian Filter Median

Dengan metode median filter, maka citra grayscale dari tahap sebelumnya akan dikurangi *noise* dan citranya akan dihaluskan.

#### 2.4. Perenggangan Kontras

Peregangan kontras (*contrast stretching*) adalah salah satu metode untuk perbaikan kualitas citra. Metode ini diharapkan dapat menghasilkan citra yang baru yang lebih baik[8]. Kontras dalam suatu citra menyatakan distribusi warna terang dan warna

gelap. Suatu citra berskala keabuan dikatakan memiliki kontras rendah apabila distribusi warna cenderung pada jangkauan aras keabuan yang sempit. Sebaliknya, citra mempunyai kontras tinggi apabila jangkauan aras keabuan lebih terdistribusi secara melebar. Kontras dapat diukur berdasarkan perbedaan antara nilai intensitas tertinggi dan nilai intensitas terendah yang menyusun piksel-piksel dalam citra.

Agar distribusi intensitas piksel berubah perlu dilakukan peregangan kontras. Hal ini dilaksanakan dengan menggunakan persamaan 3 berikut.

$$g(y, x) = a f(y, x) \quad (3)$$

Berdasarkan rumus di atas, kontras akan naik kalau  $a > 1$  dan kontras akan turun kalau  $a < 1$ .



Gambar 6. Citra Jeruk setelah Perenggangan Kontras

Setelah *noise* pada citra dikurangi dan citra menjadi halus, tahap selanjutnya adalah meregangkan kontras atau bisa juga disebut sebagai *Contrast Stretching*. Meregangkan kontras citra adalah teknik perbaikan kualitas citra dengan meningkatkan kontras citra dengan cara meregangkan rentang nilai intensitas citra.

#### 2.5. Penurunan Brightness

*Brightness* adalah proses mengatur pencahayaan citra[9]. *Brightness* juga merupakan operasi dasar yang banyak dilakukan untuk meningkatkan kecerahan citra atau menurunkan kecerahan citra. Tujuan operasi ini untuk memperbaiki kualitas citra. Secara matematis, peningkatan kecerahan dilakukan dengan cara menambahkan suatu konstanta terhadap nilai seluruh piksel. Misalkan,  $f(y, x)$  menyatakan nilai piksel pada citra berskala keabuan pada koordinat  $(y, x)$ . Maka, citra akan dilakukan persamaan 4 berikut dan disimpan menjadi citra baru.

$$g(y, x) = f(y, x) - \beta \quad (4)$$



Setelah operasi dilakukan, nilai kecerahan semua pikselnya sebesar  $\beta$  terhadap citra asli  $f(y, x)$ . Apabila  $\beta$  berupa bilangan positif, kecerahan akan meningkat atau menjadi lebih terang. Hasil dari penerapan persamaan 4 diatas dapat dilihat pada gambar 7 berikut.



Gambar 7. Citra Jeruk setelah Penurunan *Brightness*

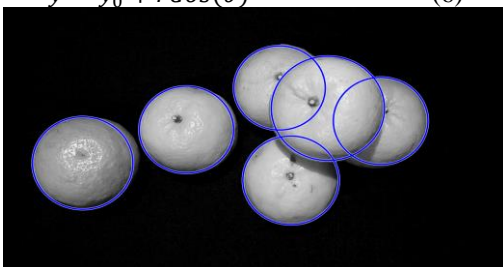
Tahap selanjutnya ialah menurunkan brightness citra. Pada tahap ini tingkat kecerahan citra akan diturunkan dengan cara mengurangi sebuah konstanta kepada atau dari setiap pixel di dalam citra.

## 2.6. Deteksi Transformasi Hough

*Hough Transform* merupakan salah satu metode image processing yang dapat digunakan untuk mendeteksi garis dan lingkaran pada suatu citra digital [10]. *Hough transform* sering digunakan untuk menentukan posisi bentuk lingkaran. Tujuan dari teknik ini adalah untuk menemukan contoh objek yang tidak sempurna dalam kelas tertentu. Penggunaan hough transform sebagai tool yang berfungsi untuk mengenal suatu pola dalam sebuah citra seperti lingkaran. Penerapan hough transform menggunakan persamaan yang mempunyai tiga buah parameter, yaitu:  $x_0$ ,  $y_0$ , dan  $r$ . Dimana  $x_0$  dan  $y_0$  merupakan pusat lingkaran.  $r$  adalah jari-jari lingkaran. Berikut persamaan 5 dan 6 yang menunjukkan bentuk parametrik.

$$x = x_0 + r \cos(\theta) \quad (5)$$

$$y = y_0 + r \sin(\theta) \quad (6)$$



Gambar 8. Citra Jeruk setelah dilakukan Transformasi Hough

Tahap terakhir adalah deteksi menggunakan transformasi hough. Transformasi Hough adalah teknik transformasi citra yang dapat digunakan untuk mengisolasi suatu objek pada citra dengan menemukan batas-batasnya (*boundary detection*).

## 3. Hasil dan Pembahasan

Hasil penelitian yang dibahas adalah hasil dari program yang telah dirancang dan dibuat untuk mendeteksi citra berekstensi \*.jpg, \*.jpeg dan \*.png. Pembahasan dilakukan mulai proses pra pengolahan citra, hingga proses deteksi objek yang dihasilkan melalui proses transformasi hough. Sistem diimplementasikan menggunakan MATLAB 2021a.

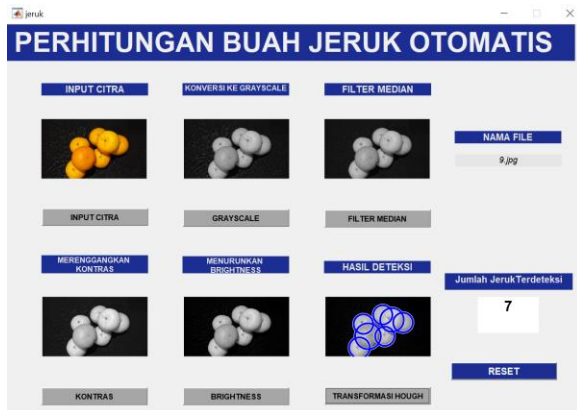
Sistem yang dibuat memiliki beberapa fitur, yaitu: mengubah citra ke citra grayscale, menurunkan brightness, meregangkan kontras, mengubah citra grayscale ke citra biner, menyaring citra menggunakan filter median, serta mendeteksi objek menggunakan transformasi hough. Penelitian ini menggunakan 70 dataset citra jeruk. Berikut ini tampilan Graphical User Interface (GUI) dari aplikasi deteksi jumlah jeruk yang telah dibuat dapat dilihat pada gambar 9.



Gambar 9. Tampilan GUI Awal Program

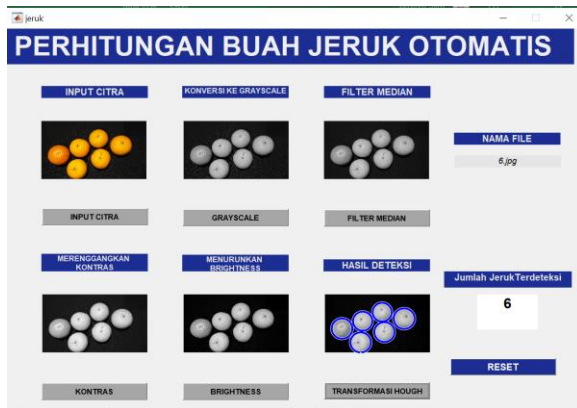
Cara penggunaan aplikasi deteksi jumlah jeruk dengan menginput citra jeruk, setelah itu menekan tombol konversi ke *grayscale* untuk mengubah citra RGB ke citra *grayscale*, selanjutnya menekan tombol filter median untuk memperhalus citra. Setelah citra sudah dihilangkan *noise*, maka langkah selanjutnya adalah menekan tombol menekan tombol meregangkan kontras agar intensitas citra lebih stabil. Tahap selanjutnya menurunkan *brightness* untuk menurunkan tingkat pencahayaan pada citra agar objek dapat terlihat jelas. Berikutnya, tahap terakhir yaitu menekan tombol transformasi hough

untuk mendeteksi jumlah jeruk yang ada pada citra, hasil deteksi jumlah jeruk akan tampil pada *text box* aplikasi. tombol reset pada aplikasi digunakan untuk mengubah citra yang ingin dideteksi. Berikut ini tampilan penggunaan aplikasi deteksi jumlah jeruk:

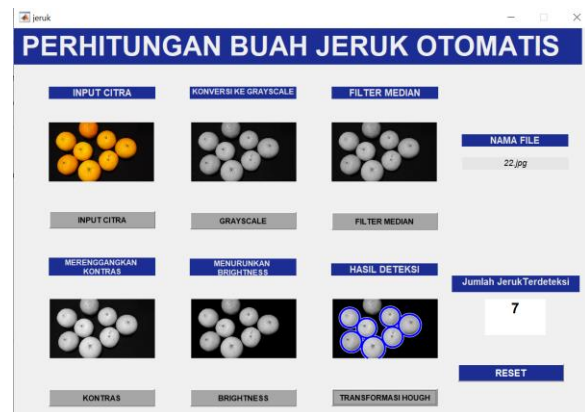


Gambar 10. Implementasi Sistem pada Citra Jeruk

Berikut ini akan diberi contoh tentang bagaimana cara menghitung jumlah buah jeruk dengan tingkat kesesuaian. Dimisalkan kita akan menghitung buah jeruk pada citra seperti yang tampak pada Gambar 10.



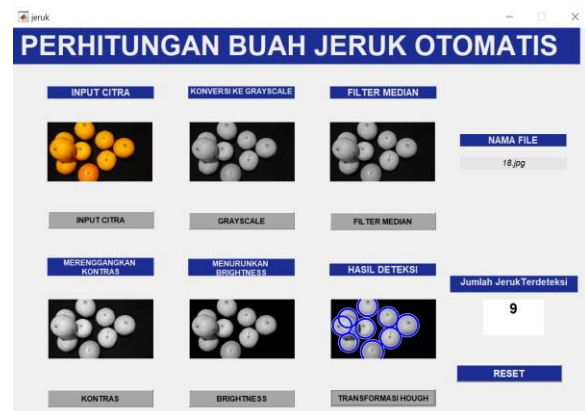
(a)



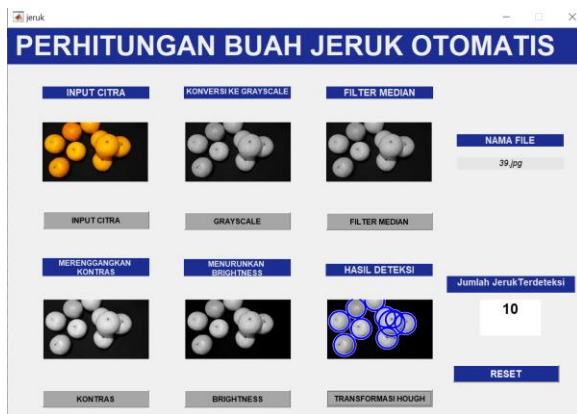
(b)

Gambar 11. Uji Sistem pada Citra Jeruk Tidak Bertumpuk

Pada gambar 11. menggunakan citra jeruk yang diletakkan secara sejajar atau tidak bertumpuk. Dapat dilihat bahwa pada gambar 11(a) jumlah jeruk yang dideteksi sama dengan jumlah jeruk yang asli, atau dapat dikatakan bahwa sistem berhasil menghitung jumlah jeruk. Akan tetapi pada gambar 11(b), sistem tidak mendeteksi jumlah jeruk pada citra dengan akurat. Padahal posisi kedua citra (11a dan 11b) sama-sama diletakkan secara sejajar atau tidak bertumpuk. Hal ini disebabkan oleh kemungkinan range jari-jari  $r$  pada buah jeruk di gambar 11(b) kurang memenuhi kondisi pada transformasi hough.



(a)



(b)

Gambar 12. Uji Sistem pada Citra Jeruk Bertumpuk

Pada kasus citra jeruk bertumpuk juga ditemukan perbedaan saat proses mendeteksi. Ketidaksesuaian saat mendeteksi buah jeruk mengakibatkan kesalahan dalam menghitung jumlah jeruk. Gambar 12 (b) memiliki jumlah buah jeruk yang lebih banyak dari jumlah aslinya. Terdapat kesalahan dalam mendeteksi lingkaran pada bagian yang bertumpuk. Faktor dari ketidaksesuaian ini disebabkan oleh nilai *sensitivity value* yang digunakan sebesar 0,99. Dengan memberikan nilai sensitivitas yang tinggi menjadikan peluang atau kemungkinan sistem mendeteksi lingkaran semakin besar.

Berdasarkan hasil percobaan yang dilakukan terhadap seluruh citra, evaluasi yang kami lakukan adalah menghitung Mean Squared Error (MSE) dan akurasi. Nilai MSE prediksi jumlah buah jeruk sebesar 2.4% yang diperoleh menggunakan persamaan (1). Sedangkan untuk nilai *accuracy* yang diperoleh sebesar 95,4%.

$$MSE = \frac{\sum_{t=1}^n (At - Ft)^2}{n} \quad (7)$$

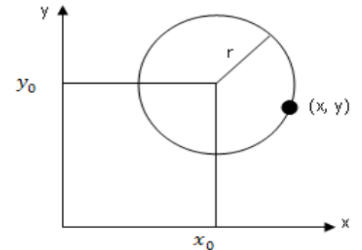
Metode menghasilkan nilai akurasi yang dapat dikatakan tinggi, Hal ini dikarenakan prosedur yang digunakan dalam mendeteksi lingkaran sama dengan transformasi hough pada objek garis, hanya saja dikerjakan pada ruang dimensi yang lebih kompleks. Dimana dalam parameter ruang 3 dimensi  $(x_0, y_0, r)$  dimana  $x_0$  dan  $y_0$  merupakan koordinat pusat lingkaran dan  $r$  adalah jari-jari lingkaran seperti persamaan (2) berikut.

$$(x - x_0)^2 + (y - y_0)^2 = r^2 \quad (8)$$

Sedangkan untuk vektor parameternya adalah seperti persamaan 9 berikut.

$$p \rightarrow = [x_0 \ y_0 \ r] \quad (9)$$

Dan ilustrasi dalam koordinat  $(x_0, y_0, r)$  adalah sebagai berikut.



Gambar 13. Ilustrasi Koordinat Parameter Lingkaran

Tabel 1 menggambarkan perbandingan hasil antara perhitungan manual dan perhitungan sistem menggunakan metode *circular hough transform*. Data diatas merupakan 10 data dari 70 citra yang diuji. Berdasarkan hasil perbandingannya, hasil akurasi yang didapatkan cukup baik.

Tabel 1. Perbandingan Hasil Perhitungan Manual dan Perhitungan Sistem

Percobaan	Perhitungan Manual	Perhitungan Sistem
Data 1	5	4
Data 2	6	6
Data 3	6	6
Data 4	6	6
Data 5	9	8
Data 6	8	9
Data 7	9	8
Data 8	9	9
Data 9	8	7
Data 10	8	7

#### 4. Kesimpulan

Berdasarkan hasil dan pembahasan uji coba, sistem dapat mendeteksi dan menghitung jeruk cukup baik dengan akurasi mencapai 95,4%. Untuk citra yang digunakan, algoritma Transformasi Hough sangat berperan dalam mendeteksi lingkaran pada objek buah jeruk. Oleh karena itu, diperlukan data citra

dengan latar belakang yang seragam guna meningkatkan kejelasan temuan identifikasi objek melingkar. Selain faktor *background* objek, nilai sensitivity juga mempengaruhi ketepatan saat mengidentifikasi lingkaran. Serta parameter yang merupakan jarak antar titik pusat objek dengan titik luar objek akan berpengaruh terhadap hasil deteksi jika kurang atau lebih dari range yang telah ditetapkan pada tahap transformasi hough. Beberapa faktor tersebut perlu perhatian yang lebih untuk meningkatkan efisiensi dalam menghitung jumlah objek buah jeruk.

### Daftar Pustaka

- [1] Sinulingga, H. R. (2019). Deteksi dan Perhitungan Jumlah Jeruk Matang dan Non Matang berdasarkan Citra Digital.
- [2] Rahmadan, M Choirul. (2016). Deteksi Banyak Buah Jeruk Keprok Dalam Citra Menggunakan Metode Circular Hough Transform. *Sarjana thesis*, Universitas Brawijaya.
- [3] K.Dhanalakshmi, K.Gowrishankar, dan N.Kalaiselvi. "Automatic Counting of Fruits Using Circle Hough Transform (CHT)", *MCAS Journal of Research*, vol-4, 2018.
- [4] Xiao C Y, Zheng L H, Li M Z, Chen Y, Mai C Y. *Apple detection from apple tree image based on BP neural network and Hough transform*. *Int J Agric & Biol Eng*, 2015; 8(6): 46–53.
- [5] Hasanah, S., Fitriyah, H., & Maulana, R. (2020). *Sistem Penghitung Jeruk Matang pada Kebun berdasarkan Hue, Saturation dan Chrominance-Red menggunakan Algoritme Watershed berbasis Raspberry Pi*. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, 3(11), 10848-10854. Diambil dari <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/6808>
- [6] Ratna, S. (2020). *Pengolahan Citra Digital Dan Histogram Dengan Phyton Dan Text Editor Phycharm*. *Technologia: Jurnal Ilmiah*, 11(3), 181-186.
- [7] I. G. A. Gunadi, "Analisis Perbandingan Metode Filter Mean, Median, Maximum, Minimum, Dan Gaussian Terhadap Reduksi Noise Gaussian, Salt&Papper, Speckle, Poisson, Dan Localvar". *Jurnal Ilmiah SINUS*, vol. 17, no. 1. p. 15, 2019, doi: 10.30646/sinus.v17i1.392.
- [8] Supiyanto, S., & Suparwati, T. (2021). *Perbaikan Citra Menggunakan Metode Contrast Stretching*. *Jurnal Siger Matematika*, 2(1), 13-18.
- [9] Azmi, Z., Pranata, A., Prayudha, J., & Phona, D. (2022). *Pengenalan Pola Rambu Lalu Lintas untuk Perancangan Smart Car Automation dengan Metode Kohonen*. *Sudo Jurnal Teknik Informatika*, 1(1), 34–41. <https://doi.org/10.56211/sudo.v1i1.7>
- [10] Sitio, A. S., & Sindar, A. (2020). *Sistem Identifikasi Biometrik Ekpresi Wajah Menggunakan Metode Transformasi Hough*. *Jurnal Nasional Komputasi dan Teknologi Informasi*, 3(3).
- [11] Rinanto, L., Sugiharto, A., & Indriyati, I. (2014). *Aplikasi Pendeteksi Objek Lingkaran pada Citra dengan Transformasi Hough*. *Journal of Informatics and Technology*, 2(4), 1–9.
- [12] G. Deng, "Double Lane Line Edge Detection Method Based on Constraint Conditions Hough Transform," in 17th International Symposium on Distributed Computing and Applications for Business Engineering and Science, Wuxi, 2018.
- [13] D. A. Prabowo, D. Abdullah, "Deteksi dan perhitungan objek berdasarkan warna menggunakan color object tracking," *Pseudocode*, vol. 5, no. 2, pp. 85–91, 2018.
- [14] E. Winata, H. Risna, and R. Angreni, "Identifikasi Jenis Bangun Datar dengan Algoritma Line Hough Transform dan Circular Hough Transform," *J. Tek. Inform. dan Sist. Inf.*, vol. 2, no. 2, pp. 120–129, 2016.
- [15] R. M. Putra, R. D. Puriyanto, K. Uad, and J. Ring, "Sistem Deteksi dan Pelacakan Bola dengan Metode Hough scircle Transform Menggunakan Kamera Omnidirectional pada Robot Sepak Bola Beroda," vol. 3, no. 3, pp. 176–184, 2021, doi: 10.12928/biste.v3i3.4786