# Penghitungan Jumlah Objek dalam Citra Menggunakan Metode Pengolahan Citra

Josep Andre Ginting - 13517108
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail: josepandregintings@gmail.com

Abstract—Sistem penghitung jumlah objek pada sebuah citra dapat sangat membantu untuk dapat mengetahui jumlah objek yang ada di dalam sebuah citra. Dengan adanya sistem penghitung objek, perhitungan dapat dilakukan secara otomatis dengan waktu yang jauh lebih singkat dibanding perhitungan secara manual. Sistem penghitung objek juga dapat mengurangi tingkat kesalahan perhitungan secara manual untuk menghitung jumlah objek kecil yang sangat banyak pada citra. Untuk dapat membedakan objek pada citra dengan background-nya, dilakukan metode pengolahan citra yaitu image segmentation, objek yang sudah dipisah dengan background-nya tersebut kemudian dihitung jumlahnya. Sistem penghitung jumlah objek dalam citra dengan metode pengolahan citra memiliki lima tahapan utama yaitu: 1) Akuisisi Citra 2) Perbaikan Citra 3) Segmentasi Citra 4) Analisis Citra dan 5) Penghitungan Objek.

Keywords—penghitungan, objek; object counting; image processing; pengolahan citra

#### I. Pendahuluan

Sistem penghitungan jumlah objek yang ada di dalam sebuah citra dapat sangat membantu untuk mengetahui jumlah objek-objek pada sebuah citra secara otomatis. Menghitung jumlah objek pada sebuah citra sebenarnya bukanlah suatu hal yang sulit dilakukan oleh seseorang, tapi hal itu akan menjadi masalah jika jumlah citra yang ingin dihitung jumlah objek di dalamnya sangat banyak, belum lagi jika pada citra tersebut terdapat objek-objek yang sangat banyak jumlahnya ataupun objek-objek yang kecil ukurannya, akan menjadi sangat sulit dalam menghitung jumlah objek di dalam citra tersebut. Sistem penghitungan jumlah objek pada citra tentu akan sangat membantu dalam meningkatkan tingkat akurasi dari perhitungan dan juga kecepatan waktu perhitungan

Saat ini, sudah banyak sistem penghitungan objek pada citra yang ada yang sebagian besar memanfaatkan metode pembelajaran mesin (*machine learning*). Namun terdapat satu kelemahan mendasar pada sistem penghitungan objek dengan metode pembelajaran mesin tersebut, yaitu sistem hanya dapat mendeteksi dan menghitung objek tertentu yang sudah dilatih pada model pembelajaran mesin tersebut. Misalnya jika kita melatih sebuah sistem pendeteksi dan penghitung dengan citra objek pensil, maka sistem hanya akan dapat mendeteksi dan menghitung objek pensil pada citra, sedangkan objek lainnya

tidak dapat dideteksi dan dihitung. Sehingga untuk dapat menghitung semua objek pada citra uji, maka setiap objek tersebut harus sudah pernah dilatih pada model terlebih dahulu.

Sedangkan pada sistem penghitung objek pada sebuah citra dengan menggunakan metode pengolahan citra memiliki kelebihan dibandingkan dengan sistem penghitung objek dengan metode pembelajaran mesin (machine learning). Pada metode pembelajaran mesin, objek yang dihitung jumlahnya harus sudah dilatih (train) terlebih dahulu untuk dapat dideteksi, sedangkan jika menggunakan metode pengolahan citra objek pada citra dapat dideteksi tanpa perlu melakukan pelatihan (train) terlebih dahulu. Pada metode pengolahan citra, sistem akan memisahkan semua objek yang ada dari latar belakangnya (background) kemudian menghitung semua objek selain latar belakang, sehingga semua objek dapat dikenali sebagai objek kemudian dihitung jumlahnya .Penggunaan metode pengolahan citra ini tentunya jauh lebih efektif dalam menghitung jumlah objek yang acak yang belum dibandingkan dengan dilatih sebelumnya metode pembelajaran mesin.

Dalam membuat sebuah sistem yang dapat melakukan penghitungan jumlah objek yang ada pada sebuah citra dibutuhkan metode-metode yang ada pada pengolahan citra, salah satunya adalah segmentasi citra (*image segmentation*). Teknik *Image segmentation* pada pengolahan citra dapat dimanfaatkan untuk memisahkan objek pada citra dari latar belakangnya (*background*-nya). Setelah memisahkan objek pada citra dari latar belakangnya, barulah dilakukan perhitungan pada objek-objek tersebut untuk mengetahui berapa banyak jumlahnya.

# II. LANDASAN TEORI

Pada bagian ini akan dibahas teori-teori pengolahan citra yang dipakai dalam implementasi solusi yang dirancang pada masalah ini.

# A. Citra Digital

Citra adalah sinyal dwimatra yang bersifat menerus (*continue*) yang dapat diamati oleh sistem visual manusia [1]. Citra adalah fungsi dwimatra yang menyatakan intensitas cahaya pada bidang dwimatra f(x, y), dengan:

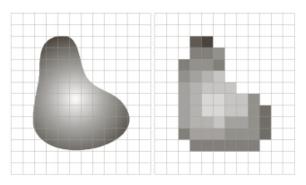
# (x, y): koordinat pada bidang dwimatra

f(x, y): intensitas cahaya (*brightness*) pada titik (, y)

Citra digital adalah representasi citra kontinu melalui pencuplikan (*sampling*) secara ruang dan waktu. Proses digitalisasi citra terbagi atas dua tahap, yaitu:

- Penerokan (sampling): yaitu digitalisasi secara spasial (x, y). bertujuan untuk menentukan seberapa banyak pixel yang diperlukan untuk merepresentasikan citra kontinu, dan bagaimana pengaturannya.
- Kuantisasi : yaitu mengangkakan nilai intensitas f(x, y) menjadi integer. Bertujuan untuk memetakan nilai dari sinyal kontinu menjadi G buah nilai diskrit (G buah level).

Dari kedua tahap tersebut, dapat dihasilkan sebuah citra digital yang berbentuk matriks berisi informasi warna dari citra asli. Gambar 2.1 adalah ilustrasi contoh hasil digitalisasi pada sebuah citra.



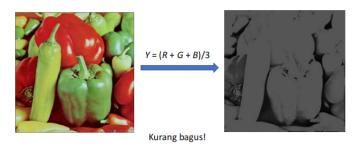
**Gambar 2.1.** Citra asli (kiri) dan citra hasil digitalisasi (kanan) (Sumber: Dr. George bebis, *Image Formation and Representation*, CS85/685 Computer Vision)

## B. Citra Grayscale

Citra grayscale merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap *pixel*-nya, artinya nilai dari *Red* = *Green* = *Blue*. Nilai-nilai tersebut digunakan untuk menunjukkan intensitas warna. Citra grayscale disimpan dalam format 8 bit untuk setiap *pixel*-nya, yang memungkinkan sebanyak 256 intensitas (0-255).

Untuk mengubah citra berwarna yang memiliki nilai matrik pada masing-masing komponen R, G, dan B menjadi citra grayscale dengan satu nilai kanal Y (*luminance*), hal yang paling sederhana dilakukan adalah mencari nilai rata-rata dari masing-masing komponen warna R, G, dan B: Y = (R + G + B)/3. Namun konversi citra berwarna menjadi citra grayscale dengan metode ini menghasilkan citra grayscale yang kurang baik. Selain mencari nilai Y dari rata-rata setiap komponen warna, ada metode lain yang dianggap dapat menghasilkan citra grayscale yang lebih baik, yaitu dengan nilai skala tertentu dari masing-masing komponen warna: Y = 0.299R + 0.587G + 0.144B.

Perbedaan hasil konversi citra berwarna dengan kedua metode yang sudah disebut sebelumnya dapat dilihat pada gambar 2.2 dan gambar 2.3.



Gambar 2.2. Konversi citra warna ke grayscale dengan rata-rata pada setiap komponen warna R, G dan B (Sumber: Slide Kuliah IF4073 Materi Operasi-operasi Dasar Pengolahan Citra oleh Rinaldi Munir)

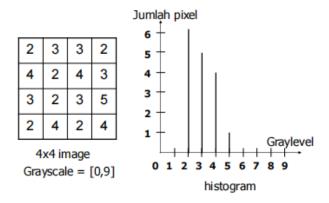


Gambar 2.3. Konversi citra warna ke grayscale dengan skala tertentu untuk setiap komponen warna R, G dan B (Sumber: Slide Kuliah IF4073 Materi Operasi-operasi Dasar Pengolahan Citra oleh Rinaldi Munir)

## C. Histogram Citra

Histogram citra (*image histogram*) merupakan informasi yang penting mengenai isi citra digital. Histogram citra adalah grafik yang menggambarkan penyebaran nilai-nilai intensitas *pixel* dari suatu citra atau bagian tertentu di dalam citra [2]. Dari sebuah histogram dapat diketahui frekuensi kemunculan nisbi (*relative*) dari intensitas pada citra tersebut. Histogram juga dapat menunjukkan banyak hal tentang kecerahan (*brightness*) dan kontras (*contrast*) dari sebuah gambar. Histogram merupakan alat bantu yang berharga dalam pekerjaan pengolahan citra baik secara kualitatif maupun kuantitatif.

Perhitungan histogram dapat dilakukan dengan mengiterasi *pixel-pixel* citra dan menghitung setiap kemunculan intensitas dari citra tersebut. Dari nilai jumlah kemunculan intensitas tersebut dapat digambarkan menjadi diagram *horizontal bar* sehingga dapat dianalisis dengan lebih mudah. Gambar 2.4 adalah contoh histogram dari suatu citra berukuran 4x4 dengan model warna *grayscale* [0,9].



Gambar 2.4. Ilustrasi citra (kiri) dan Histogram-nya (Kanan) (Sumber: ALI JAVED, Digital Image Processing, Chapter #3, Image Enhancement in Spatial Domain)

## D. Perbaikan Kualitas Citra (Image Enhancement)

*Image enhancement* (perbaikan kualitas citra) bertujuan untuk memperoleh citra yang lebih sesuai digunakan untuk aplikasi lebih lanjut. *Image enhancement* merupakan satu proses awal dalam pengolahan citra (*preprocessing*) [3].

Berdasarkan ranah (domain) operasinya, metode-metode untuk perbaikan kualitas citra dapat dikelompokkan menjadi dua kategori:

- 1. Image enhancement dalam ranah spasial
- 2. Image enhancement dalam ranah frekuensi

Metode-metode *image enhancement* dalam ranah spasial dilakukan dengan memanipulasi secara langsung *pixel-pixel* di dalam citra, sedangkan pada ranah frekuensi dilakukan dengan mengubah citra terlebih dahulu dari ranah spasial ke ranah frekuensi, baru kemudian memanipulasi nilai-nilai frekuensi tersebut.

Proses-proses yang termasuk ke dalam perbaikan kualitas citra diantaranya adalah pengubahan kecerahan gambar (*image brightening*), citra negatif (*image negatives*), peregangan kontras (*contrast stretching*), pengubahan histogram citra, pelembutan citra (*image smoothing*), penajaman (*sharpening*) tepi (edge), pewarnaan semu (*pseudocoloring*), pengubahan geometrik, dll.



Gambar 2.5. Citra asli (kiri), citra hasil peregangan kontras (tengah), dan citra hasil pengambangan (kanan) (Sumber: Slide Kuliah IF4073 Materi Image Enhancement oleh Rinaldi Munir)

Contoh perbaikan kualitas citra dengan peregangan kontras dan pengambangan dapat dilihat pada gambar 2.5.

## E. Segmentasi Citra (Image Segmentation)

Segmentasi citra (*image segmentation*) bertujuan untuk membagi citra menjadi region-region atau objek-objek atau dapat juga untuk memisahkan objek dengan latar belakang-nya. Tujuan dari segmentasi citra adalah menemukan bagian citra yang koheren atau objek spesifik [4]. Citra disegmentasi berdasarkan properti yang dipilih seperti kecerahan, warna, tekstur, dan sebagainya. Segmentasi membagi citra menjadi sejumlah region yang terhubung, tiap region bersifat homogen berdasarkan properti yang dipilih. Segmentasi citra merupakan tahapan melakukan *image/object recognition, image understanding*, dll.

Metode segmentasi citra umumnya dikelompokkan berdasarkan dua pendekatan:

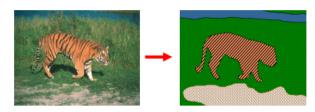
# 1. Diskontinuitas

Mempartisi citra berdasarkan perubahan nilai intensitas *pixel* yang cepat seperti tepi (*edge detection*).

## 2. Similarity

Mempartisi citra berdasarkan kemiripan area menurut properti yang ditentukan. Metode segmentasi citra yang termasuk ke dalam pendekatan ini adalah pengambangan (thresholding), region growing, split and merge, dan clustering.

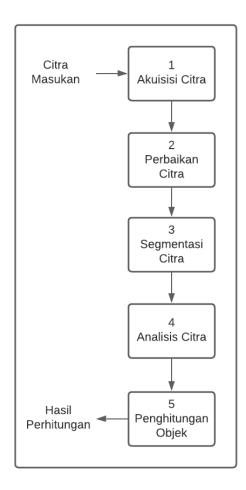
Gambar 2.6 adalah ilustrasi segmentasi pada citra berwarna.



Gambar 2.6. Citra asli (kiri) dan citra hasil segmentasi citra (kanan) (Sumber: Slide Kuliah IF4073 Materi Segmentasi Citra oleh Rinaldi Munir)

## III. IMPLEMENTASI SOLUSI

Secara umum, tahapan solusi yang diajukan untuk menyelesaikan masalah deteksi dan perhitungan objek pada citra digital terdiri dari lima tahapan utama yaitu 1) Akuisisi Citra 2) Perbaikan Citra 3) Segmentasi Citra 4) Analisis Citra dan 5) Penghitungan Objek. Secara lebih jelas kelima tahapan tersebut dapat dilihat pada gambar 3.1 berikut.



**Gambar 3.1.** Tahapan rancangan solusi penghitung objek

Pada solusi sistem penghitung objek dengan menggunakan metode pengolahan citra ini, terdapat beberapa batasan, diantaranya adalah:

- 1. Citra masukan merupakan citra berwarna
- Objek yang ingin dihitung harus saling terpisah. Objek yang saling bertumpuk akan dihitung menjadi satu objek yang sama.
- 3. Rasio ukuran objek pada citra tidak terlalu kecil, sehingga tidak dianggap sebagai *noise*.

Dalam mengimplementasikan solusi, penulis menggunakan kakas *python 3* sebagai bahasa pemrograman dengan bantuan *library OpenCV* dan *Numpy* untuk pengolahan citra, serta *library Matplotlib* untuk menampilkan citra. Setiap tahapan pada gambar 3.1 akan dijelaskan secara terpisah sebagai berikut.

#### A. Akuisisi Citra

Pada tahap ini, citra akan dibaca sehingga siap untuk diolah lebih lanjut. Pembacaan citra dilakukan menggunakan *library OpenCV.* Pada tahapan ini, diasumsikan citra masukan merupakan citra berwarna dengan format RGB, jika citra tidak

dalam format warna RGB, maka citra harus dikonversi terlebih dahulu ke format warna RGB untuk dapat diproses lebih lanjut.



**Gambar 3.2.** Citra masukan berupa citra lima buah coin (Sumber: https://www.mathworks.com/)

Citra yang ada pada gambar 3.2 berikutnya akan menjadi citra yang akan diproses untuk menampilkan hasil dari setiap tahap pada setiap tahapan proses implementasi solusi yang diajukan.

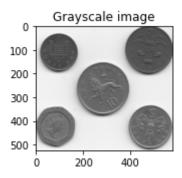
#### B. Perbaikan Citra

Pada tahap ini, citra akan diolah sedemikian rupa sehingga dihasilkan citra yang lebih sesuai untuk pengolahan/ proses berikutnya. Perbaikan yang dilakukan pada citra diantaranya adalah:

- 1. Konversi citra berwarna ke dalam bentuk *grayscale*: Tujuan mengubah citra masukan yang berwarna menjadi citra *grayscale* adalah agar citra dapat lebih mudah diproses dan dideteksi objek yang ada pada citra tersebut. Contoh hasil dari proses konversi ini dapat dilihat pada gambar 3.3.
- Penyesuaian kontras dengan menggunakan gamma correction:
   Menyesuaikan kontras pada citra sehingga

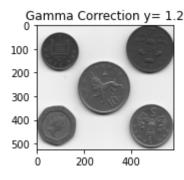
diharapkan dapat meningkatkan akurasi pendeteksian objek dan perhitungan objek. Contoh hasil dari proses konversi ini dapat dilihat pada gambar 3.4.

3. Penyesuaian kontras dengan *histogram equalization*. Contoh hasil dari proses konversi ini dapat dilihat pada gambar 3.5.



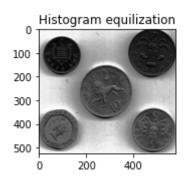
Gambar 3.3. Citra setelah diubah ke grayscale

Gambar 3.3 merupakan citra grayscale yang merupakan citra original yang sudah diproses dan sudah melewati tahap konversi citra berwarna menjadi citra grayscale.



**Gambar 3.4.** Citra setelah dilakukan penyesuaian kontras dengan *gamma correction* 

Gambar 3.3 yang merupakan citra yang contoh sudah melewati tahap konversi citra berwarna menjadi citra grayscale kemudian dilakukan penyesuaian kontras dengan menggunakan *gamma correction* dengan nilai y = 1.2 sehingga menjadi citra seperti gambar 3.4.



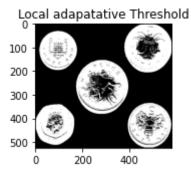
**Gambar 3.5.** Citra setelah dilakukan penyesuaian kontras dengan *histogram equalization* 

Gambar 3.5 merupakan citra keluaran dari citra 3.4 yang diproses dengan perbaikan citra *histogram equalization* untuk penyesuaian kontras.

# C. Segmentasi Citra

Pada tahap ini, dilakukan proses segmentasi pada citra dengan tujuan untuk memisahkan objek yang ada pada citra dari latar belakangnya (background). Segmentasi dilakukan dengan pendekatan similarity lebih tepatnya pengambangan (thresholding) dengan teknik adaptive thresholding dimana nilai ambang akan berubah secara dinamis tergantung pada perubahan pencahayaan di dalam citra. Hasil dari tahap ini adalah citra yang sudah dipisahkan antara objek dan latar belakangnya.

Objek pada citra akan berwarna putih, dan latar belakangnya akan berwarna hitam. Hasil pada tahap ini mungkin masih terdapat *noise* pada area objek akibat segmentasi yang tidak sempurna. Proses menghilangkan *noise* tersebut akan dilakukan pada tahap berikutnya.



**Gambar 3.6.** Citra yang telah dipisahkan dari latar belakangnya dengan metode *local adaptive thresholding* 

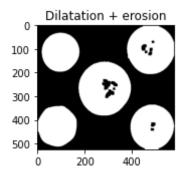
Citra yang telah melalui tahap perbaikan citra kemudian diproses ke tahap segmentasi citra untuk memisahkan objek dengan latar belakangnya. Gambar 3.6 merupakan citra hasil dari proses segmentasi, objek pada citra sudah terlihat terpisah dengan latar belakangnya, dimana latar belakang citra akan berwarna hitam, sedangkan objek berwarna putih.

# D. Analisis Citra

Pada tahap ini, akan dilakukan teknik *erosion* dan *dilatation* secara umum adalah untuk dapat mengurangi *noise* pada area objek (area berwarna putih) pada citra sehingga objek pada citra dapat terpisah satu sama lain dan dapat dianalisis dengan lebih baik..

Secara spesifik, tujuan dari dilakukannya teknik *erosion* adalah untuk memperkecil area objek (mengikis area objek) sehingga objek yang berdekatan dapat dipisah dengan lebih jelas. Sedangkan tujuan dilakukannya teknik *dilatation* adalah untuk melebarkan area objek pada citra sehingga objek pada citra menjadi lebih besar sehingga lebih mudah dianalisis.

Perbandingan perbedaan citra yang belum dan sudah diterapkannya teknik *dilatation* dan *erosion* dapat dilihat pada gambar 3.7.

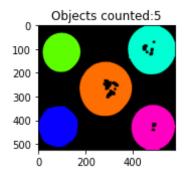


**Gambar 3.7.** Citra setelah diterapkan teknik *dilatation* dan *erosion* 

Gambar 3.7 merupakan citra yang sudah melewati tahap keempat yaitu analisis citra yaitu *dilatation* dan *erosion*, dimana objek pada citra akan terlihat lebih jelas dan terpisah dari latar belakangnya, serta *noise* dari citra juga berkurang.

## E. Penghitungan Objek

Pada tahap terakhir, akan dilakukan perhitungan jumlah objek yang ada pada citra, setiap satu wilayah/area yang saling terhubung akan mewakili satu objek. Sebelum menghitung jumlah wilayah/area yang ada, setiap wilayah/area akan diberi label terlebih dahulu dengan warna yang berbeda, kemudian dihitung jumlah wilayah/area tersebut sebagai representasi jumlah objek pada citra tersebut.



Gambar 3.8. Citra output beserta hasil perhitungan objek

Gambar 3.8 merupakan citra yang sudah dihitung jumlah objek yang ada didalamnya. Jumlah objek pada citra akan ditulis di bagian atas citra.

## IV. HASIL EKSPERIMEN

Eksperimen dilakukan pada beberapa citra berbeda yang didapat dari berbagai sumber di internet. hasil eksperimen menunjukkan bahwa solusi yang dibagun sudah dapat menyelesaikan masalah perhitungan objek pada citra digital. hanya saja terdapat kekurangan dari sistem, yaitu hanya dapat diterapkan pada citra dengan objek statis yang saling terpisah satu sama lain. Hal ini dikarenakan perhitungan objek

didasarkan pada wilayah/area yang dibentuk setelah melakukan segmentasi pada citra, sehingga objek yang saling terkait/bersentuhan atau menumpuk akan dianggap menjadi satu objek yang sama.

Untuk memudahkan mengetahui sistem sudah bekerja dengan baik atau belum, dibutuhkan sebuah parameter yang dapat mengukur akurasi dari sistem penghitung objek dengan metode pengolahan citra ini. karena kebutuhan tersebut dibuat cara perhitungan sederhana yang dapat menjadi acuan pengukuran akurasi perhitungan objek.

Perhitungan akurasi dihitung dengan cara jumlah objek sebenarnya pada citra dikurangi dengan nilai mutlak selisih hasil perhitungan sistem dengan jumlah objek sebenarnya dibagi dengan jumlah objek pada citra.

$$akurasi = \frac{N - |N - p|}{N} \tag{1}$$

atau

$$akurasi = 1 - |1 - \frac{p}{N}| \tag{2}$$

dengan:

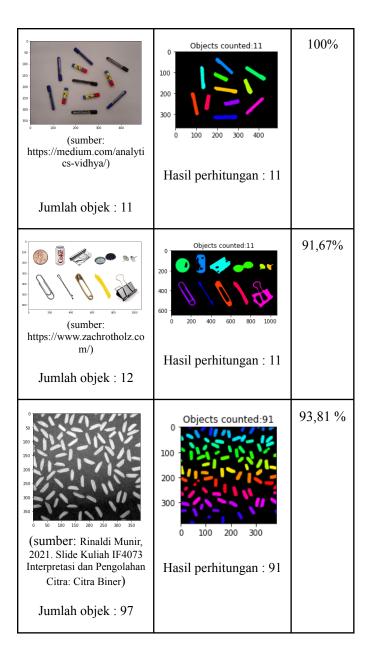
N = jumlah objek sebenarnya pada citra

p = hasil perhitungan objek dengan sistem

Hasil eksperimen dengan beberapa jenis citra yang dilakukan pada makalah ini dapat dilihat pada tabel 4.1 dibawah ini.

Tabel 4.1. Eksperimen dengan beberapa macam citra

Citra Asli dan Jumlah Objek	Hasil Perhitungan Objek	Akurasi
(sumber: Sumber: https://www.mathworks.com/)  Jumlah objek: 5	Objects counted:5  100 200 300 400 500 200 400  Hasil perhitungan: 5	100%



Dari percobaan yang dilakukan dapat disimpulkan bahwa rancangan solusi yang diajukan oleh penulis dapat menjadi solusi atas permasalahan perhitungan objek pada sebuah citra digital. Walaupun masih terdapat batasan-batasan dari jenis citra yang dapat dihitung jumlah objek-objek di dalamnya yaitu objek pada citra harus saling terpisah satu sama lain, namun sistem sudah dapat menghitung dengan akurasi yang cukup baik.

# V. KESIMPULAN DAN SARAN

Dari percobaan yang dilakukan dapat disimpulkan bahwa rancangan solusi yang diajukan oleh penulis dapat menjadi solusi atas permasalahan perhitungan objek pada sebuah citra digital. Walaupun masih terdapat batasan-batasan dari jenis citra yang dapat dihitung jumlah objek-objek di dalamnya yaitu citra input merupakan citra berwarna, objek pada citra harus saling terpisah satu sama lain (tidak saling menumpuk) dan rasio ukuran objek pada citra tidak terlalu kecil, namun sistem sudah dapat menghitung dengan akurasi yang cukup baik.

Kelebihan dari sistem perhitungan objek menggunakan metode pengolahan citra adalah dapat menghitung objek yang acak (semua jenis objek) dimana pada metode pembelajaran mesin, hanya objek yang sudah dilatih yang dapat dideteksi dan dihitung jumlahnya.

#### VI. PENGEMBANGAN SELANJUTNYA

Saran untuk pengembangan selanjutnya adalah sistem dapat dibangun dengan menggunakan GUI yang baik dan fitur dimana pengguna dapat melakukan pemilihan jenis perbaikan citra yang ingin diterapkan pada citra masukan, bukan hanya beberapa seperti yang masih diimplementasikan pada solusi di makalah ini. Dengan dapat memilih jenis teknik perbaikan citra yang ingin diterapkan, dapat membuat sistem menjadi lebih sesuai dan dapat mengatasi berbagai jenis citra. Selain pada pemilihan jenis teknik perbaikan citra (image enhancement) dapat juga menambahkan pemilihan parameter ukuran mask yang digunakan untuk dapat menyesuaikan dengan semua jenis ukuran objek pada citra, serta pemilihan jenis teknik yang ingin digunakan pada saat melakukan proses segmentasi citra. Dengan menambahkan fitur tersebut diharapkan dapat meningkatkan nilai akurasi karena setiap citra akan kompatibel dengan sistem penghitung objek ini.

Saran untuk pengembangan selanjutnya juga, sistem juga dapat diimplementasikan pada platform yang siap pakai langsung misalnya pada platform android maupun iOS sehingga lebih mudah dipakai di kehidupan nyata.

Selain itu, untuk penelitian yang lebih kompleks lagi dapat juga diselesaikan untuk batasan yang ada saat ini, yaitu dapat membedakan objek yang saling berkaitan/bersentuhan ataupun saling bertumpukkan sehingga dapat dianggap sebagai objek yang berbeda.

# UCAPAN TERIMA KASIH

Penulis mengucapkan syukur kepada Tuhan Yang Maha Esa karena atas hikmat, berkat dan rahmat-Nya penulis dapat menyelesaikan tugas ini dengan baik. Penulis juga berterima kasih kepada Bapak Dr. Ir. Rinaldi Munir, MT. selaku dosen pengampu mata kuliah IF4073 Interpretasi dan Pengolahan Citra yang selama satu semester ini telah membimbing saya dan mengajar saya tentang konsep-konsep penting dalam pengolahan citra. Serta tidak lupa penulis berterima kasih atas dukungan orang tua dan teman-teman seperjuangan di kelas maupun diluar kelas. Kiranya damai sejahtera selalu menyertai kita semua.

#### REFERENSI

- Rinaldi Munir, 2021. Slide Kuliah IF4073 Interpretasi dan Pengolahan Citra: Pembentukan Citra dan Digitalisasi Citra. https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2020-2021/03-Pembentukan-Citra-dan-Digitalisasi-Citra.pdf.
- [2] Rinaldi Munir, 2021. Slide Kuliah IF4073 Interpretasi dan Pengolahan Citra: Histogram Citra. https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2020-2021/06-Imag e-Histogram-2021.pdf.
- [3] Rinaldi Munir, 2021. Slide Kuliah IF4073 Interpretasi dan Pengolahan Citra: Image Enhancement (Bagian 1). https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2020-2021/08-Imag e-Enhancement-Bagian1.pdf.
- [4] Rinaldi Munir, 2021. Slide Kuliah IF4073 Interpretasi dan Pengolahan Citra: Segmentasi Citra (Image Segmentation). https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2020-2021/17-Segmentasi-Citra-2021.pdf.

- [5] Rinaldi Munir, 2021. Slide Kuliah IF4073 Interpretasi dan Pengolahan Citra: Operasi-operasi Dasar Pengolahan Citra. https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2020-2021/05-Oper asi-dasar-pengolahan-citra-2021.pdf.
- [6] Prem Kumar, V., Barath, V., & Prashanth, K. (2013). Object Counting and Density Calculation Using MATLAB.
- [7] Pornpanomchai, C., Stheitsthienchai, F., & Rattanachuen, S. (2008, May). Object detection and counting system. In 2008 Congress on Image and Signal Processing (Vol. 2, pp. 61-65). IEEE.
- [8] OpenCV Documentation. Disadur dari https://opencv-python-tutroals.readthedocs.io/en/latest/.

#### **PERNYATAAN**

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 25 Mei 2021

Josep Andre Ginting 13517108