

Aplikasi Transformasi *Watershed* untuk Segmentasi Objek Bersentuhan dan *Overlapping*

Girvin Junod 13519096 (*Author*)

Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jalan Ganesha 10 Bandung
E-mail (gmail): 13519096@std.stei.itb.ac.id

Abstrak—Segmentasi citra adalah proses untuk memisahkan citra menjadi bagian-bagian yang sering dilakukan untuk pendeteksian objek di dalam citra. Salah satu masalah dalam segmentasi objek dalam citra adalah jika objek yang ingin disegmentasikan itu bersentuhan atau tumpang tindih dengan objek lain. Seringkali ini membuat objek-objek yang bersentuhan atau tumpang tindih itu dideteksi menjadi satu objek. Pada makalah ini, dibahas tentang metode untuk melakukan segmentasi citra dengan objek bersentuhan menggunakan transformasi *watershed*.

Kata kunci—segmentasi citra; objek bersentuhan; objek *overlapping*; image processing; transformasi *watershed*

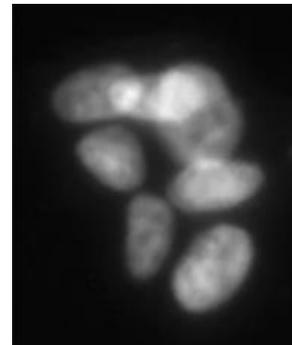
I. PENDAHULUAN

Segmentasi citra adalah suatu proses dalam *image processing*. Dalam segmentasi citra, dilakukan partisi suatu citra digital menjadi beberapa. Ini dilakukan untuk memisahkan suatu *region of interest*, yaitu bagian yang diinginkan dari citra, dari bagian-bagian lain. Ini juga dilakukan untuk dapat memisahkan objek-objek yang terdapat didalam citra menjadi segmen citra yang berbeda-beda.

Segmentasi citra memiliki banyak aplikasi. Salah satunya adalah dalam deteksi objek, misal dari suatu sistem CCTV atau dalam bidang *computer vision* secara umum. Pada aplikasi ini, segmentasi citra diperlukan agar dapat ditemukan bagian citra yang diinginkan dari citra secara keseluruhan agar citra menjadi lebih bermakna dan lebih mudah untuk dianalisis. Misal, ingin dilakukan analisis terhadap citra tulang dari suatu pasien. Agar lebih mudah untuk dilakukan, perlu dilakukan segmentasi citra terhadap citra tulang untuk memisahkan bagian citra yang berupa tulang dari bagian tubuh yang lain.

Salah satu masalah dari metode segmentasi citra konvensional adalah sulitnya untuk memisahkan objek yang bersentuhan atau tumpang tindih, seperti misal citra sel tubuh pada Gambar 1. Mengingat salah satu aplikasi segmentasi citra terbesar ada di bidang medis, maka masalah ini merupakan sesuatu yang signifikan.

Salah satu solusi yang dapat dipakai untuk menyelesaikan masalah segmentasi citra ini adalah menggunakan transformasi *watershed* untuk segmentasi citranya. Pada makalah ini, akan ditunjukkan penggunaan transformasi *watershed* untuk melakukan segmentasi objek yang bersentuhan atau tumpang tindih dalam citra. Pada makalah ini akan ditunjukkan juga aplikasi teknik ini dalam dunia nyata beserta contoh-contoh hasilnya.



Gambar 1. Citra sel yang bersentuhan dan tumpang tindih

(<https://www.nature.com/articles/srep32412>)

II. LANDASAN TEORI

A. Citra

Citra adalah gambar pada bidang 2-D atau dwimatra. Secara teknis, citra dapat dibidang merupakan suatu sinyal dwimatra yang bersifat kontinu dan dapat diamati oleh sistem visual manusia melalui mata. Sinyal dapat didefinisikan sebagai suatu fungsi dwimatra yang menyatakan nilai intensitas cahaya pada bidang dwimatra tersebut. Karena citra berada pada bidang 2-D, maka citra umumnya dinyatakan dalam koordinat x dan y yang diskrit pada fungsi $f(x,y)$.

Citra digital adalah citra yang didapatkan melalui pencuplikan secara ruang dan waktu. Ini dicapai dengan melakukan penerokan pada titik-titik tertentu pada citra dan

dilakukan kuantisasi untuk nilai intensitas cahayanya. Dalam citra digital, nilai intensitas cahaya ini direpresentasikan dengan nilai tingkat keabuan. Citra digital inilah yang biasa dijumpai ketika melihat gambar melalui perangkat seperti *handphone* atau laptop. Citra digital direpresentasikan dalam bentuk matriks $M \times N$ dengan setiap elemen matriks merepresentasikan pixel dalam citra beserta nilai tingkat keabuannya.

$$f(x, y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \dots & \dots & \dots & \dots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

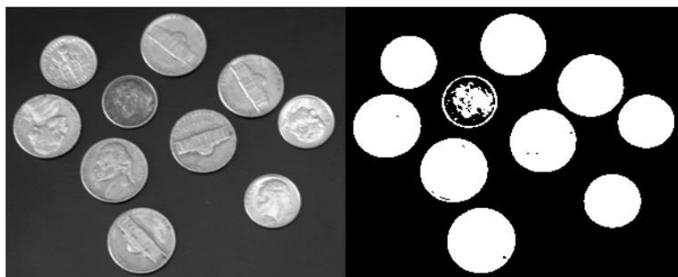
Gambar 2. Matriks untuk merepresentasikan citra digital

(<https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2022-2023/01-Pengantar-Pengolahan-Citra-Bag1-2022.pdf>)

Matriks seperti yang ditunjukkan pada Gambar 2 adalah untuk gambar *grayscale*. Untuk gambar berwarna, nilai setiap elemen dalam matriks biasanya akan berupa sebuah *array*. Pada sistem RGB, setiap elemen dalam matriks merupakan nilai tingkat keabuan untuk setiap warna dalam sistem RGB yaitu warna merah, hijau, dan biru.

B. Citra Biner

Citra biner adalah citra yang direpresentasikan dengan nilai biner. Dalam kata lain, setiap nilai tingkat keabuan dalam citra biner adalah 0 atau 1. Dalam citra biner, umumnya nilai 0 adalah untuk warna hitam dan nilai 1 untuk warna putih. Citra biner umumnya digunakan dalam segmentasi objek untuk memisahkan *foreground* dan *background* dengan cepat. Proses untuk menghasilkan citra biner dari citra berwarna sendiri sebenarnya juga merupakan suatu proses segmentasi citra. Ini dapat dicapai melalui pengambangan citra yang dapat dilakukan melalui metode seperti metode Otsu.



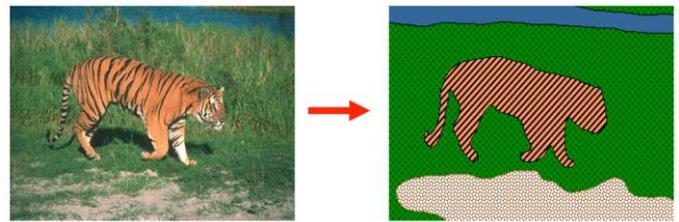
Gambar 3. Citra asli (kiri) dan citra biner (kanan)

(<https://www.mathworks.com/help/images/ref/imbinarize.html>)

C. Segmentasi Citra

Segmentasi citra adalah proses untuk membagi citra menjadi bagian-bagian. Segmentasi ini dapat dilakukan untuk membagi citra menjadi *region*, struktur linier, dan bentuk 2-D. Segmentasi ini dilakukan untuk membuat citra menjadi lebih bermakna untuk dianalisis atau untuk mempermudah pemrosesan dengan memisahkan citra dari bagian yang tidak

diperlukan. Segmentasi juga dapat dilakukan untuk melakukan pengelompokan objek di dalam citra berdasarkan karakteristiknya. Pada umumnya, metode untuk segmentasi objek dibagi menjadi berdasarkan diskontinuitas dan berdasarkan similaritas.



Gambar 4. Contoh segmentasi citra

(<https://ai.stanford.edu/~syyeung/cvweb/tutorial3.html>)

Segmentasi citra berdasarkan diskontinuitas mencoba untuk melihat di mana suatu bagian berhenti pada citra. Untuk mencapai ini, dicari bagian yang ada perubahan nilai intensitas pixel yang cepat seperti pada tepi. Oleh karena itu, metode segmentasi citra ini biasanya menggunakan metode berbasis deteksi tepi.

Segmentasi citra berdasarkan similaritas ditentukan berdasarkan kemiripan area. Pengambangan citra termasuk ke dalam metode segmentasi citra ini. Pengambangan citra membagi nilai intensitas pixel dalam gambar menjadi antara nilai 0 dan 1 berdasarkan nilai ambang T . Teknik pengambangan dapat dibagi menjadi:

1. Pengambangan global

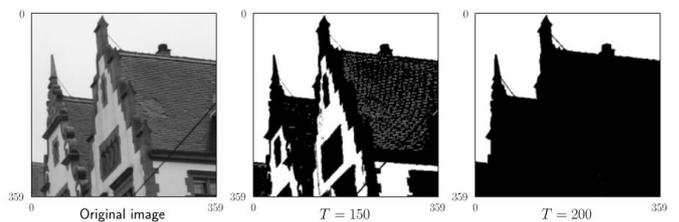
Nilai ambang T ditentukan dari nilai semua pixel dan digunakan untuk semua pixel.

2. Pengambangan lokal

Nilai ambang T ditentukan dari pixel yang memiliki hubungan ketertanggaan dan digunakan untuk sekelompok pixel saja.

3. Pengambangan adaptif

Nilai ambang T berubah secara dinamis tergantung dengan pencahayaan pada citra.



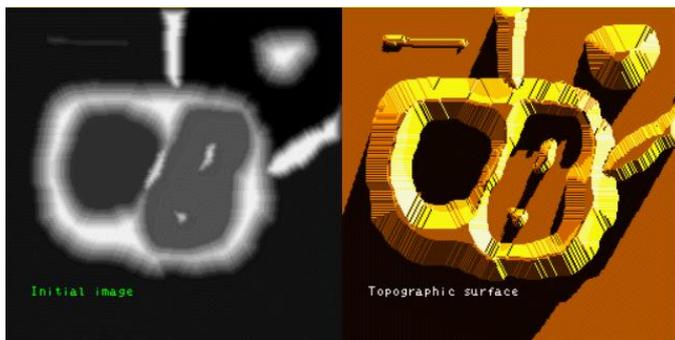
Gambar 5. Contoh pengambangan citra

(<https://vincmagnet.github.io/bip/segmentation/histogram.html>)

Transformasi *watershed* sendiri dapat dikategorikan sebagai segmentasi citra berdasarkan similaritas. Lebih tepatnya, transformasi *watershed* adalah metode yang mirip dengan pengambangan secara lokal.

D. Transformasi Watershed

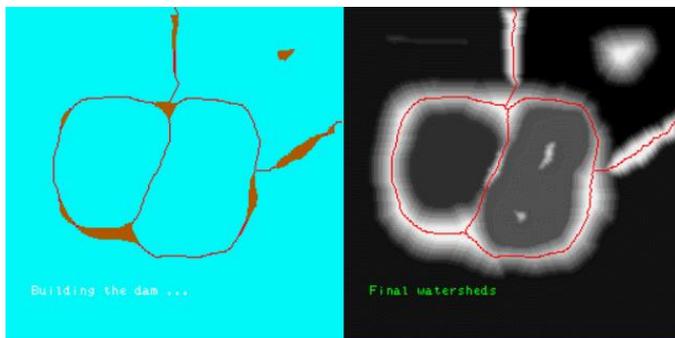
Transformasi *watershed* adalah suatu metode untuk segmentasi citra yang didasarkan pada konsep geografi yaitu *watershed* dan *catchment basin*. Kedua konsep ini membahas tentang struktur tanah dan mengalirnya air di alam. Sederhananya, sebuah *watershed line* adalah garis yang membatasi antar *catchment basins* di alam dan *catchment basin* sendiri adalah suatu daerah di mana air akan berkumpul akibat struktur geografinya. Pada intinya, transformasi *watershed* ingin menciptakan suatu gambar topografi dari gambar *grayscale* lalu “mengisi gambar tersebut” dengan air untuk membagi gambar menjadi *watershed line* dan *catchment basin*. Ini dicapai dengan membuat sehingga setiap nilai intensitas dalam pixel merepresentasikan tinggi tertentu pada gambar topografi. Pada Gambar 6, ditunjukkan hasil perubahan *watershed* pada suatu gambar jika misal daerah gelap dalam gambar menjadi rendah dan daerah terang menjadi tinggi di gambar topografi.



Gambar 6. Visualisasi konsep *watershed*

(<https://people.cmm.minesparis.psl.eu/users/beucher/wtshed.html>)

Setelah terbentuk gambar topografi, ada berbagai pendekatan untuk transformasi *watershed* ini. Pada umumnya, akan dibuat sehingga seakan-akan gambar topografi diisi dengan air lalu dibuat garis pembatas untuk memastikan air dari *catchment basin* berbeda tidak bercampur. Garis pembatas inilah yang akan menjadi *watershed line* akhir dan yang digunakan untuk segmentasi gambar. Pada Gambar 7, dapat dilihat hasil dari proses ini dan hasil akhir segmentasi gambar.

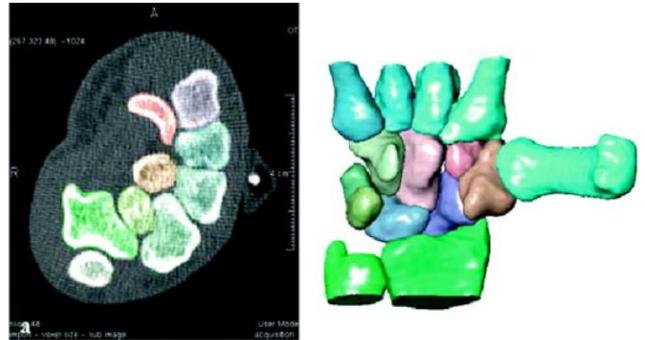


Gambar 7. Visualisasi proses transformasi *watershed* pada gambar

(<https://people.cmm.minesparis.psl.eu/users/beucher/wtshed.html>)

Secara teori, transformasi *watershed* ini akan membagi gambar berdasarkan similaritas nilai intensitas pixel. Namun, dalam penggunaannya, transformasi *watershed* sederhana cenderung menyebabkan segmentasi berlebihan akibat *noise* atau *local minimum* yang terlalu banyak. Oleh karena itu, telah dikembangkan banyak pendekatan terhadap transformasi *watershed* untuk mengatasi masalah ini.

Transformasi *watershed* banyak digunakan untuk segmentasi objek yang bersentuhan atau tumpang tindih, terutama di bidang medis seperti pada Gambar 8. Ini dikarenakan kemampuannya untuk segmentasi berdasarkan *region* dengan baik dan fleksibilitasnya untuk dimodifikasi sesuai dengan kebutuhan.



Gambar 8. Aplikasi Transformasi *watershed* pada bidang medis untuk segmentasi citra tulang

(<https://www.sciencedirect.com/topics/computer-science/watershed-transform>)

E. Distance Transform

Distance transform adalah suatu teknik *image processing* yang umumnya dilakukan terhadap citra biner. Pada teknik ini, citra digital diubah menjadi nilai jarak terhadap pixel terdekat dengan nilai bukan nol. Contoh perubahan *distance transform* adalah seperti pada Gambar 9.

1	1	0	0	0	0.00	0.00	1.00	2.00	3.00
1	1	0	0	0	0.00	0.00	1.00	2.00	3.00
0	0	0	0	0	1.00	1.00	1.41	2.00	2.24
0	0	0	0	0	1.41	1.00	1.00	1.00	1.41
0	1	1	1	0	1.00	0.00	0.00	0.00	1.00

Gambar 9. Contoh *distance transform* pada citra biner

(<https://www.mathworks.com/company/newsletters/articles/the-watershed-transform-strategies-for-image-segmentation.html>)

Distance transform sering digunakan bersamaan dengan transformasi *watershed*. Ini dikarenakan *distance transform* dapat dengan mudah membuat *catchment basin* dalam gambar

sesuai dengan objek yang ingin dideteksi, terutama jika objeknya bundar seperti sel tubuh.

III. METODOLOGI

Untuk mengimplementasikan aplikasi transformasi *watershed* dalam segmentasi objek bersentuhan atau tumpang tindih, telah dibuat suatu program yang menerima suatu gambar dan menunjukkan hasil segmentasi dari gambar tersebut. Program dibuat dengan MATLAB memanfaatkan fungsi MATLAB untuk transformasi *watershed*. Dalam implementasi ini, program dicoba terhadap beberapa contoh gambar yaitu gambar koin, sel, dan tumpukan jeruk.

Untuk setiap gambar, perlu dilakukan *preprocessing* sesuai dengan gambar. Pada umumnya, *preprocessing* yang dilakukan adalah mengubah gambar menjadi citra biner, menghilangkan *noise*, dan *distance transform*. Selain itu, untuk mencegah segmentasi berlebihan, dilakukan juga *filtering* untuk *local minimum* yang kecil.

Jadi, langkah umum untuk segmentasi objek dengan transformasi *watershed* adalah:

1. Melakukan konversi gambar ke *grayscale* lalu mengubahnya ke citra biner.
2. Menghilangkan *noise* gambar.
3. Melakukan *distance transform* jika sesuai dan diperlukan.
4. Melakukan *filtering* untuk *local minimum* yang kecil untuk mencegah segmentasi berlebihan.
5. Melakukan transformasi *watershed* untuk mendapatkan garis segmentasi.

IV. HASIL DAN PEMBAHASAN

Berikut adalah hasil aplikasi transformasi *watershed* untuk transformasi gambar untuk beberapa gambar.

A. Gambar Blobs

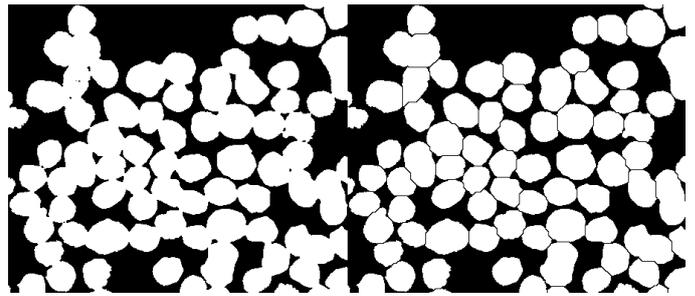
Pada bagian ini, dicoba untuk melakukan segmentasi citra pada gambar gumpalan-gumpalan putih. Gumpalan-gumpalan ini merupakan gambar uji coba karena mirip dengan citra biner untuk sel.



Gambar 10. Gambar *blobs* 1 (kiri) dan hasil segmentasi (kanan)

Pada Gambar 10, dapat dilihat bahwa untuk gambar dengan dua gumpalan yang saling bersentuhan, berhasil dilakukan segmentasi dua objek melalui transformasi *watershed*.

Distance transform berperan besar dalam pembuatan *catchment basin* untuk gambar seperti ini akibat terciptanya garis *watershed* antar gumpalan.

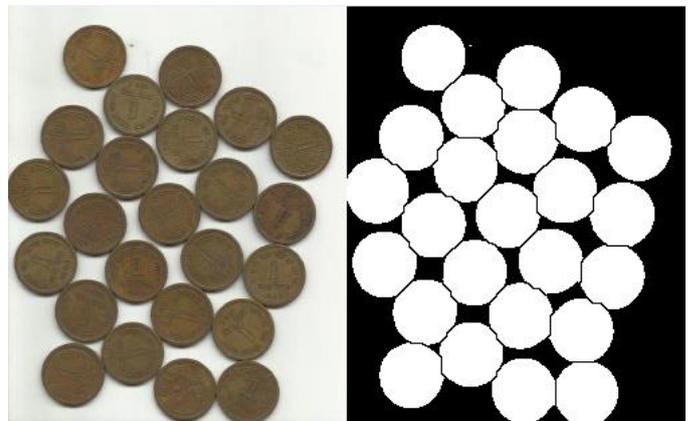


Gambar 11. Gambar *blobs* 2 (kiri) dan hasil segmentasi (kanan)

Pada Gambar 11, dapat dilihat bahwa gumpalan-gumpalan berhasil disegmentasi dengan baik meski gumpalan-gumpalan banyak yang bersentuhan atau tumpang tindih. Metode yang dipakai untuk gambar ini sama dengan gambar sebelumnya.

B. Gambar Koin

Pada gambar ini, terdapat koin-koin yang saling bersentuhan. Jika dilakukan segmentasi secara konvensional, seringnya koin-koin ini akan terdeteksi sebagai satu objek.

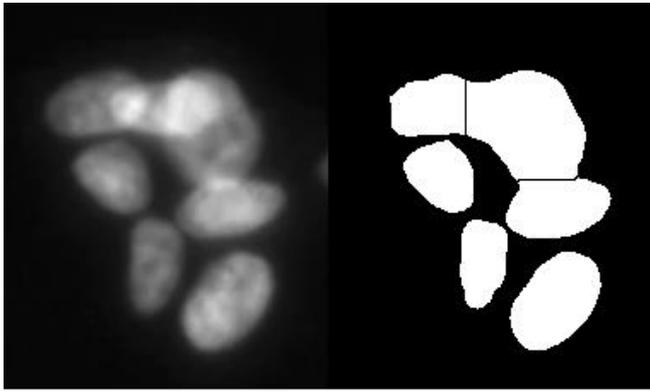


Gambar 12. Gambar koin (kiri) dan hasil segmentasi (kanan)

Pada Gambar 12, dapat dilihat bahwa setiap koin berhasil tersegmentasi dengan baik. Tercipta garis *watershed* yang memisahkan setiap koin. Ini berarti transformasi *watershed* telah berhasil digunakan untuk melakukan segmentasi objek yang bersentuhan ini.

C. Gambar Sel

Pada bagian ini, dicoba untuk melakukan segmentasi terhadap gambar sel yang saling tumpang tindih. Setiap sel individu dapat terlihat menyambung dengan sel lainnya dan dapat dibedakan berdasarkan strukturnya.

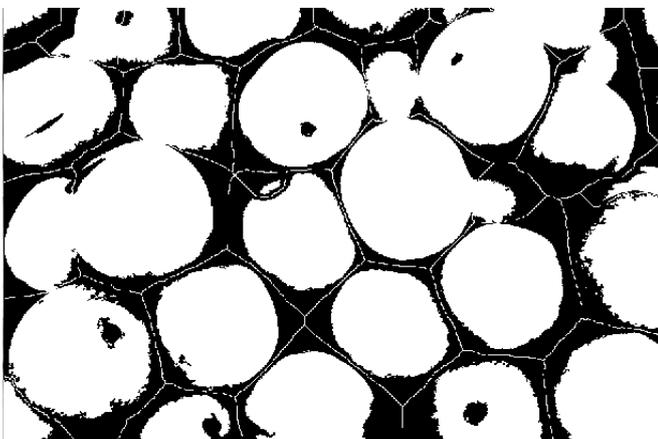


Gambar 13. Gambar sel (kiri) dan hasil segmentasi (kanan)

Pada Gambar 13, dapat dilihat bahwa untuk gambar sel ini telah berhasil dilakukan segmentasi dengan cukup baik. Dapat dilihat bahwa untuk sel-sel yang tumpang tindih di citra namun sebenarnya merupakan sel yang berbeda telah berhasil disegmentasikan.

D. Gambar Jeruk

Pada bagian ini, dicoba untuk melakukan segmentasi citra pada citra tumpukan jeruk. Citra ini jauh lebih kompleks dibanding citra sebelumnya akibat banyaknya jeruk yang tersembunyi dan hanya muncul sebagian serta perbedaan pencahayaan untuk jeruk-jeruk tersebut.



Gambar 14. Gambar jeruk (atas) dan hasil segmentasi (bawah)

Pada Gambar 14, dapat dilihat bahwa segmentasi berhasil dilakukan dengan hasil yang cukup baik. Segmentasi untuk gambar jeruk ini dilakukan dengan *preprocessing* yang kurang baik sehingga dapat dilihat masih terdapat banyak *noise* dalam gambar. Itu adalah bagian yang dapat diperbaiki untuk meningkatkan performa segmentasi dari transformasi *watershed* untuk gambar ini.

V. KESIMPULAN DAN SARAN

Dari eksperimen yang telah dilakukan di atas, dapat dilihat bahwa segmentasi citra dengan transformasi *watershed* telah menghasilkan hasil yang cukup baik untuk citra-citra yang bersentuhan atau tumpang tindih.

Ada banyak hal yang bisa dikembangkan dari implementasi yang telah dilakukan. Salah satunya adalah dalam *preprocessing image* sebelum dimasukkan ke transformasi *watershed*. Transformasi *watershed* bekerja baik dengan *noise* yang rendah.

Teknik *watershed* yang dipakai juga dapat dikembangkan lagi. Dalam makalah ini, digunakan transformasi *watershed* konvensional sehingga memerlukan cara seperti *distance transform* untuk mempermudah melakukan segmentasi. Terdapat teknik *watershed* lain seperti *marker-based watershed* yang dapat dipakai untuk melakukan segmentasi citra dengan lebih baik dan konsisten. *Marker-based watershed* adalah transformasi *watershed* interaktif di mana dapat dilakukan pemilihan label *region* secara manual dalam algoritma *watershed*. Ini memungkinkan untuk dilakukannya pendefinisian objek yang ingin dideteksi. Ada banyak variasi algoritma *watershed* yang telah dikembangkan di dunia untuk berbagai *task*. Transformasi *watershed* ini telah dipakai dalam pendeteksian tulang, pendeteksian jalan, dan lain-lain.

PRANALA PROGRAM

Kode program untuk implementasi beserta hasil seperti yang telah ditunjukkan di atas dapat ditemukan pada pranala di bawah ini:

<https://github.com/girvinjunod/Watershed-Transformation-Experiment>

PRANALA VIDEO YOUTUBE

Pranala untuk video Youtube tentang penjelasan dan demo solusi adalah sebagai berikut:

<https://youtu.be/mfzaWKxSarM>

UCAPAN TERIMA KASIH

Penulis mengucapkan syukur kepada Tuhan Yang Maha Esa karena atas berkat-Nya penulis dapat mengikuti mata kuliah IF4073 Interpretasi dan Pengolahan Citra ini dari awal sampai pembuatan makalah ini. Penulis menyampaikan terima kasih juga kepada Bapak Rinaldi Munir sebagai dosen pengampu mata kuliah ini yang telah mengajarkan kepada penulis materi yang diperlukan dalam pembuatan makalah ini.

REFERENSI

- [1] Munir, Rinaldi, <https://informatika.stei.itb.ac.id/~rinaldi.munir/Citra/2022-2023/citra2-2-23.htm>, diakses pada 19 Desember 2022.
- [2] Eddins, Steve, Watershed transform question from tech support, <https://blogs.mathworks.com/steve/2013/11/19/watershed-transform-question-from-tech-support/>, diakses pada 19 Desember 2022.
- [3] Eddins, Steve, Cell segmentation, <https://blogs.mathworks.com/steve/2006/06/02/cell-segmentation/>, diakses pada 19 Desember 2022.
- [4] Beucher, Serge, <https://people.cmm.minesparis.psl.eu/users/beucher/wtshed.html>, diakses pada 19 Desember 2022.
- [5] OpenCV Documentation, https://docs.opencv.org/4.x/d3/db4/tutorial_py_watershed.html, diakses pada 19 Desember 2022.
- [6] Mathworks Documentation, <https://www.mathworks.com/help/images/marker-controlled-watershed-segmentation.html>, diakses pada 19 Desember 2022.

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, 19 Desember 2022



Girvin Junod 13519096