TESIS – KI142502

SEGMENTASI CITRA MULTI-OBJEK MENGGUNAKAN *TRANSITION REGION* DAN K-MEANS *CLUSTERING*

Ahmad Wahyu Rosyadi
5116201003

DOSEN PEMBIMBING
Dr. Eng. Nanik Suciati, S.Kom., M.Kom.

PROGRAM MAGISTER
BIDANG KEAHLIAN KOMPUTASI CERDAS DAN VISI
DEPARTEMEN INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI DAN KOMUNIKASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2018

[Halaman ini sengaja dikosongkan]

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M.Kom.)
di
Institut Teknologi Sepuluh Nopember Surabaya

oleh:
Ahmad Wahyu Rosyadi
NRP. 5116201003

Dengan judul :
Segmentasi Citra Multi-Objek Menggunakan Transition Region dan K-Means Clustering

Tanggal Ujian : 24 Juli 2018
Periode Wisuda : September 2018

Disetujui oleh:

Dr. Eng. Nanik Suciati, S.Kom., M.Kom.
NIP. 197104281994122001 ..... (Pembimbing 1)

Prof. Ir. Handayani Tjandrasa, M.Sc., Ph.D.
NIP. 194908231976032001 ..... (Penguji 1)

Dr. Agus Zainal Arifin, S.Kom., M.Kom.
NIP. 197208091995121001 ..... (Penguji 2)

Dr.Eng. Darlis Herumurti, S.Kom., M.Kom.
NIP. 197712172003121001 ..... (Penguji 3)

Dekan Fakultas Teknologi Informasi dan Komunikasi,

Dr. Agus Zainal Arifin, S.Kom., M.Kom.
NIP. 197208091995121001

[Halaman ini sengaja dikosongkan]

Segmentasi Citra Multi-Objek Menggunakan Transition Region dan K-Means Clustering

Nama Mahasiswa : Ahmad Wahyu Rosyadi

NRP : 5116201003

Pembimbing : Dr. Eng.Nanik Suciati, S.Kom.,M.Kom.

ABSTRAK

Berbagai macam metode untuk segmentasi citra telah dikembangkan, salah satunya dengan menggunakan *transition region*. Segmentasi citra dengan *transition region* yang dikombinasikan dengan operasi morfologi terbukti mampu untuk mensegmentasi citra yang mengandung objek tunggal ataupun banyak. Namun, metode ini memiliki dua kelemahan. Pertama, metode ini hanya menggunakan batasan jarak untuk proses *edge linking* yang mungkin akan menyebabkan salah penyambungan antara *transition region* di area objek dan *backgroud*. Kedua, metode ini kemungkinan masih membangkitkan *transition region* pada area *background* apabila citra *input* memiliki tingkat keabuan dengan variasi yang tinggi. Untuk mengatasi permasalahan tersebut, diusulkan sebuah sistem untuk segmentasi citra multi-objek menggunakan *transition region* dan k-means clustering.

Pada sistem ini citra *grayscale* digunakan sebagai citra *input* yang akan disegmentasi. Citra tersebut kemudian disederhanakan tingkat keabuannya dengan menggunakan *adaptive* k-means *clustering.* Citra *grayscale* akan digunakan oleh proses perbaikan citra dan ekstraksi *transition region* citra asli, sedangkan citra sederhana digunakan untuk proses ekstraksi *transition region* citra sederhana. Selanjutnya akan dilakukan proses *edge linking* berbasis A* pada *transition region* citra asli berdasarkan *transition region* citra sederhana. *Region filling* dilakukan pada *transition region* hasil dari proses *edge linking* untuk menghasilkan area yang dianggap sebagai objek. Kemudian piksel-piksel yang berada pada area *transition region* ditampilkan. Dan hasil dari sistem ini adalah sebuah citra segmentasi yang berisi objek tunggal atau banyak.

Untuk mengukur kinerja dari metode ini, maka nilai *misclassification error* (ME), *false positive rate* (FPR), dan *false negative rate* (FNR) digunakan berdasarkan citra *ground truth*. Berdasarkan hasil pengujian pada 128 citra**,** metode ini mampu menghasilkan rata-rata nilai ME sebesar 9.29%, FPR 9.57%, dan FNR 9.80% menandakan bahwa metode ini mampu melakukan segmentasi objek tunggal maupun banyak dengan cukup baik.

Kata kunci: *Transition region, adaptive k-means clustering, median filter, edge linking, segmentasi citra, A*.*

[Halaman ini sengaja dikosongkan]

Multi-Object Image Segmentation Using Transition Region and K-Means Clustering

Student Name : Ahmad Wahyu Rosyadi

NRP : 5116201003

Supervisor : Dr. Eng.Nanik Suciati, S.Kom.,M.Kom.

ABSTRACT

Various methods for image segmentation have been developed, one of them is by using transition region. Combination of transition region and morphological operation can segment images containing single or multiple objects. However, this method has two weaknesses. First, this method only uses the distance constraint for the edge linking process which may cause a wrong connection between the transition region in the object area and backgroud. Second, this method may still generate transition region in the background area if the variance of gray level is quite high. To overcome this problem, a system for multi-object image segmentation using transition region and k-means clustering is proposed.

Grayscale image is used as the input image to be segmented. It's gray level is then simplified by using adaptive k-means clustering. The grayscale image will be used to image refinement process and original transition region extraction, while the simplified image is used to extract simple transition region. The A* edge linking process will be conducted on the original transition region based on the simple transition region. Region filling is performed to obtain area that is considered an object. Then, the pixels in the transition region area are displayed. And the result of this system is a segmented image that contains single or multiple objects.

Misclassification error (ME), *false positive rate* (FPR), dan *false negative rate* (FNR) is used to measure the performance of this method. The results of segmentation process on 128 images show that this method is able to segment the image containing single or multiple objects with ME, FPR, and FNR values are 9.29%, 9.57%, and 9.80%.

Keywords: *Transition region, adaptive k-means clustering, median filter, edge linking, segmentasi citra, A**.

[Halaman ini sengaja dikosongkan]

KATA PENGANTAR

Assalamualaikum warohmatullohi wabarokatuh.

Segala puja dan puji syukur penulis panjatkan kepada Allah SWT atas berkat, rahmat, taufik dan hidayah-Nya sehingga penyusunan dan penulisan Tesis yang berjudul “Klasifikasi Multilabel Motif Citra Batik Menggunakan Boosted Random Ferns Dengan Ekstraksi Fitur Histogram of Oriented Gradient” dapat terselesaikan dengan baik.

Shalawat serta salam semoga tercurah terhadap junjungan Nabi Besar Muhammad saw. Beliaulah yang membimbing umat Islam dari gelapnya kekufuran menuju cahaya Islam yang terang benderang.

Dengan segala kerendahan hati, penulis menyadari keterbatasan pengetahuan yang penulis miliki, karena itu tanpa keterlibatan dan sumbangsih dari berbagai pihak, sulit bagi penulis untuk menyelesaikan tesis ini. Oleh karena itu, dalam kesempatan ini penulis ingin mengcapkan terima kasih tak terhingga kepada :

1. Allah SWT atas limpahan rahmat-Nya sehingga penulis dapat menyelesaikan tesis ini dengan baik.
2. Ibu Hj. Aminah Ulfah, Bapak H. Iskandar dan Kakek H. Muhammad, selaku orang tua dan kakek penulis yang senantiasa mendoakan agar penulis selalu diberikan kelancaran dan kemudahan dalam menyelesaikan tesis ini. Serta menjadi motivasi terbesar untuk mendapatkan hasil yang terbaik.
3. Ibu Dr. Eng. Nanik Suciati, S.Kom, M.Kom, selaku dosen pembimbing yang senantiasa membimbing penulis dengan penuh kesabaran dengan tetap memberikan kepercayaan, motivasi, nasehat, perhatian dan semua bantuan sehingga penulis berhasil menyelesaikan tesis ini.
4. Ibu Prof. Ir. Handayani Tjandrasa, M.Sc., Ph.D., Bapak Dr. Agus Zainal Arifin, S.Kom., M.Kom., dan Bapak Dr.Eng. Darlis Herumurti, S.Kom., M.Kom. selaku dosen penguji yang telah memberikan bimbingan, saran, arahan, dan koreksi dalam pengerjaan tesis ini.

5. Bapak Waskitho Wibisono, S.Kom., M.Eng., PhD selaku ketua program pascasarjana Teknik Informatika ITS dan segenap dosen Teknik Informatika yang telah dengan ikhlas memberikan ilmu-ilmu beliau.
6. Mbak Lina, Mas Kunto dan segenap staf Tata Usaha yang telah memberikan segala bantuan dan kemudahan kepada penulis selama menjalani kuliah di Teknik Informatika ITS.
7. Seluruh keluarga besar yang senantiasa memberi dorongan moril dan menjadi inspirasi penulis.
8. Rekan-rekan angkatan 2016 Pasca Sarjana Teknik Informatika ITS dan teman-teman penulis yang telah menemani dan memberikan bantuan serta motivasi untuk segera menyelesaikan tesis ini.
9. Juga tidak lupa kepada semua pihak yang belum sempat disebutkan yang telah membantu terselesaikannya tesis ini.

Hanya ucapan terima kasih dan doa tulus penulis berikan atas apa yang telah mereka berikan, semoga apa yang telah mereka lakukan dapat memberikan kebaikan kembali kepada mereka. Aamiin yaa robbal ‘aalamiin.

Sebagai manusia biasa, penulis menyadari bahwa tesis ini masih jauh dari kesempurnaan dan memiliki banyak kekurangan. Sehingga dengan segala kerendahan hati, penulis mengharapkan saran dan kritik yang membangun dari pembaca. Wassalamualaikum warohmatullohi wabarokatuh.

Gresik, 26 Juni 2018

Penulis

DAFTAR ISI

ABSTRAK v
ABSTRACT vii
KATA PENGANTAR ix
DAFTAR ISI xi
DAFTAR GAMBAR xiii
DAFTAR TABEL xv
BAB 1 PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Perumusan Masalah 5
1.3 Tujuan 6
1.4 Manfaat 6
1.5 Kontribusi Penelitian 6
1.6 Batasan Masalah 7
BAB 2 KAJIAN PUSTAKA 9
2.1. *Transition Region* 9
2.2. K-means *Clustering* 11
2.3. *Probability Density Estimate Function* 12
2.4. Metode A* 13
2.5. *Thinning* (penipisan) 13
2.6. *Region Filling* 13
2.7. *Median Filter* 13
2.8. *Misclassification Error* (ME) 15
2.9. *False Positive Rate* (FPR) 15
2.10. *False Negative Rate* (FNR) 15
BAB 3 METODOLOGI PENELITIAN 17
3.1. Studi Literatur 17
3.2. Desain Sistem 18
3.2.1. Ekstraksi *Transition Region* 19
3.2.2. Perbaikan Citra 20
3.2.3. Pengelompokan Piksel-Piksel Citra dengan K-means *Clustering* 23
3.2.4. *Edge Linking* pada Transition Region 26
3.2.5. *Region Filling* Transition Region 32

3.2.6. Proses Pemisahan Objek dan *Background* 32
3.3. Skenario Pengujian 33
3.4. Dokumentasi dan Jadwal Penelitian 34
BAB 4 HASIL DAN PEMBAHASAN 35
4.1. Lingkungan Pengujian dan Dataset 35
4.2. Hasil Pengujian dan Pembahasan 36
4.2.1. Hasil Pengujian Keperluan Menggunakan *Filtering* 36
4.2.2. Hasil Pengujian Pencarian Nilai Parameter Yang Cocok..... 41
4.2.3. Hasil Pengujian dan Pembahasan Penghapusan *Transition Region* Citra Asli yang Sedikit Beririsan 50
4.2.4. Hasil Pengujian dan Pembahasan Proses *Edge Linking* 51
4.2.5. Pembahasan Hasil dari Sistem Secara Umum 52
BAB 5 PENUTUP 63
5.1. Kesimpulan 63
5.2. Saran 64
DAFTAR PUSTAKA 65
Lampiran A. Hasil Uji Coba 67
Lampiran B. Citra Hasil Segmentasi 81
Biografi Penulis 89

DAFTAR GAMBAR

gambar 2.1 Citra *Grayscale* 10
Gambar 2.2 Citra Hasil Ekstraksi *Transition Region*..... 10
Gambar 2.3 Histogram Tingkat Keabuan Citra Dan PDF 12
Gambar 2.4 *Pseudocode* Metode A* 14
Gambar 3.1 Alur Metode Penelitian 17
Gambar 3.2 Desain Sistem Yang Diusulkan..... 18
Gambar 3.3 Citra Dengan *Transition Region* Palsu..... 20
Gambar 3.4 *Flowchart* Perbaikan Citra 21
Gambar 3.5 Efek Penggunaan Median Filter Pada Citra Sederhana. 22
Gambar 3.6 Histogram Citra *Input* Dan Kepadatan Tingkat Keabuan 24
Gambar 3.7 *Flowchart* Pengelompokan Piksel-Piksel Citra..... 25
Gambar 3.8 Citra Dan *Transition Region* 26
Gambar 3.9 Ilustrasi Proses *Edge Linking* Pada *Transition Region* 27
Gambar 3.10 *Flowchart* Proses *Edge Linking* 30
Gambar 3.11 Ilustrasi Penyambungan Dengan A*. 31
Gambar 3.12 (A) Citra Hasil *Region Filling,* (B) Citra Hasil Segmentasi 32
Gambar 4.1 Contoh Citra Dari Dataset Weizmann, Msra, Dan Barkeley 35
Gambar 4.2 Citra Uji Untuk Pengujian Keperluan Menggunakan Filtering 36
Gambar 4.3 *Ground Truth* Dan *Transition Region* Untuk Citra *Simple Background & Simple Foreground: (A) Ground Truth,* (B) Menggunakan *Filtering,* (C) Tanpa *Filtering* 39
Gambar 4.4 *Ground Truth* Dan *Transition Region* Untuk Citra *Textured Background & Simple Foreground: (A) Ground Truth,* (B) Menggunakan *Filtering,* (C) Tanpa *Filtering* 39
Gambar 4.5 *Ground Truth* Dan *Transition Region* Untuk Citra *Simple Background & Textured Foreground: (A) Ground Truth,* (B) Menggunakan *Filtering,* (C) Tanpa *Filtering* 40
Gambar 4.6 *Ground Truth* Dan *Transition Region* Untuk Citra *Textured Foreground & Textured Background: (A) Ground Truth,* (B) Menggunakan *Filtering,* (C) Tanpa *Filtering* 40

Gambar 4.7 Citra Hasil Segmentasi Dengan Nilai Rata-Rata FPR Terkecil..... 46
Gambar 4.8 Citra Hasil Segmentasi Dengan Nilai Rata-Rata FNR Terkecil..... 46
Gambar 4.9 Hasil Segmentasi Dengan Kombinasi Parameter Paling Optimal..... 47
Gambar 4.10 Hasil Segmentasi Dengan Kombinasi Parameter Paling Optimal... 49
Gambar 4.11 *Transition Region* Setelah Proses Penghapusan..... 50
Gambar 4.12 Hasil Proses *Edge Linking* Dan *Region Filling*..... 51
Gambar 4.13 (A) Citra *Input* Dengan *Background* Bertekstur, Hasil (B) *Transition Region* Standar, (C) *Region Filling* Oleh Sistem, (D) Segmentasi. 54
Gambar 4.14 Hasil Segmentasi Oleh Penelitian Parida & Bhoi, 2016 dan Sistem Yang Diusulkan..... 55
Gambar 4.15 Hasil Segmentasi Oleh Penelitian Parida & Bhoi, 2016 dan Sistem Yang Diusulkan..... 56

DAFTAR TABEL

Tabel 3.1 Kelebihan, Kekurangan, Dan Fungsi Dua Citra *Transition Region* 28
Tabel 3.2 Jadwal Kegiatan Penelitian 34
Tabel 4.1 Hasil Pengujian Penggunaan *Filtering* 37
Tabel 4.2 Hasil Dari Jarak Kedekatan Maksimal 1 Piksel Dengan Ukuran *Window Filtering* 3 42
Tabel 4.3 Hasil Dari Jarak Kedekatan Maksimal 2 Piksel Dengan Ukuran *Window Filtering* 3 42
Tabel 4.4 Hasil Dari Jarak Kedekatan Maksimal 3 Piksel Dengan Ukuran *Window Filtering* 3 42
Tabel 4.5 Hasil Dari Jarak Kedekatan Maksimal 4 Piksel Dengan Ukuran *Window Filtering* 3 43
Tabel 4.6 Hasil Dari Jarak Kedekatan Maksimal 5 Piksel Dengan Ukuran *Window Filtering* 3 43
Tabel 4.7 Hasil Dari Jarak Kedekatan Maksimal 6 Piksel Dengan Ukuran *Window Filtering* 3 43
Tabel 4.8 Hasil Dari Jarak Kedekatan Maksimal 2 Piksel Dengan Ukuran *Window Filtering* 5 44
Tabel 4.9 Hasil Dari Jarak Kedekatan Maksimal 3 Piksel Dengan Ukuran *Window Filtering* 5 44
Tabel 4.10 Hasil Dari Jarak Kedekatan Maksimal 4 Piksel Dengan Ukuran *Window Filtering* 5 44
Tabel 4.11 Hasil Dari Jarak Kedekatan Maksimal 5 Piksel Dengan Ukuran *Window Filtering* 5 45
Tabel 4.12 Hasil Dari Jarak Kedekatan Maksimal 6 Piksel Dengan Ukuran *Window Filtering* 5 45
Tabel 4.13 Kombinasi Parameter Optimal Dan Rata-Rata ME 48
Tabel 4.14 Data Perbandingan Hasil Dari Sistem Yang Diusulkan Dengan Hasil Dari Penelitian Parida & Bhoi, 2016 53
Tabel 4.15 Nilai ME, FPR, FNR, Dan Lama Segmentasi Oleh Sistem Yang Diusulkan 57

BAB 1
PENDAHULUAN

Pada bab ini dijelaskan mengenai beberapa hal dasar dalam pembuatan proposal penelitian yang meliputi latar belakang, perumusan masalah, tujuan, manfaat, kontribusi penelitian, dan batasan masalah.

1.1 Latar Belakang

Segmentasi menjadi salah satu proses yang sangat penting dalam pengolahan citra digital dan visi komputer karena citra hasil segmentasi biasanya digunakan untuk proses lebih lanjut seperti analisa citra biomedis, identifikasi karakter dan pengenalan objek. Segmentasi citra merupakan salah satu tahap *preprocessing* yang sangat penting dan bertujuan untuk mendapatkan objek dari suatu citra dengan menggunakan karakteristik citra seperti tingkat keabuan, warna, dan tekstur (Parida & Bhoi, 2016)**.**

Secara umum, metode segmentasi citra bisa dikategorikan menjadi beberapa tipe pendekatan: pendekatan berbasis *thresholding* atau nilai ambang (Wang, Chung, & Xiong, 2008), pendekatan berbasis *boundary* (Felzenszwalb & Huttenlocher, 2004), pendekatan berbasis *region* (Salmbier & Marqués, 1999) dan pendekatan *hybrid* (Nyma, Kang, Kwon, Kim, & Kim, 2012; Yousefi, Azmi, & Zahedi, 2012). Pada segmentasi dengan *thresholding*, objek dan *background* dianggap memiliki distribusi tingkat keabuan yang berbeda. Hal ini berarti terdapat dua atau lebih *peak* (puncak) pada histogram citra yang bisa dipisahkan dengan *threshold*. Sehingga, proses segmentasi dalam pendekatan ini dilakukan dengan cara menentukan piksel-piksel yang memiliki tingkat keabuan di atas nilai *threshold* sebagai objek dan piksel-piksel yang memiliki tingkat keabuan di bawah nilai *threshold* sebagai *background*, begitu juga sebaliknya. Pendekatan berbasis *boundary* membangkitkan zona transisi, *edge* atau *boundary* pada citra yang memisahkan objek dan *background*. Segmentasi berbasis *region* dilakukan dengan cara mengelompokkan piksel dengan piksel tetangganya yang memiliki kesamaan nilai, memisahkan piksel yang memiliki nilai berbeda ke dalam grup

lain, atau kombinasi dari kedua cara tersebut (Kang, Wang, & Kang, 2012; Strasters & Gerbrands, 1991). Segmentasi *hybrid* bisa dilakukan dengan cara mengkombinasikan dua atau lebih pendekatan dengan tujuan untuk menghasilkan segmentasi yang lebih baik. Segmentasi dengan *thresholding* untuk menentukan *transition region* merupakan metode *hybrid* yang mengkombinasikan pendekatan berbasis *region* dan *thresholding* (Parida & Bhoi, 2016). Pada metode segmentasi ini, piksel-piksel yang memiliki nilai di atas *threshold* dianggap sebagai *transitional pixels* yang selanjutnya membentuk *transition region*. Proses *region filling* kemudian dilakukan pada *transition region* untuk menghasilkan citra biner yang di dalamnya terdapat area atau *region* dari objek. Setelah *region* diperoleh, proses segmentasi dilakukan dengan cara menampilkan piksel-piksel yang berada dalam *region* dari *transition region*. Seluruh piksel pada citra keabuan yang memiliki posisi beririsan dengan posisi objek pada citra biner, akan ditampilkan sebagai objek hasil segmentasi.

Zhang dan Gerbrands (1991) mengusulkan metode segmentasi citra berbasis *transition region* yang menggunakan *effective average gradient* (EAG) sebagai deskriptor *transition region*. Metode ini hanya bisa digunakan pada citra yang memiliki perubahan tingkat keabuan yang drastis sehingga tidak cocok bagi citra yang kompleks atau memiliki tingkat keabuan dengan variasi yang tinggi. Untuk mengatasi kelemahan tersebut, Yan et al. (2003) mengusulkan ekstraksi *transition region* berdasarkan *local entropy* (LE). Metode ini masih memiliki kekurangan apabila terdapat variasi tingkat keabuan yang banyak di *neighborhood*, sehingga akan meningkatkan nilai *local entropy* suatu piksel dan menjadikan piksel tersebut sebagai *transition region* walaupun dia sebenarnya adalah *foreground* atau *background*. Selain itu, metode ini juga membutuhkan kompleksitas komputasi yang tinggi karena melibatkan banyak perkalian dan operasi logaritmik. Untuk mengatasi kekurangan tersebut, Li & Liu (2009) mengusulkan sebuah metode baru *local gray level difference (LGLD) based transition region extraction method.* Metode ini selain mempertimbangkan perubahan tingkat keabuan juga mempertimbangkan banyaknya perubahan tersebut. Selanjutnya, sebuah metode *modified local entropy method (MLE)* diusulkan oleh Li et al., (2011). Metode ini memperhatikan frekuensi dan tingkat

perubahan *gray level* untuk ekstraksi *transition region.* Metode ini dapat menghasilkan transition region yang lebih bagus dibandingkan *LE*. Tetapi metode ini masih memiliki kekurangan dalam pemilihan nilai parameter untuk menyeimbangkan kontribusi *local complexity* dan *local variance*, dan nilai parameter untuk mengkontrol jumlah piksel di *transition region*. Kemudian Li et al. (2016) mengusulkan sebuah metode baru untuk segmentasi citra yang berisi objek tunggal dengan menggunakan *salient transition region*. Penelitian tersebut terbukti efektif dalam hal segmentasi citra dengan objek tunggal. Selanjutnya terdapat penelitian oleh Parida dan Bhoi (2016) yang menggunakan morfologi berdasarkan *transition region* pada segmentasi citra dengan objek tunggal maupun banyak. Metode ini menggunakan nilai *local variance* setiap piksel sebagai dasar untuk menentukan *transition region*. Setiap piksel yang memiliki nilai *local variance* lebih besar dari *threshold* akan dianggap sebagai *transitional pixel*. Kemudian setiap *transitional pixel* akan digabungkan dengan *transitional pixel* yang terhubung dengannya dan dijadikan sebagai *transition region.* Beberapa operasi *morfologi: thinning, cleaning, h-break removal, spurious pixel removal* kemudian dilakukan pada citra *transition region. Thinning* dilakukan untuk mendapatkan tepi dengan tebal satu piksel. Citra hasil *thinning* akan diproses dengan *cleaning* untuk menghapus *transition region* kecil dan terpisah. *H-break removal* kemudian dilakukan untuk menghapus *transitional pixel* yang berada di antara dua tepi *transition region*. *Spurious pixel removal* dilakukan untuk menghapus *transitional pixel* yang terhubung ke *transition region* pendek. Setelah melalui beberapa proses morfologi, proses *edge linking* untuk menyambungkan *transition region* yang tidak utuh dan saling berdekatan kemudian dilakukan. *Transition region* hasil dari proses *edge linking* mampu mensegmentasi objek tunggal maupun banyak. Namun metode ini masih memiliki dua kekurangan dalam hal *edge linking* dan *transition region* yang dibangkitkan dari *background*. *Edge linking* pada metode ini masih menggunakan batasan jarak antar *transition region* sehingga ada kemungkinan *transition region* akan disambungkan dengan *transition region* lain yang dekat tetapi tidak memiliki kesamaan visual. Penyambungan *transition region* yang tidak tepat akan menimbulkan hasil segmentasi yang kurang bagus karena akan terdapat *background* yang tergabung

dengan objek. Metode ini juga masih membangkitkan *transition region* di area *background* jika tingkat keabuan dari citra terlalu kompleks.

Menurut penelitian Parida & Bhoi (2016) sebuah metode ekstraksi *transition region* yang dapat menghasilkan *transition region* yang lebih utuh dan tidak muncul di area *background* sangatlah dibutuhkan. *Transition region* yang dihasilkan sangatlah tergantung pada karakteristik *background* yang ada pada citra. Pada citra dengan *background* yang mengandung variasi tingkat keabuan yang tinggi, terdapat perubahan tingkat keabuan yang drastis, atau *background* bertekstur, maka mungkin mengakibatkan munculnya *transition region* palsu di area *background.* Untuk mengatasi masalah tersebut, penelitian ini menambahkan sebuah proses perbaikan citra yang menggunakan teknik penghalusan citra dengan *median filter* untuk memperbaiki citra yang banyak menghasilkan *transition region* palsu.

Berdasarkan beberapa penelitian (Li et al., 2011; Li & Liu, 2009; Yan et al., 2003) menunjukkan bahwa transition region bisa diekstraksi dengan bagus atau utuh pada citra dengan variasi tingkat keabuan tidak terlalu banyak (sederhana). Penyederhaan tingkat keabuan pada citra bisa dilakukan dengan mengelompokkan piksel-piksel berdasarkan tingkat keabuannya. Pengelompokan piksel bisa dilakukan dengan metode k-means *clustering* (Dhanachandra, Manglem, & Chanu, 2015) dengan nilai k adaptif yang bisa didapat dari histogram citra (Shakeri, Dezfoulian, Khotanlou, Barati, & Masoumi, 2017). Citra yang disederhanakan bisa meminimalisir kemunculan *transition region* pada area *background* karena sebagian tingkat keabuan *background* telah diseragamkan. Selain itu, citra tersebut juga dapat menghasilkan *transition region* yang lebih utuh dibandingkan pada citra asli. Hal itu bisa terjadi karena tingkat keabuan dari setiap piksel pada citra sederhana langsung diganti menjadi keabuan dari *centroid cluster*nya, sehingga akan menyebabkan adanya perubahan tingkat keabuan yang drastis pada setiap tepi. Berdasarkan hal-hal tersebut, *transition region* hasil dari citra yang telah disederhanakan digunakan sebagai acuan untuk memperbaiki *transition region* pada citra asli. Perbaikan tersebut berupa penghapusan dan penyambungan *transition region* pada citra asli.

Penghapusan *transition region* dilakukan terhadap *transition region* pada citra asli yang memiliki persentase irisan yang sedikit dengan *transition region* pada citra yang disederhanakan. Proses penghapusan ini akan mengurangi jumlah *transition region* yang muncul di area background. *Transition region* yang sudah melalui proses penghapusan masih membutuhkan proses penyambungan tepi apabila terdapat *transition region* yang kurang utuh. Penyambungan tepi yang tidak utuh bisa dilakukan dengan cara *edge linking* menggunakan metode pencarian rute seperti penelitian Ji et al (2013) untuk segmentasi citra *remote sensing*. Dalam penelitian tersebut, *edge strength* setiap piksel dihitung terlebih dahulu berdasarkan *Sobel gradient strength*, *gradient direction*, dan *forward direction.* Kumpulan *initial edge points* selanjutnya ditentukan dari setiap piksel dengan nilai *edge strength* yang besar. Pencarian rute selanjutnya dilakukan menggunakan metode A* untuk menyambungkan setiap *initial edge points*. Proses edge linking tersebut selesai apabila semua *initial edge points* sudah terlibat. Metode ini dapat melakukan segmentasi dengan bagus karena setiap tepi objek pada citra dapat terhubung tanpa adanya batas jarak antar tepi. Berdasarkan uraian tersebut, penelitian ini akan menggunakan metode A* untuk melakukan proses *edge linking* pada *transition region* yang kurang utuh. Proses *edge linking* dengan metode A* ini diharapkan bisa menyambungkan *transition region* yang kurang utuh berdasarkan kesamaan visual yang diperoleh dari *transition region* citra sederhana.

Penelitian ini mengusulkan sebuah metode baru untuk segmentasi citra dengan mengkombinasikan metode *transition region* dan metode *adaptive* k-means *clustering*. Segmentasi citra dengan skema tersebut diharapkan mampu menghasilkan *transition region* yang lebih utuh sehingga memberikan hasil segmentasi yang lebih bagus.

1.2 Perumusan Masalah

Rumusan masalah yang diangkat dalam penelitian ini adalah sebagai berikut:

1. Bagaimana mendapatkan *transition region* pada citra dengan karakteristik yang lebih baik, yaitu yang mengelilingi objek dan bersih

dengan menggunakan metode *transition region* berbasis *local variance*, *median filter* dan metode *adaptive* k-means *clustering*?

2. Bagaimana menghubungkan *transition region* yang belum utuh menggunakan operasi morfologi *thinning* dan *edge linking* berbasis A*?
3. Bagaimana melakukan pengisian lubang pada *transition region* untuk mendapatkan objek pada citra menggunakan operasi morfologi *region filling*?
4. Bagaimana mengevaluasi kinerja metode segmentasi ini?

1.3 Tujuan

Tujuan penelitian ini adalah melakukan segmentasi citra multi-objek menggunakan metode *transition region*, k-means dan A* yang dapat memisahkan objek dari *background* secara utuh.

1.4 Manfaat

Hasil dari penerapan metode segmentasi yang diusulkan pada tesis ini diharapkan dapat memberikan hasil segmentasi citra yang lebih baik, sehingga dapat memisahkan objek secara utuh dari *background*. Hasil segmentasi yang baik bisa meningkatkan kinerja aplikasi berbasis *computer vision* seperti alat bantu untuk pembelajaran struktur anatomi, navigasi saat operasi, menentukan lokasi tumor, mendeteksi wajah, deteksi pejalan kaki, deteksi objek pada citra satelit, deteksi cahaya lampu rem, dan sebagainya.

1.5 Kontribusi Penelitian

Kontribusi penelitian ini adalah mengusulkan sebuah metode baru untuk segmentasi citra berbasis *transition region* dengan memperbaiki hasil ekstraksi *transition region* menggunakan metode *adaptive* k-means *clustering* dan *median filter* untuk menghasilkan *transition region* yang lebih bersih serta metode A* untuk menghasilkan *transition region* yang lebih utuh.

1.6 Batasan Masalah

Batasan masalah pada penelitian ini adalah:

1. Dataset yang digunakan adalah dataset Weizmann, MSRA dan Barkeley yang berisi citra satu dan banyak objek, dan beberapa citra *input* yang digunakan oleh penelitian Parida & Bhoi (2016).
2. Sistem ini dibuat menggunakan aplikasi Matlab 2017a.

BAB 2
KAJIAN PUSTAKA

Pada bab ini dijelaskan tentang kajian pustaka dan dasar teori yang terkait dengan landasan penelitian ini. Pustaka yang terkait adalah tentang *transition region*, *k-means*, *probability density estimate function*, *A**, *thinning*, *region filling*, *median filter*, *misclassification error* (ME), *false positive rate* (FPR), dan *false negative rate* (FNR).

2.1. *Transition Region*

Transition region adalah sebuah struktur dalam citra yang mirip dengan tepi. *Transition region* memiliki tiga karakteristik. Pertama, *transition region* biasanya memiliki lebar beberapa piksel dekat tepi. Kedua, *transition region* mengelilingi objek dan seharusnya terletak antara objek dan *background*. Ketiga, terdapat perubahan tingkat keabuan pada piksel-piksel di *transition region*. Informasi untuk ekstraksi *transition region* bisa diperoleh dari *transitional pixels* yang memiliki ukuran dan frekuensi tingkat keabuan yang lebih besar daripada *non-transitional pixels*. Proses ekstraksi *transition region* bertujuan untuk mendapatkan zona transisi yang dekat dengan garis luar objek sehingga dapat digunakan untuk memisahkan objek dan *background* (Li et al., 2016).

Terdapat banyak deskriptor yang sudah dikembangkan untuk ekstraksi *transition region* (Li et al., 2011; Li & Liu, 2009; Yan et al., 2003). Pada penelitian ini *local variance* digunakan untuk ekstraksi *transition region*. *Local variance* bisa membedakan area yang mengandung tepi atau tidak (Parida & Bhoi, 2016). Tepi biasanya terdapat pada area yang memiliki nilai *variance* yang tinggi.

Untuk piksel tengah *(i,j)* dari *local neighborhood window* yang berukuran *m x m*, *local variance* bisa dihitung dengan menggunakan Persamaan 2.1.

$$LV(i,j) = \frac{1}{m^2 - 1} \sum_{x=1}^{m} \sum_{y=1}^{m} (ab(x,y) - abm)^2 \tag{2.1}$$

Dengan *ab(x,y)* menyatakan nilai tingkat keabuan piksel lokal dalam *local neighborhood window* dan abm menyatakan rata-rata tingkat keabuan *local neighborhood window* tersebut.

Dengan menggerakkan *window* dari kiri ke kanan dan atas ke bawah, perhitungan *local variance* dijalankan di seluruh citra untuk mendapatkan matriks nilai *local variance* seperti yang ditunjukkan pada Persamaan 2.2.

$$LV = \begin{bmatrix} LV(1,1) & LV(1,2) & \dots & LV(1,N) \\ LV(2,1) & LV(2,2) & \dots & LV(2,N) \\ \dots & \dots & \dots & \dots \\ LV(M,1) & LV(M,1) & \dots & LV(M,N) \end{bmatrix} \tag{2.2}$$

Dengan *M* menyatakan tinggi dari citra dan *N* menyatakan lebar dari citra.

Setelah matriks *local variance* diperoleh, nilai-nilai dalam matriks tersebut akan dibandingkan dengan nilai *global threshold* T_g. Nilai T_g bisa dihitung menggunakan persamaan 2.3.

$$T_g = \frac{1}{M \times N} \sum_{k=1}^{M} \sum_{l=1}^{N} AB(k,l) \tag{2.3}$$

dimana *AB(k,l)* merupakan tingkat keabuan piksel *(k,l)* pada citra *input*. Piksel yang memiliki nilai *local variance* lebih besar atau sama dengan T_g akan dianggap sebagai *transitional pixel* yang ditunjukkan pada persamaan 2.4.

$$\Omega_R(i,j) = \begin{cases} 0 & \text{jika } LV(i,j) \geq T_g \\ 255 & \text{Sebaliknya} \end{cases} \tag{2.4}$$

Gambar 2.1 Citra *Grayscale*

Gambar 2.2 Citra Hasil Ekstraksi *Transition Region*

Transition region disusun dengan mengelompokkan *transitional pixels* Ω_R yang terhubung dan menjadikan setiap kelompok menjadi *transition region* yang berbeda seperti pada Gambar 2.2.

2.2. K-means *Clustering*

K-means *clustering* merupakan salah satu metode clustering yang popular. K-means *clustering* mengelompokkan sekumpulan data ke sejumlah *k* grup data. Algoritma K-means *clustering* terdiri dari dua fase yang berbeda. Fase yang pertama algoritma ini menghitung *centroid* setiap *cluster*, fase yang kedua algoritma ini menetapkan setiap poin ke dalam grup dengan *centroid* yang memiliki jarak terdekat ke poin tersebut. Terdapat beberapa cara atau metode yang berbeda untuk menghitung jarak *centroid* terdekat. Jarak Euclidean menjadi salah satu metode yang paling sering digunakan untuk menghitung jarak tersebut (Dhanachandra et al., 2015).

Perhitungan untuk *centroid* baru setiap *cluster* dijalankan kembali setelah proses pengelompokan selesai. Berdasarkan *centroid* baru tersebut jarak Euclidean baru antara poin dan *centroid* dihitung kembali dan kemudian menetapkan poin tersebut ke *cluster* dengan jarak Euclidean terdekat dengannya. Setiap *cluster* didefinisikan dengan poin-poin anggota dan *centroid*. *Centroid* untuk setiap *cluster* adalah poin yang memiliki jarak minimum dengan poin-poin lainnya di *cluster* tersebut. Sehingga k-means *clustering* merupakan algoritma iteratif yang meminimalkan jumlah jarak dari setiap poin ke *centroid cluster* (Dhanachandra et al., 2015). Berikut langkah-langkah algoritma k-means *clustering* (Jaroš et al., 2016):

a. Inisialisasi jumlah *cluster k* dan *centroid cluster*.
b. Hitung jarak Euclidean *d* antara *centroid* c_k dan setiap piksel $ab(x, y)$ menggunakan persamaan berikut ini:

$$d = \sqrt{(ab(x, y) - c_k)^2} \tag{2.5}$$

c. Kelompokkan setiap piksel ke *cluster* yang terdekat berdasarkan jarak Euclidean *d*.

d. Setelah semua piksel dikelompokkan, hitung kembali posisi baru untuk *centroid* menggunakan persamaan berikut ini

$$c_k = \frac{1}{Nc_k}\sum_{i=1}^{Nc_k} ab_i \tag{2.6}$$

dimana Nc_k merupakan jumlah anggota *cluster k.*

e. Ulangi proses a–d sampai tidak ada perubahan pada *centroid* baru.

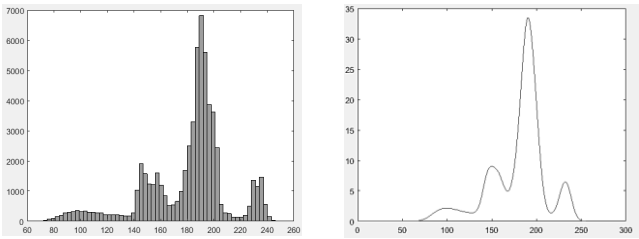
Dalam penelitian ini K-means *clustering* digunakan untuk mengelompokan piksel-piksel citra sesuai tingkat keabuannya. Pengelompokan tersebut berfungsi untuk menyederhanakan tingkat keabuan citra dengan cara mengganti tingkat keabuan piksel dengan *centroid* clusternya. Citra yang telah disederhanakan diharapkan dapat digunakan untuk menghasilkan *transition region* yang utuh sesuai bentuk objeknya.

2.3. *Probability Density Estimate Function*

Probability density estimate function diperlukan untuk menghitung kepadatan tingkat keabuan pada citra dan dapat menghasilkan perkiraan jumlah *cluster*. Nilai *probability density estimate function* setiap tingkat keabuan pf yang ada di histogram dihitung menggunakan Persamaan 2.7 (Shakeri et al., 2017). Histogram kepadatan tingkat keabuan kemudian dibangkitkan dari nilai-nilai tersebut. Histogram tersebut akan lebih halus daripada histogram tingkat keabuan citra seperti yang terlihat pada Gambar 2.3.

$$pf(i) = \frac{1}{n}\sum_{x=1}^{M}\sum_{y=1}^{N} K\left(\frac{i - ab(x,y)}{h}\right), \qquad i = 0, \ldots, 255 \tag{2.7}$$

dengan *n* menyatakan panjang array pf, *h* merupakan *bandwidth*, dan fungsi *K(.)* menyatakan kernel Gauss (Shakeri et al., 2017).



(a) (b)

Gambar 2.3 Histogram (a) Tingkat Keabuan dan (b) Hasil Proses *pdf*

2.4. Metode A*

Metode A* merupakan salah satu metode pencarian *Best-first search* yang sering digunakan. Metode ini memperhatikan node yang memiliki jarak terdekat menuju node tujuan *f(n)*. Nilai *f(n)* diperoleh dari penjumlahan jarak dari *node* awal ke *node* saat ini *g(n)* dan perkiraan jarak dari *node* saat ini ke *node* tujuan (target) *h(n)* yang ditunjukkan pada Persamaan 2.8. Metode A* secara detail ditunjukkan pada Gambar 2.4 (Suyanto, 2011).

$$f(n) = g(n) + h(n) \tag{2.8}$$

2.5. *Thinning* (penipisan)

Thinning merupakan salah satu operasi morfologi yang dapat menghapus piksel-piksel *foreground* pada citra biner. *Thinning* dapat digunakan untuk mengubah ketebalan *foreground* menjadi satu piksel. Proses ini biasa dilakukan pada citra biner dan akan menghasilkan citra biner lainnya. Penipisan citra biner A oleh strel B dapat didefinisikan dalam transformasi hit-or-miss dengan Persamaan 2.9 (Prasetyo, 2011).

$$A \otimes B = A - (A \circledast B) \tag{2.9}$$

2.6. *Region Filling*

Region filling merupakan sebuah metode untuk mengisi area yang dikelilingi oleh tepi dengan sebuah warna atau citra (Duo-le & Ming, 2011). Proses ini memberikan warna pada piksel yang terhubung dengan 4 atau 8 piksel lainnya, dan kemudian proses ini mewarnai semua piksel dalam suatu *region*.

2.7. *Median Filter*

Filter smoothing dapat digunakan untuk membuat citra menjadi lebih halus sehingga noise pada citra dapat berkurang. Salah satu metode yang dapat digunakan untuk penghalusan adalah *median filter*. *Median filter* merubah nilai piksel saat ini dengan *median* dari intensitas piksel-piksel tetangganya (Prasetyo, 2011).

```
Open ← simpul awal
Closed ← array kosong
   Loop sampai goal ditemukan atau sampai tidak ada simpul di dalam Open
     If open = kosong then
        Gagal
     Else
        Bestnode = simpul yang ada di open dengan f minimal
        Pindahkan simpul terbaik tersebut dari open ke closed
        If bestnode = goal then
            Sukses
        Else
            Bangkitkan semua suksesor bestnode
            Untuk setiap suksesor kerjakan:
            Hitung g(suksesor) = g(bestnode) +
            actual cost(dari bestnode ke suksesor)
            If suksesor ada di open then
               Old = simpul di open yang sama dengan suksesor tersebut
               Bandingkan nilai g(old) dengan g(suksesor)
               If g(suksesor) lebih baik then
                  Ubah parent old ke bestnode
                  Ubah nilai g dan f yang ada pada old
               End
            Else
               If suksesor ada di closed then
                  Old = simpul di closed yang sama dengan suksesor tersebut
                  Bandingkan nilai g(old) dengan g(suksesor)
                  If g(suksesor) lebih baik then
                     Ubah parent old ke bestnode
                     Ubah nilai g dan f yang ada pada old
                     Loop sampai simpul suksesor tidak ada di open
                       atau simpul tidak punya suksesor
                        Ubah nilai g dan f untuk suksesor old
                     End
                  End
               Else
                  Masukkan suksesor ke open
                  Tambahkan suksesor tersebut sebagai suksesornya bestnode
                  Hitung f=g(suksesor)+h(suksesor)
               End
            End
        End
     End
End
```

Gambar 2.4 *Pseudocode* Metode A*

2.8. *Misclassification Error* (ME)

Misclassification error (ME) merupakan sebuah metode yang digunakan untuk menghitung banyaknya kesalahan klasifikasi pada piksel. ME dihitung dengan menggunakan persamaan 2.10 (Parida & Bhoi, 2016).

$$ME = 1 - \frac{|B_o \cap B_t| + |Fr_o \cap Fr_t|}{|B_o| + |Fr_o|} \tag{2.10}$$

dimana B_o menyatakan kumpulan piksel *background* pada citra *ground truth*, Fr_o menyatakan kumpulan piksel *foreground* pada citra *ground truth*, B_t menyatakan kumpulan piksel *background* pada citra hasil segmentasi oleh sistem, dan Fr_t menyatakan kumpulan piksel *foreground* pada citra hasil segmentasi oleh sistem. Nilai ME kemudian ditunjukkan dalam bentuk persen.

2.9. *False Positive Rate* (FPR)

False Positive Rate menunjukkan kapasitas jumlah piksel *background* yang terklasifikasi sebagai piksel *foreground.* FPR dapat dihitung menggunakan persamaan 2.11 (Parida & Bhoi, 2016). Nilai FPR tersebut kemudian ditunjukkan dalam bentuk persen.

$$FPR = \frac{|B_o \cap Fr_t|}{|B_o|} \tag{2.11}$$

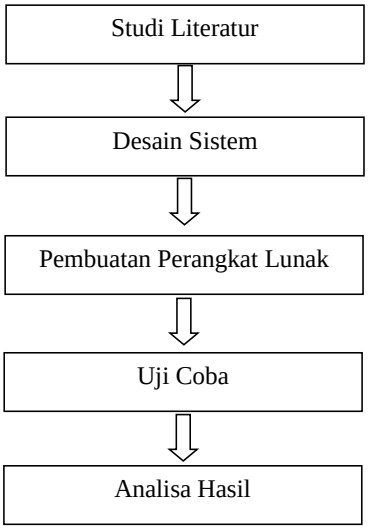
2.10. *False Negative Rate* (FNR)

False Negative Rate menunjukkan kapasitas jumlah piksel *foreground* yang terklasifikasi sebagai piksel *background.* FNR dapat dihitung menggunakan persamaan 2.12 (Parida & Bhoi, 2016). Nilai FNR tersebut kemudian ditunjukkan dalam bentuk persen.

$$FNR = \frac{|Fr_o \cap B_t|}{|Fr_o|} \tag{2.12}$$

BAB 3
METODOLOGI PENELITIAN

Terdapat beberapa tahap dalam metodologi penelitian ini, meliputi studi literatur, desain sistem, pembuatan perangkat lunak, uji coba, dan analisa hasil. Alur metode penelitian tersebut dapat dilihat pada Gambar 3.1.



Gambar 3.1 Alur Metode Penelitian

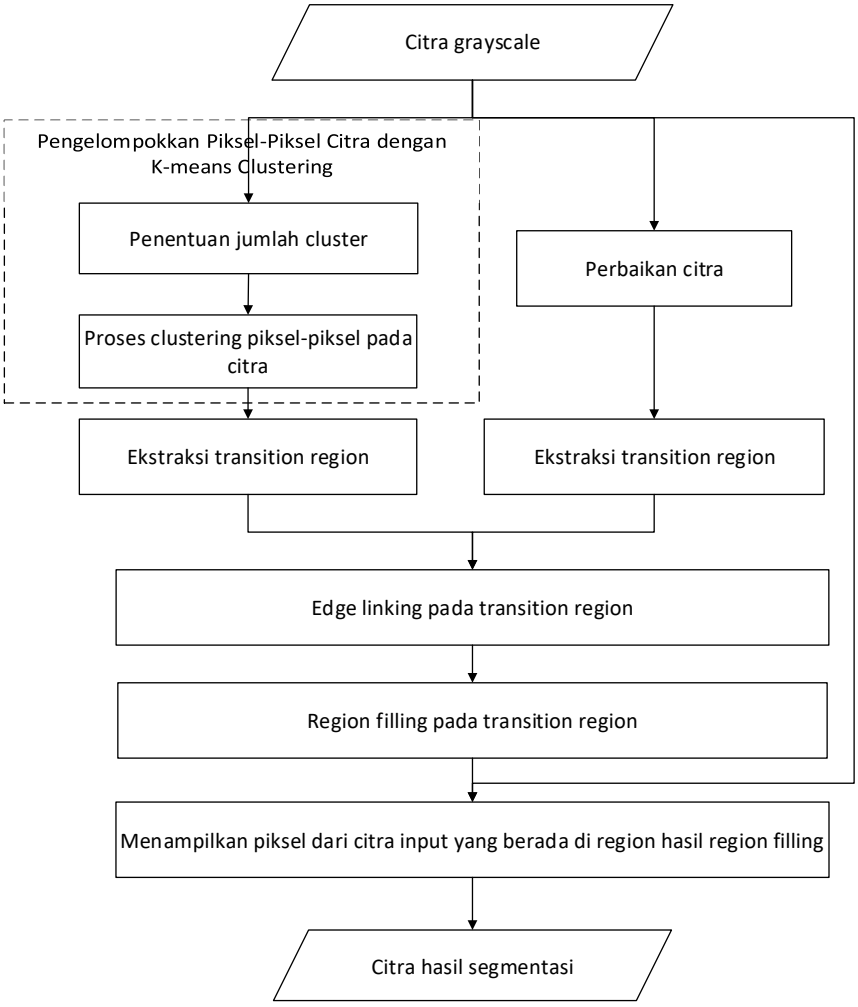
3.1. Studi Literatur

Studi literatur dilakukan untuk mendapatkan informasi dan menganalisa perkembangan metodologi pada penelitian ini. Beberapa literatur yang dibutuhkan oleh sistem ini adalah:

1. Ekstraksi *transition region*
2. Pengelompokan piksel menggunakan metode *k-means clustering*
3. Penghalusan histogram menggunakan *probability density estimate function*
4. Penyambungan ujung dengan metode *A**
5. Morfologi *thinning*
6. *Region filling* untuk mengisi area *transition region*
7. Metode *misclassification error* (ME), *False Positive Rate* (FPR), dan *False Negative Rate* (FNR) untuk menganalisa hasil.

3.2. Desain Sistem

Tahap segmentasi citra dengan metode yang diusulkan ditunjukan pada Gambar 3.2. Citra *input* yang akan disegmentasi menggunakan citra *grayscale*. Citra tersebut akan digunakan oleh dua macam proses: ekstraksi *transition region* citra asli dan ekstraksi *transition region* citra sederhana secara terpisah. Selanjutnya akan dilakukan proses *edge linking* pada *transition region* citra asli berdasarkan *transition region* citra sederhana. *Region filling* kemudian dilakukan pada *transition region* hasil proses *edge linking* untuk menghasilkan area yang dianggap sebagai objek. Kemudian dilakukan proses segmentasi dengan cara menampilkan piksel-piksel yang berada pada area *transition region*. Dan hasil dari sistem ini adalah sebuah citra segmentasi yang berisi objek tunggal atau banyak.



Gambar 3.2 Desain Sistem yang Diusulkan

3.2.1. Ekstraksi *Transition Region*

Fase ini bertujuan untuk menghasilkan *transition region* dari citra asli dan citra yang tingkat kebuannya telah disederhanakan menggunakan nilai *local variance* tiap piksel dengan *local neighborhood* berukuran 3 x 3 (Parida & Bhoi, 2016). Penjelasan lebih detail dari proses ekstraksi *transition region* bisa dilihat pada Sub bab 2.1. Kualitas *transition region* yang dihasilkan oleh fase ini dipengaruhi oleh kompleksitas tingkat keabuan citra. Pada citra dengan kompleksitas tingkat keabuan sederhana *transition region* dapat diekstraksi dengan utuh. Tetapi, *transition region* mungkin akan diekstraksi kurang utuh atau banyak bangkit di *background* jika tingkat keabuan citra cukup kompleks seperti pada Gambar 3.8 (b). Supaya *transition region* dari citra yang tingkat keabuannya telah disederhanakan selalu dibangkitkan dengan utuh maka *global threshold* yang digunakan untuk ekstraksi *transition region* dari citra sederhana adalah 0.

Menurut penelitian Parida, *transition region* yang dianggap tidak penting atau bagian dari *background* dapat dikurangi dengan cara menghapus *transition region* yang berukuran kecil. Proses penghapusan *transition region* yang kecil mengadopsi cara dari penelitian Rosyadi et al. (2018). Terdapat beberapa langkah untuk menghapus *transition region* yang kecil, diantaranya:

1. Membagi citra menjadi *b x b* blok, berdasarkan penelitian Rosyadi et al. (2018) *b* yang digunakan adalah 4.
2. Menghitung tinggi blok HB, lebar blok WB, dan luas blok LB dengan Persamaan 3.14, 3.2, dan 3.3.

$$HB = \frac{M}{b} \tag{3.1}$$

$$WB = \frac{N}{b} \tag{3.2}$$

$$LB = HB * WB \tag{3.3}$$

3. Menghitung tinggi *transition region* HTR_l dengan cara mengurangi koordinat paling bawah bb_l dengan koordinat paling atas bt_l *transition region*.

4. Menghitung lebar *transition region* WTR_l dengan cara mengurangi koordinat paling kanan br_l dengan koordinat paling kiri bl_l *transition region*.
5. Menghitung luas area *transition region* LTR_l menggunakan Persamaan 3.4.

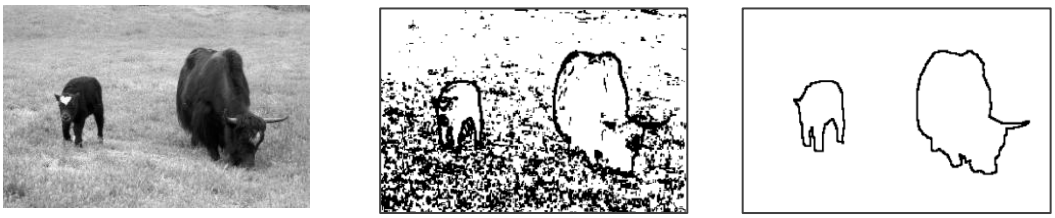
$$LTR_l = HTR_l * WTR_l \tag{3.4}$$

6. Menghapus *transition region* jika luasnya lebih kecil dari 3% luas blok *LB*.

Proses penghapusan ini akan dilakukan pada *transition region* dari citra asli dan *transition region* dari citra yang tingkat keabuannya telah disederhanakan. *Transition region* dari citra asli yang tersisa akan diproses oleh fase-fase selanjutnya berdasarkan *transition region* dari citra yang telah disederhanakan. *Transition region* dari citra yang telah disederhanakan dapat digunakan sebagai penyambung transition region citra asli yang belum utuh karena bentuknya yang lebih utuh dibandingkan *transition region* dari citra asli seperti pada Gambar 3.8 (d). Selain itu, *transition region* dari citra sederhana juga dapat digunakan untuk menghapus *transition region* yang mirip dengan tingkat keabuan *background.*

3.2.2. Perbaikan Citra

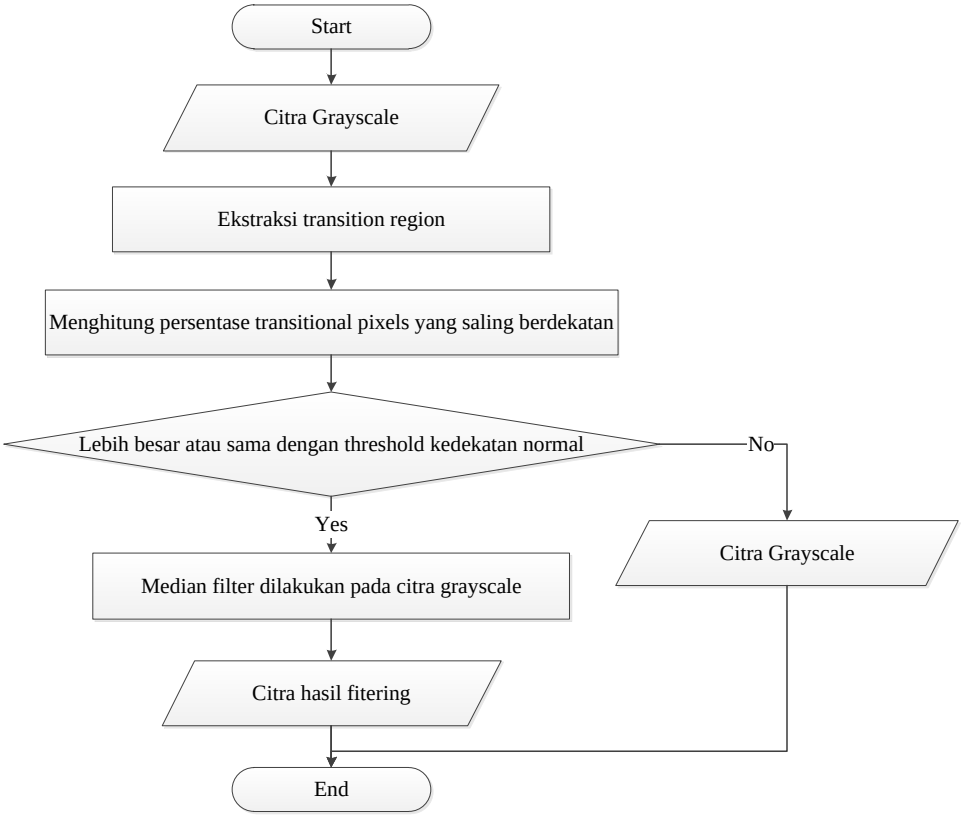
Transition region yang dihasilkan sangatlah tergantung pada *background* yang ada pada citra. Pada citra dengan *background* yang sederhana, *transition region* akan dihasilkan dengan bagus. Tetapi, pada citra dengan *background* yang mengandung variasi tingkat keabuan yang tinggi, terdapat perubahan tingkat keabuan yang drastis, atau *background* bertekstur, maka mungkin mengakibatkan munculnya *transition region* atau *transitional pixels* palsu di area *background.*



(a) (b) (c)

Gambar 3.3 Citra dengan *Transition Region* Palsu, (a) Citra *Input*, (b) *Transition Region*, (c) *Transition Region* yang Diharapkan

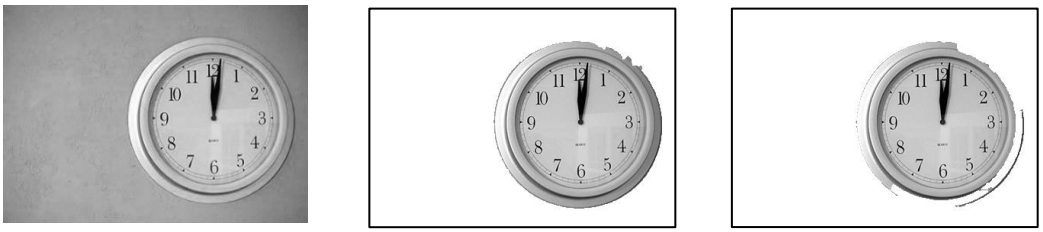
Munculnya *transition region* palsu di area *background* diharapkan bisa diatasi menggunakan teknik penghalusan atau *filtering*, salah satunya dengan menggunakan *median filter*. Namun, penggunaan teknik *filtering* juga bisa memperburuk hasil ekstraksi *transition region* pada citra dengan *background* yang bagus (variasi tingkat keabuan sedikit, tidak ada perubahan tingkat keabuan yang drastis dan *background* tidak bertekstur). Sehingga, penelitian ini juga mengusulkan sebuah teknik baru yang dapat menentukan secara otomatis apakah teknik *filtering* perlu digunakan untuk memperbaiki citra *input* atau tidak.



Gambar 3.4 *Flowchart* Perbaikan Citra

Transition region diharapkan utuh dan muncul di tepi objek sehingga dapat diisi oleh *region filling* dan sistem dapat menghasilkan hasil segmentasi yang baik seperti Gambar 3.3 (c). *Transitional pixels* yang melekat satu sama lain dianggap normal karena dianggap sebagai satu *transition region*. Namun, *transitional pixels* yang berdekatan dengan *transitional pixels* lain tetapi tidak melekat dapat dianggap sebagai *transitional pixels* palsu karena sering muncul di

tengah *foreground* atau *background* seperti yang ditunjukkan pada Gambar 3.3 (b). Kemunculan *transitional pixels* palsu yang sedikit tidak terlalu berpengaruh pada citra hasil segmentasi. Tetapi apabila kemunculan *transitional pixels* palsu cukup banyak, maka kualitas citra hasil segmentasi akan menjadi lebih buruk. Hal itu terjadi karena sistem akan menampilkan banyak piksel *background* sebagai objek. Sehingga munculnya *transitional pixels* palsu yang cukup banyak diharapkan dapat dikurangi oleh proses perbaikan citra menggunakan *median filter*.



(a) (b) (c)

Gambar 3.5 Efek Penggunaan Median Filter pada Citra Sederhana (a) Citra *Input*, Hasil Segmentasi (b) tanpa *Median Filter* pada Citra *Input* untuk Proses Pengelompokan Piksel-Piksel Citra, (c) dengan *Median Filter* pada Citra *Input* untuk Proses Pengelompokan Piksel-Piksel Citra.

Transitional pixels palsu biasanya memiliki jarak yang cukup dekat dengan *transitional pixel* lainnya. Jumlah *transitional pixels* yang saling berdekatan *dtp* dihitung secara horizontal. Jumlah tersebut akan dibandingkan dengan jumlah *transitional pixel* yang saling melekat *dtm* untuk memperoleh nilai persentase *transitional pixels* dekat *ptp* menggunakan Persamaan 3.5. Apabila nilai *ptp* lebih besar atau sama dengan *threshold* persentase normal *TPn* maka proses *median filter* pada citra *input* dilakukan. *Flowchart* perbaikan citra dapat dilihat pada Gambar 3.4

$$ptp = \left(\frac{dtp}{dtm}\right) * 100 \tag{3.5}$$

Proses perbaikan citra ini tidak dilakukan terhadap citra *input* untuk proses pengelompokan piksel-piksel citra. Hal ini bertujuan untuk tetap menjaga bentuk dari objek pada *transition region* citra sederhana tetap utuh karena akan digunakan sebagai sambungan *transition region* citra asli yang putus. Apabila

perbaikan citra ini tetap dilakukan pada citra *input* untuk proses pengelompokan piksel-piksel citra, dikhawatirkan bentuk objek dari citra bisa berubah atau ada *transition region* pada objek yang putus karena efek penghalusan oleh *median filter* yang mengakibatkan hasil segmentasi menjadi buruk seperti Gambar 3.5.

3.2.3. Pengelompokan Piksel-Piksel Citra dengan K-means *Clustering*

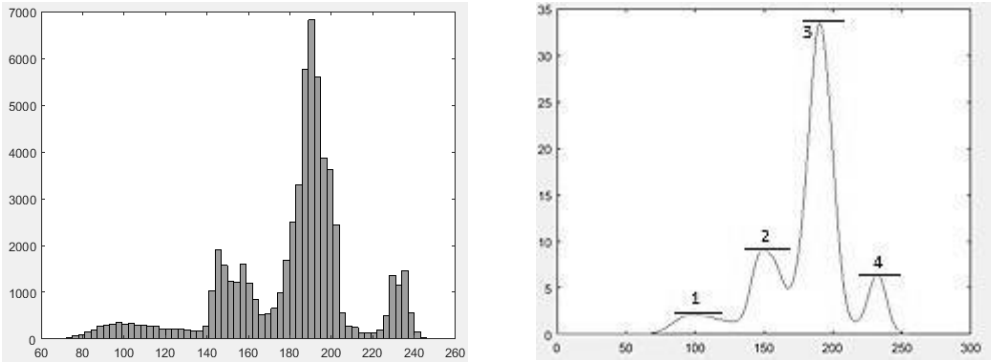
Fase ini bertujuan untuk menghasilkan citra dengan tingkat keabuan yang lebih sederhana seperti pada Gambar 3.8 (c). Citra tersebut akan digunakan untuk proses ekstraksi *transition region* citra sederhana. *Transition region* dari citra ini diharapkan lebih utuh dan bisa digunakan untuk menyeleksi dan menyambungkan *transition region* dari citra asli yang kurang utuh. Fase ini meliputi proses penentuan jumlah *cluster* dan proses *clustering* piksel-piksel pada citra. Fase ini dapat dilihat secara detail pada Gambar 3.7.

Penelitian ini menggunakan teknik *clustering* dengan jumlah *cluster* yang adaptif sesuai dengan citra *input*. Histogram tingkat keabuan yang dimiliki oleh citra *input* terlebih dahulu dibangkitkan seperti yang terlihat pada Gambar 3.6 (a). Histogram tersebut kemudian dihaluskan dengan menggunakan perhitungan *probability density estimate function* yang bisa dilihat pada Sub bab 2.3. Dalam histogram hasil operasi penghalusan, terdapat poin-poin yang bisa digunakan sebagai *peak* atau *cluster* seperti pada Gambar 3.6 (b). Untuk menentukan poin-poin yang digunakan sebagai *peak*, setiap poin yang terdapat di histogram dibandingkan dengan dua poin tetangganya. Poin yang memiliki nilai pf lebih besar dari nilai pf satu poin sebelum dan satu poin sesudahnya akan memiliki nilai δ_j 1 seperti yang terlihat pada Persamaan 3.6. Setiap poin dengan nilai δ_j 1 akan ditetapkan sebagai *peak*.

$$k = \sum_{j=0}^{n-1} \delta_j, \qquad \delta_j = \begin{cases} 1, pf(j-1) < pf(j)\ dan\ pf(j+1) < pf(j) \\ 0, sebaliknya \end{cases}, \tag{3.6}$$

dimana *n* menyatakan panjang array pf dan *k* menyatakan jumlah *peak* di histogram.

Apabila terdapat *peak* yang memiliki keabuan mirip dengan *peak* tetangganya (berjarak lebih kecil dari *threshold* kemiripan) maka *peak* tetangganya akan dihapus. *Threshold* kemiripan Tk antara dua *peak* diperoleh dari perhitungan *mean* μ dan standar deviasi *std* menggunakan Persamaan 3.7, 3.8 dan 3.9 (Shakeri et al., 2017). *Bin* dari *peak* yang ada kemudian dijadikan sebagai *centroid* awal setiap *cluster* pada proses *clustering* piksel menggunakan metode k-means *clustering*.



(a) (b)

Gambar 3.6 Histogram (a) Citra *Input* dan (b) Nilai Kepadatan Tingkat Keabuan

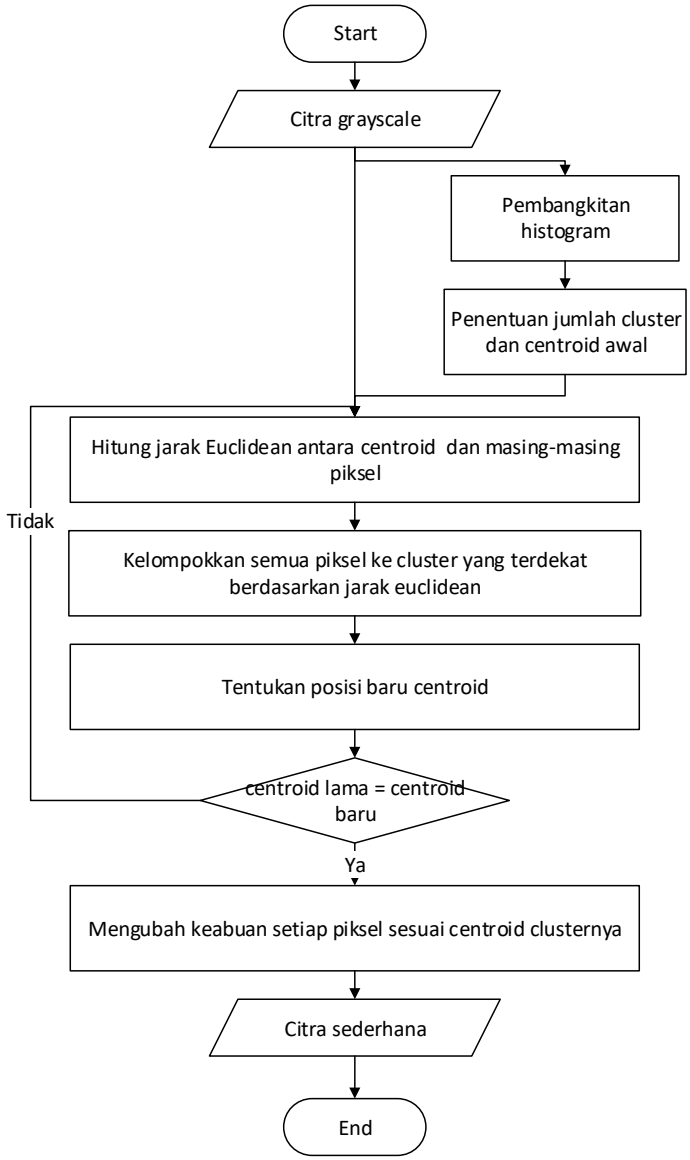
$$\mu = \frac{\sum_{i=0}^{255}(pf(i) * i)}{\sum_{i=0}^{255} pf(i)} \tag{3.7}$$

$$std = \sqrt{\frac{\sum_{i=0}^{255}(i-\mu)^2 * pf(i)}{\sum_{i=0}^{255} pf(i) - 1}} \tag{3.8}$$

$$Tk = \frac{std}{2} \tag{3.9}$$

Setelah data-data awal yang dibutuhkan untuk proses *clustering* diperoleh, metode k-means *clustering* kemudian dilakukan untuk mengelompokkan piksel-piksel sesuai dengan tingkat keabuannya. Proses *clustering* dengan metode k-means *clustering* dijalankan pada piksel-piksel yang terdapat pada citra *input*. Piksel-piksel yang telah dikelompokkan kemudian dirubah tingkat keabuannya sesuai dengan *centroid* dari clusternya. *Clustering*

dan perubahan tingkat keabuan ini bisa membuat tingkat keabuan citra menjadi lebih sederhana.

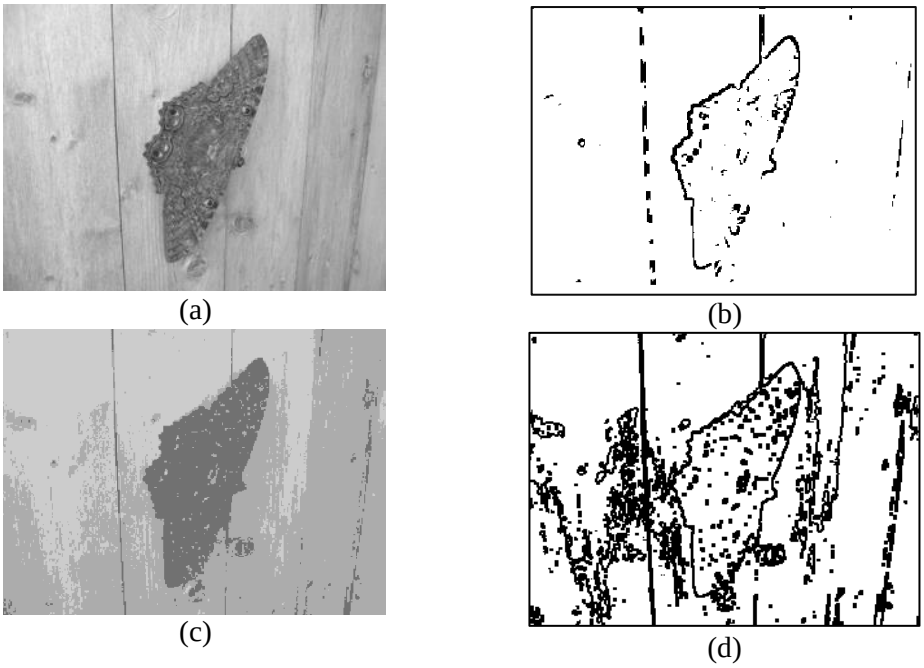


Gambar 3.7 *Flowchart* Pengelompokan Piksel-Piksel Citra dengan K-Means *Clustering*

Perubahan tingkat keabuan pada citra yang menjadi lebih sederhana atau sedikit dapat menghasilkan *transition region* yang lebih banyak. Hal ini disebabkan karena terdapat perbedaan nilai *local variance* yang tinggi pada piksel yang berada di tengah dengan piksel yang berada di tepi *foreground* atau *background*. Piksel-piksel yang berada di tengah *foreground* atau *background*

memiliki nilai *local variance* 0 karena piksel-piksel yang berada satu *local neighborhood window* dengannya memiliki tingkat keabuan yang sama hasil dari pengelompokan piksel. Sedangkan piksel-piksel yang ada di tepi *foreground* atau *background* memiliki nilai *local variance* yang tinggi akibat terdapat perubahan tingkat keabuan di area *local neighborhood window*.

Proses ini perlu dilakukan dengan harapan dapat membuat *transition region* yang dihasilkan menjadi lebih utuh karena *transition region* bisa dibangkitkan dengan baik pada citra dengan tingkat keabuan yang sederhana (Parida & Bhoi, 2016). Penjelasan lebih detail tentang metode k-means *clustering* bisa dilihat pada Sub bab 2.2.

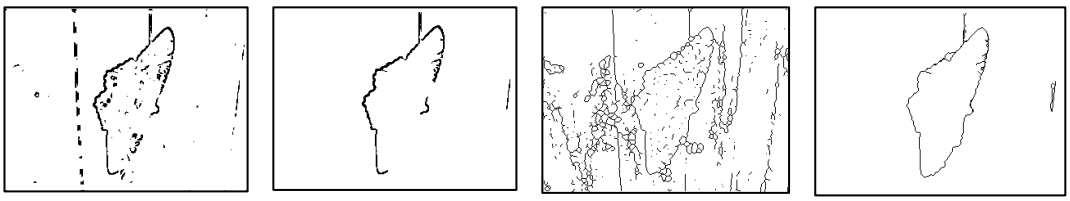


Gambar 3.8 Citra dan *Transition Region* (a) Citra Asli, (b) *Transition Region* Citra Asli, (c) Citra Sederhana, dan (d) *Transition Region* Citra Sederhana

3.2.4. *Edge Linking* pada Transition Region

Kualitas *transition region* yang dihasilkan tergantung pada kompleksitas tingkat keabuan pada citra *input*. *Transition region* akan bisa diekstraksi dengan bagus pada citra dengan tingkat keabuan yang sederhana. Namun, pada citra dengan variasi tingkat keabuan yang tinggi, *transition region* mungkin akan sulit

untuk dihasilkan secara utuh. Sehingga proses *edge linking* sangat diperlukan agar *transition region* yang pecah bisa disambungkan dan menjadi lebih utuh.



(a) (b) (c) (d)

Gambar 3.9 Ilustrasi Proses *Edge Linking* pada *Transition Region* (a) *Transition Region* Citra Asli, (b) *Transition Region* Hasil Proses Perhitungan Irisan, (c) Rute Sambungan dari *Transition Region* Citra Sederhana, dan (d) *Transition Region* yang Telah Tersambung

Proses *edge linking* pada penelitian ini sedikit berbeda dengan *edge linking* pada umumnya. *Edge linking* pada umumnya digunakan untuk menghubungkan dua ujung yang terpotong dengan cara menambahkan garis yang terhubung dengan kedua ujung seperti metode *curve-fitting edge linking* dan *heuristic edge-linking methods* (Pratt, 2001). Metode-metode tersebut tidak digunakan dalam penelitian ini karena *edge linking* pada penelitian ini lebih mirip dengan masalah pencarian rute. Sambungan atau rute yang bisa digunakan untuk menghubungkan dua ujung *transition region* dari citra asli yang terpisah sudah tersedia di *transition region* dari citra sederhana. Sehingga masalah yang perlu diselesaikan oleh *edge linking* pada penelitian ini adalah memilihkan jalur rute sambungan yang cocok dan bisa menghubungkan dua ujung *transition region* yang terpisah. Untuk mengatasi masalah tersebut maka metode A* digunakan untuk proses *edge linking* pada penelitian ini karena metode A* merupakan metode yang dapat menemukan rute dengan optimal, sehingga diharapkan rute *linking* yang dipilih tidak terlalu melebar dari objek.

Edge linking hanya dilakukan pada *transition region* citra asli yang belum utuh. Penentuan *transition region* yang belum utuh dilakukan dengan menggunakan *hole filling* pada *transition region* citra asli. Apabila jumlah piksel setelah ditambahkan oleh proses *hole filling* pada suatu *transition region* lebih

banyak daripada jumlah piksel kosong di areanya, maka *transition region* tersebut dianggap sebagai sudah utuh. Jika sebaliknya maka dianggap sebagai belum utuh.

Proses *edge linking* pada *transition region* yang belum utuh menggunakan *transition region* dari citra sederhana sebagai penyambungnya. *Transition region* dari citra asli dan sederhana memiliki kelebihan, kekurangan, dan fungsi yang berbeda yang dijelaskan di Tabel 3.1.

$$sa_l = \sum_{x \in ta_l} \sum_{y \in ta_l} sp = ta(x,y) * ts(x,y) \tag{3.10}$$

$$pa_l = \left(\frac{sa_l}{la_l}\right) * 100 \tag{3.11}$$

fase *edge linking* untuk *transition region* yang belum utuh menggunakan citra *transition region* dari citra asli ta dan citra *transition region* dari citra sederhana ts kemudian disusun berdasarkan data pada Tabel 3.1. Terdapat beberapa proses dalam fase ini seperti yang terlihat pada Gambar 3.10.

Tabel 3.1 Kelebihan, Kekurangan, dan Fungsi Dua Citra *Transition Region*

	Transition region dari citra asli	**Transition region dari citra sederhana**
Kekurangan	Sulit untuk menghasilkan *transition region* yang utuh pada citra dengan tingkat keabuan yang kompleks.	Sulit menentukan letak objek karena terlalu banyak menghasilkan *transition region*.
Kelebihan	Mudah untuk menentukan posisi objek karena mayoritas *transition region* diekstraksi di sekitar objek.	Dapat menghasilkan *transition region* yang lebih utuh karena citra yang digunakan telah disederhanakan.
Fungsi dalam sistem ini	Sebagai *transition region* yang akan disambungkan dan digunakan dalam proses segmentasi.	Sebagai penyambung *transition region* dari citra asli yang putus.

Proses penentuan *transition region* citra asli ta_l yang akan digunakan untuk *edge linking* dilakukan dengan cara menghitung persentase piksel transition region citra asli yang beririsan dengan piksel transition region citra sederhana ts_l pa_l. Dalam proses ini, terlebih dahulu dihitung jumlah piksel yang beririsan sa_l

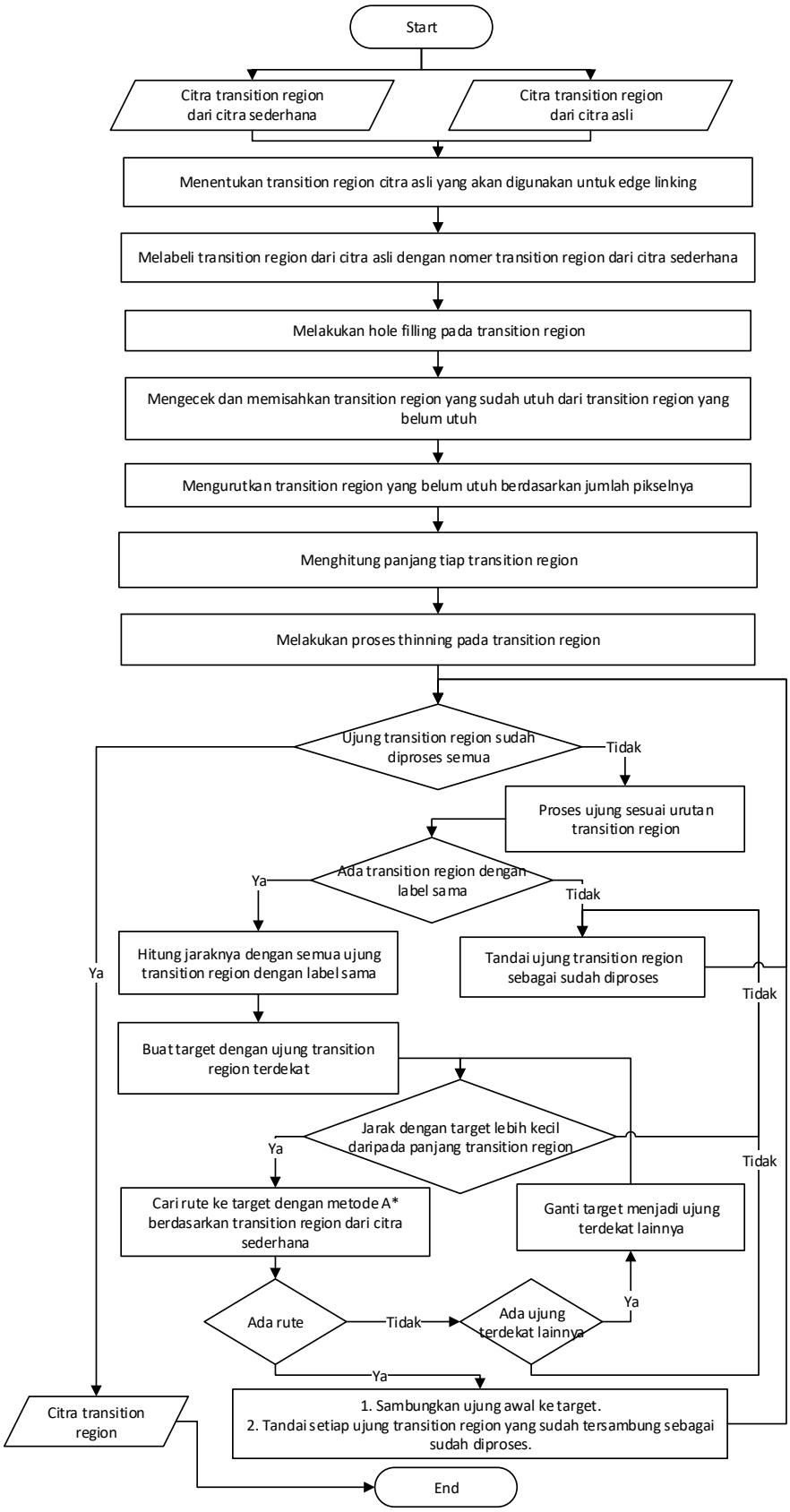
antara *transition region* ta_l dan ts_l menggunakan Persamaan 3.10. Setiap piksel pada transition region citra asli $ta(x, y)$ dibandingkan dengan piksel pada transition region citra sederhana $ts(x, y)$. Nilai perbandingan tiap piksel sp akan bernilai 1 apabila kedua piksel yang dibandingkan merupakan *transition region* pada citra asli dan sederhana. Kemudian setiap nilai sp dijumlahkan untuk mendapatkan jumlah piksel *transition region* ta_l dan ts_l yang beririsan.

$$ta_l = \begin{cases} 1 & pa_l \geq Ti \\ 0 & sebaliknya \end{cases}, \tag{3.12}$$

Persentase perpotongan pa_l dihitung dengan cara membagi nilai sa_l oleh jumlah piksel pada *transition region* ta_l la_l dan dikalikan 100 seperti Persamaan 3.11. Kemudian nilai pa_l dibandingkan dengan *threshold* irisan *transition region* Ti seperti Persamaan 3.12. Threshold irisan transition region Ti menggunakan nilai 50% sesuai dengan hasil uji coba. Apabila nilai pa_l sama atau lebih besar dari Ti maka *transition region* tersebut menjadi anggota *transition region* yang akan diproses dengan *edge linking*. Sedangkan *transition region* ta_l yang tidak memenuhi kriteria akan diproses lagi untuk menentukan persentase irisan seperti proses sebelumnya tetapi hanya menggunakan *border* atau tepi *transition region* citra asli saja. Apabila nilai pa_l yang dihasilkan oleh proses ini masih lebih kecil dari Ti maka *transition region* tersebut akan dihapus sehingga akan mengurangi *transition region* yang perlu diproses *edge linking* seperti yang terlihat pada Gambar 3.9 (b).

Transition region ta_l akan dilabeli dengan nomer *transition region* citra sederhana yang beririsan dengannya. *Transition region* tersebut kemudian diurutkan secara *descending* sesuai jumlah pikselnya. Selanjutnya, panjang setiap transition region pta_l dihitung menggunakan Persamaan 3.13. Proses *thinning* dilakukan untuk menipiskan dan mendapatkan ujung *transition region*. Ujung-ujung tersebut nantinya akan disambungkan satu sama lain sesuai dengan jarak dan kesamaan visualnya yang didapat dari *transition region* citra sederhana.

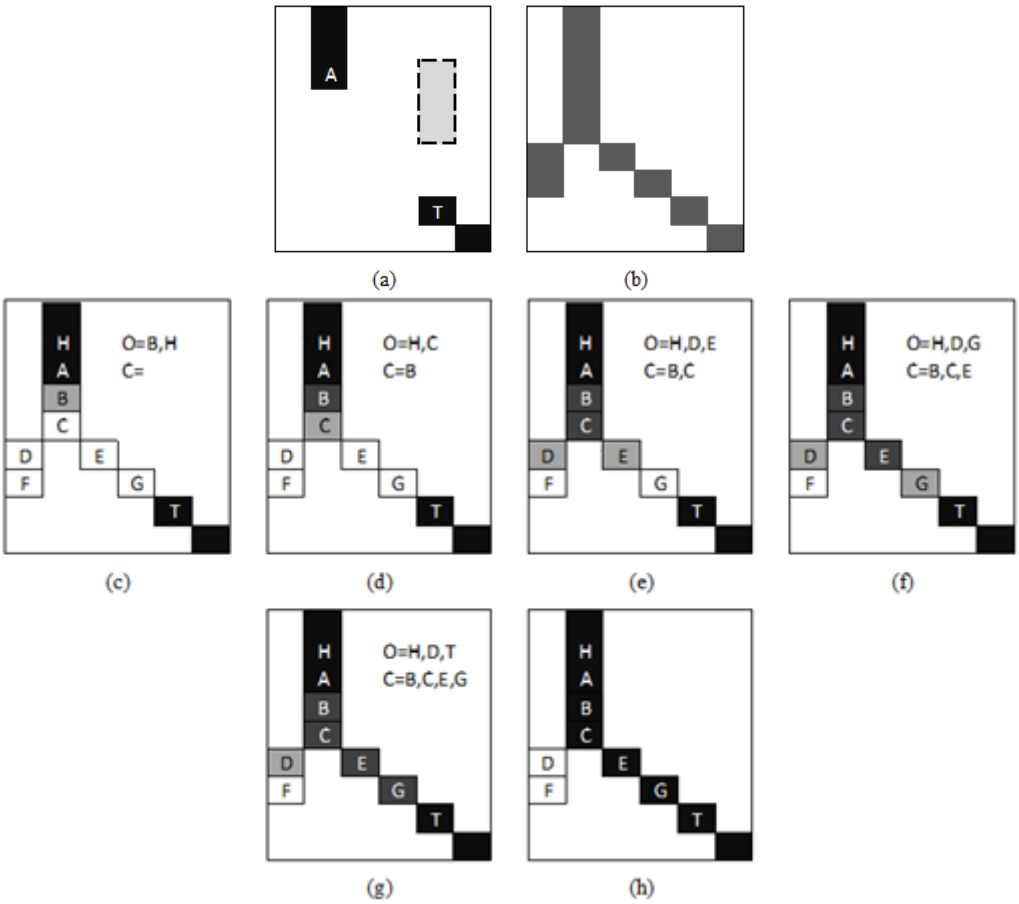
$$pta_l = \sqrt{(bb_l - bt_l)^2 + (br_l - bl_l)^2} \tag{3.13}$$



Gambar 3.10 *Flowchart* Proses *Edge Linking*

Proses penyambungan tiap ujung *transition region* dilakukan secara berurutan sesuai urutan *transition region* yang diperoleh sebelumnya. Dalam proses ini, jarak antara ujung *transition region* yang akan disambungkan sebagai titik awal dengan ujung *transition region* lain yang memiliki label sama sebagai kandidat target ju_t dihitung dengan mengurangi koordinat piksel titik awal (*xa*, *ya*) dengan koordinat piksel target (xt_t, yt_t) menggunakan Persamaan 3.14. Namun, apabila tidak ada *transition region* lain yang berlabel sama maka ujung tersebut ditandai sudah diproses dan dilanjutkan dengan ujung lainnya.

$$ju_t = \sqrt{(xa - xt_t)^2 + (ya - yt_t)^2} \tag{3.14}$$

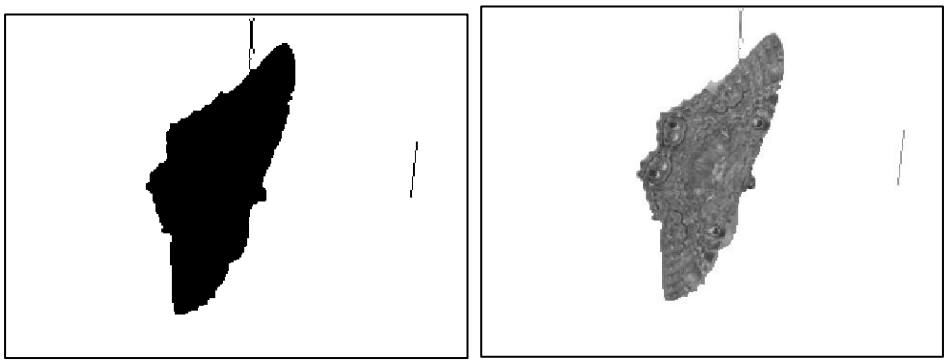


Gambar 3.11 Ilustrasi Penyambungan dengan A*, (a) *Transition Region* Citra Asli, (b) *Transition Region* Citra Sederhana, (c)-(g) Langkah Pencarian Rute A*, (h) Hasil Penyambungan *Transition Region* Berdasarkan Rute dari A*. Kotak Berwarna Hitam adalah *Transition Region*, Kotak Berwarna Abu-Abu adalah Kandidat Rute, Kotak Berwarna Putih dan Bergaris Hitam adalah Kandidat yang Tidak Dipakai, dan Kotak Dengan Garis Putus-Putus adalah Transition Region dengan Label Berbeda.

Proses pencarian rute dengan A* dari titik awal ke target yang paling dekat dan berjarak lebih kecil dari panjang *transition region* titik awal dilakukan seperti yang diilustrasikan pada Gambar 3.11. Jika rute ditemukan maka proses penyambungan dari titik awal ke target tersebut dilakukan dan tandai ujung yang telah tersambung menjadi sudah diproses. Namun, jika rute tidak ditemukan maka target dirubah menjadi target terdekat lainnya dan berjarak lebih kecil dari panjang *transition region* titik awal, kemudian dilakukan proses pencarian rute lagi. Proses pencarian rute ini dilakukan terus sampai semua ujung sudah diproses. Hasil dari proses ini diharapkan dapat menghasilkan *transition region* yang lebih utuh seperti Gambar 3.9 (d).

3.2.5. *Region Filling* Transition Region

Transition region yang sudah utuh dan belum utuh yang telah melalui proses *edge linking* kemudian digabungkan. *Transition region* hasil penggabungan tersebut kemudian menjalani proses *region filling* untuk mengisi lubang atau areanya sehingga diperoleh citra biner yang di dalamnya terdapat area atau *region* dari objek seperti yang terlihat pada Gambar 3.12 (a).



(a) (b)

Gambar 3.12 (a) Citra Hasil *Region Filling*, (b) Citra Hasil Segmentasi

3.2.6. Proses Pemisahan Objek dan *Background*

Tahap terakhir dari sistem ini adalah proses pemisahan objek dan *background*. Proses ini menggunakan citra hasil *region filling* yang telah dihasilkan oleh proses sebelumnya. Proses segmentasi ini dilakukan dengan cara menampilkan piksel-piksel dari citra *grayscale* yang beririsan dengan area

transition region pada citra biner. Piksel-piksel yang ditampilkan merupakan piksel yang dianggap sebagai objek oleh sistem ini, sedangkan piksel-piksel yang tidak ditampilkan adalah *background*. Hasil dari proses ini dapat dilihat pada Gambar 3.12 (b).

3.3. Skenario Pengujian

Uji coba dilakukan untuk menganalisa performa dari system yang dilakukan dengan beberapa tahap. Pertama, citra *grayscale* dan citra *ground truth* atau citra hasil segmentasi manual dari citra tersebut disiapkan untuk pengujian sistem. Kemudian citra *grayscale* dimasukkan ke sistem. Citra tersebut akan disegmentasi oleh sistem dan sistem akan menghasilkan citra segmentasi yang berisi objek tunggal atau banyak. Nilai *misclassification error* (ME), *false positive rate* (FPR), *false negative rate* (FNR) dihitung untuk citra tersebut berdasarkan citra *ground truth*. Nilai ME, FPR, dan FNR hasil dari sistem kemudian dibandingkan dengan hasil dari penelitian Parida & Bhoi (2016).

Proses pengujian juga dilakukan untuk keperluan penentuan apakah *filtering* dibutuhkan oleh sistem ini ataukah tidak. Pengujian ini dilakukan dengan cara melalukan dua kali proses segmentasi pada setiap citra. Proses segmentasi yang pertama dilakukan dengan menggunakan *filtering* pada semua citra, sementara proses segmentasi yang kedua dilakukan tanpa *filtering.* Nilai ME, FPR, dan FNR dari setiap proses segmentasi kemudian dihitung. Kemudian dilakukan perbandingan hasil segmentasi yang menggunakan *filtering* dan tanpa *filtering*.

Proses pengujian kemudian dilakukan untuk menentukan nilai yang cocok untuk beberapa parameter: parameter ukuran *window* untuk proses *filtering uw*, parameter jarak maksimal *transitional pixels* yang dianggap berdekatan *jt*, parameter untuk menentukan persentase *transitional pixels* dekat yang normal *TPn*, dan parameter untuk menentukan *transition region* citra asli yang sedikit beririsan dengan *transition region* citra sederhana *Ti*. Proses pengujian ini dilakukan pada 13 citra dari penelitian Parida & Bhoi (2016) yang memiliki karakteristik yang berbeda-beda. Segmentasi dilakukan dengan menggunakan

kombinasi 4 parameter tersebut. Hasil terbaik dari pengujian tersebut akan digunakan sebagai nilai-nilai parameter yang paling optimal.

3.4. Dokumentasi dan Jadwal Penelitian

Tabel 3.2 menunjukkan jadwal kegiatan penelitian sistem ini. Jadwal tersebut ditampilkan per-minggu selama 5 bulan.

Tabel 3.2 Jadwal Kegiatan Penelitian

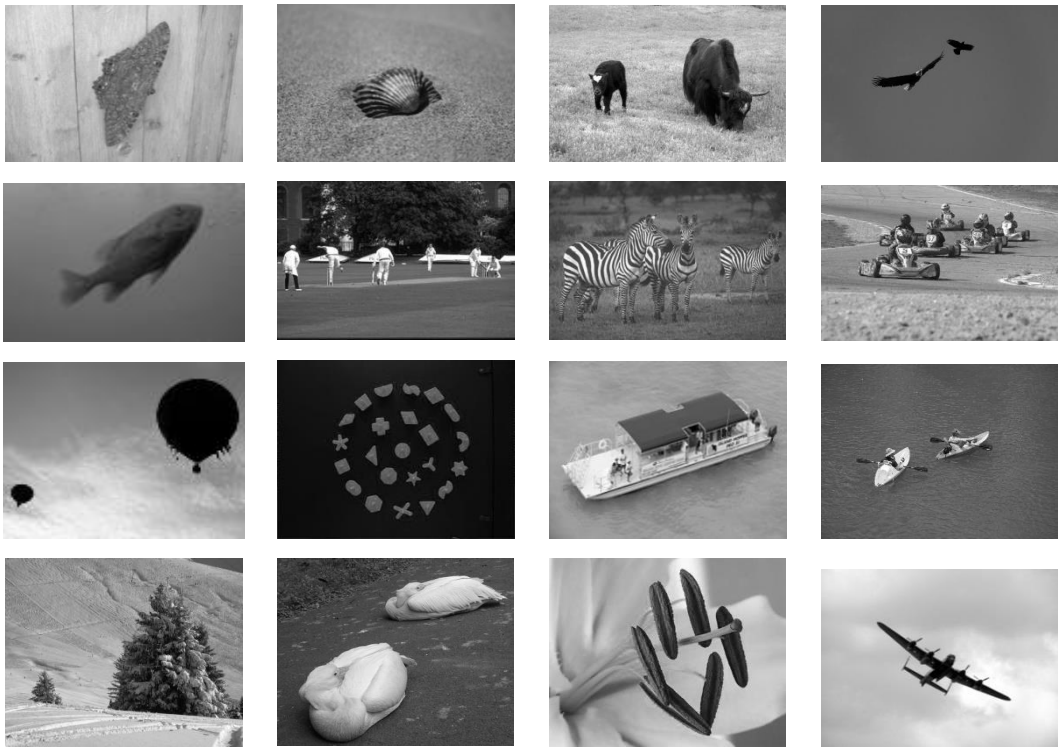
No.	Kegiatan	Bulan																			
		Januari				Pebruari				Maret				April				Mei			
1.	Studi literatur																				
2.	Desain sistem																				
3.	Pembuatan perangkat lunak																				
4.	Uji coba																				
5.	Analisa hasil																				
6.	Penyusunan buku tesis																				

BAB 4
HASIL DAN PEMBAHASAN

Pada bab ini dijelaskan tentang hasil pengujian terhadap penelitian yang dilakukan. Selanjutnya dilakukan analisa terhadap hasil pengujian untuk mendapatkan informasi yang akan menjadi kesimpulan penelitian.

4.1. Lingkungan Pengujian dan Dataset

Penelitian diimplementasikan menggunakan *hardware* dengan spesifikasi prosesor Intel core i3 2.20 GHz dan RAM 4GB. Perangkat lunak yang digunakan adalah Matlab R2017a dan sistem operasi Windows 10. Dataset yang digunakan adalah dataset Weizmann yang berisi citra dengan objek satu dan ganda, MSRA dan Barkeley dataset yang berisi citra dengan banyak objek, dan beberapa citra input yang digunakan oleh penelitian Parida & Bhoi (2016). Total dari citra yang digunakan adalah sebanyak 128 buah. Beberapa citra dari dataset ditunjukkan pada Gambar 4.1.

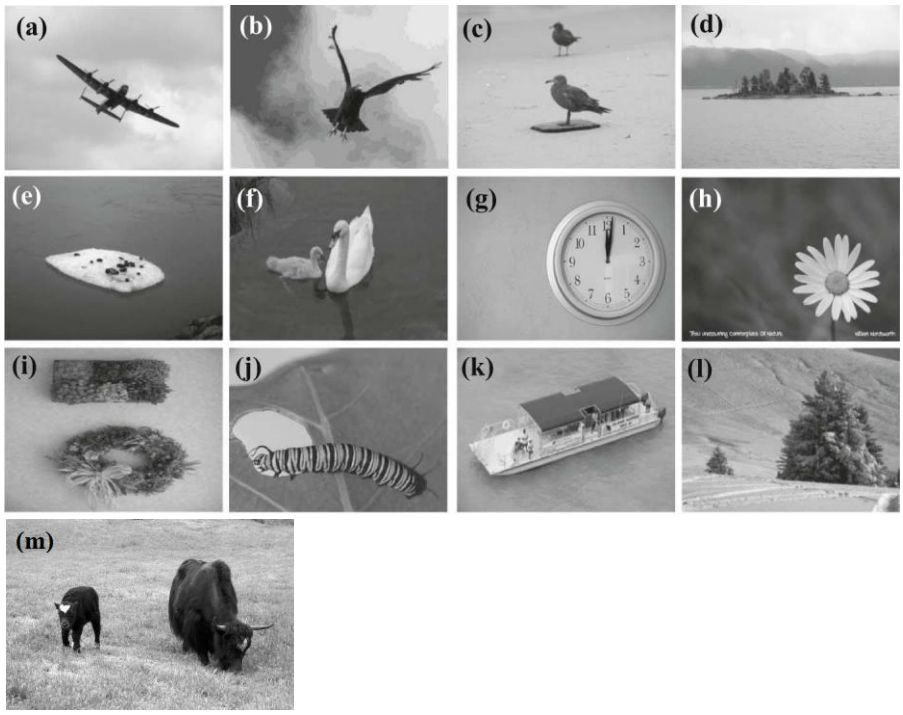


Gambar 4.1 Contoh Citra dari Dataset Weizmann, MSRA, dan Barkeley

4.2. Hasil Pengujian dan Pembahasan

Dalam penelitian ini, proses pengujian dilakukan untuk menentukan keperluan menggunakan *filtering* dan pencarian nilai parameter yang cocok sekaligus penentuan kinerja sistem yang diusulkan. Terdapat beberapa tahap dalam pengujian sistem ini. Pertama, citra *grayscale* dimasukkan ke sistem. Citra tersebut kemudian disegmentasi oleh sistem dan sistem akan menghasilkan citra segmentasi yang berisi objek tunggal atau banyak. Nilai ME, FPR, dan FNR hasil segmentasi kemudian dihitung berdasarkan citra *ground truth*.

4.2.1. Hasil Pengujian dan Pembahasan Keperluan Menggunakan *Filtering*



Gambar 4.2 Citra Uji untuk Pengujian Keperluan Menggunakan Filtering (a) *Airplane*, (b) *Eagle*, (c) *Bird*, (d) *Island*, (e) *Iceberg*, (f) *Duck*, (g) *Clock*, (h) *Flower*, (i) *Wall décor*, (j) *Caterpillar*, (k) *Boat*, (l) *Mountain-tree*, (m) *yack*

Pengujian ini dilakukan untuk menentukan apakah *smoothing* citra menggunakan teknik *median filter* dibutuhkan dalam sistem ini. Pengujian dilakukan menggunakan beberapa citra dari penelitian Parida & Bhoi (2016) yang sudah didefinisikan karakteristik *background* dan *foreground* setiap citra dan ditunjukkan pada Gambar 4.2.

Terdapat 4 karakteristik pada citra input:

1. *Simple background & simple foreground.* Beberapa citra dengan karakteristik ini adalah *airplane, eagle* dan *bird.*
2. *Textured background & simple foreground.* Beberapa citra dengan karakteristik ini adalah *island, iceberg* dan *duck.*
3. *Simple background & textured foreground.* Beberapa citra dengan karakteristik ini adalah *clock , flower* dan *decorwall.*
4. *Textured foreground & textured background.* Beberapa citra dengan karakteristik ini adalah *caterpiller, boat, mountain trees* dan *yack1*.

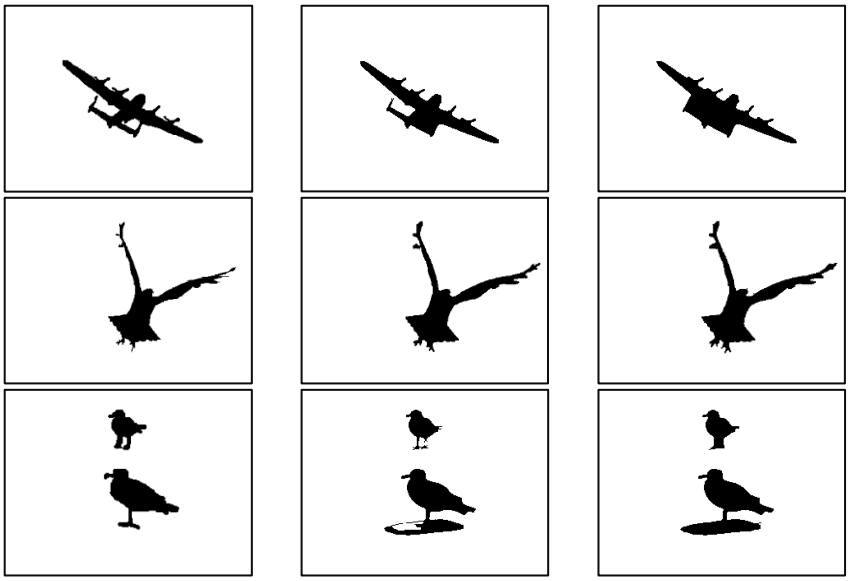
Pengujian dilakukan dengan cara membandingkan hasil segmentasi dari sistem yang tanpa menggunakan *filtering* dengan yang ditambah proses *filtering*.

Tabel 4.1 Hasil Pengujian Penggunaan *Filtering*

Citra input	**ME**		**FPR**		**FNR**	
	Filter	**tanpa**	**filter**	**tanpa**	**filter**	**tanpa**
(simple background & simple foreground)						
airplane	**0.75%**	1.2%	**0.39%**	0.96%	6.72%	**5.24%**
eagle	**0.84%**	1%	**0.87%**	1.05%	0.29%	**0.02%**
bird	**2.04%**	2.38%	**1.5%**	1.98%	11.25%	**9.25%**
(textured background & simple foreground)						
island	1.26%	**0.93%**	**0.59%**	0.68%	10%	**4.16%**
iceberg	**0.36%**	0.49%	**0.16%**	0.32%	2.25%	**2.09%**
duck	**1.34%**	3.09%	**0.58%**	2.64%	6.82%	**6.27%**
(simple background & textured foreground)						
clock	1.19%	**0.51%**	**0.21%**	0.56%	3.17%	**0.41%**
flower	**2.05%**	2.6%	**2.36%**	2.99%	**0%**	**0%**
decorwall	13.53%	**4.03%**	**1.28%**	2.55%	35.71%	**6.71%**
(textured foreground & textured background)						
caterpiller	10.21%	**10.14%**	**9.68%**	11.03%	13.6%	**4.38%**
boat	9.96%	**1.24%**	**0.07%**	0.3%	33.02%	**3.42%**
mountain trees	**3.53%**	7.15%	**2.83%**	9.01%	5.56%	**1.78%**
yack1	**1.51%**	18.23%	**1.64%**	21.58%	0.82%	**0.35%**
Rata-rata	**3.74%**	4.07%	**1.7%**	4.28%	9.94%	**3.39%**

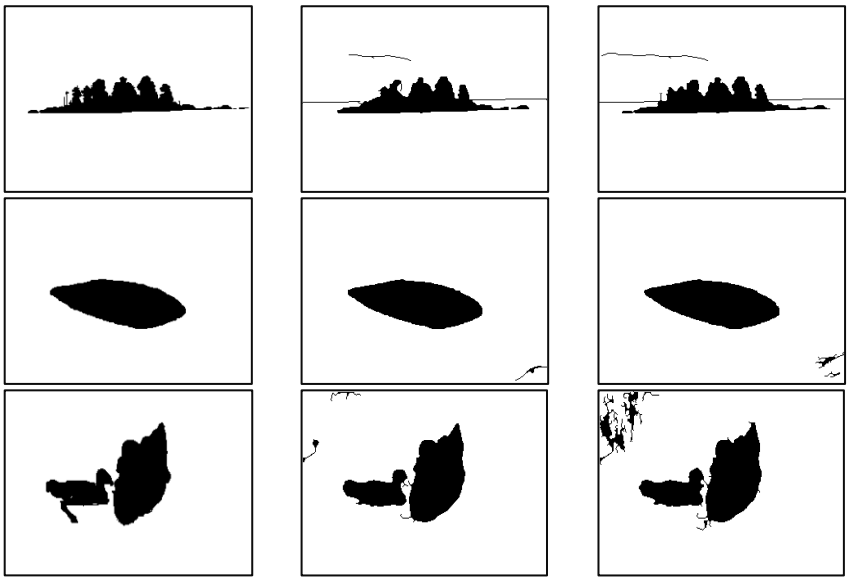
Hasil dari pengujian ini dapat dilihat secara detail pada Tabel 4.1. Tabel 4.1 menunjukkan bahwa penggunaan *filtering* pada sistem ini terbukti mampu menghasilkan nilai FPR yang lebih bagus daripada hasil dari sistem tanpa *filtering* untuk semua citra. Hal tersebut menunjukkan bahwa penggunaan *filtering* dapat mengurangi kesalahan klasifikasi pada *background* karena citra telah dihaluskan, sehingga perubahan tingkat keabuan antar piksel menjadi lebih halus yang mengakibatkan tekstur dari *background* menjadi lebih sederhana seperti yang ditunjukkan pada Gambar 4.3 sampai 4.6. Pada Gambar tersebut terlihat bahwa *background* pada semua citra terlihat lebih bersih daripada citra hasil dari sistem tanpa *filtering.* Hasil paling mencolok terlihat pada hasil dari citra *yack* dan *mountain trees.* Kedua citra tersebut merupakan citra dengan *background* yang kompleks atau bertekstur. Berdasarkan hasil pengujian tersebut, terbukti bahwa penggunaan *filtering* dalam sistem ini cocok digunakan untuk citra dengan *background* yang kompleks atau bertekstur.

Meskipun begitu, penggunaan *filtering* ternyata memiliki hasil yang lebih buruk jika dilihat dari nilai FNR yang menyatakan kesalahan klasifikasi *foreground. Filtering* yang memiliki kemampuan untuk menghaluskan citra secara otomatis akan menghaluskan juga *foreground* pada citra. Penghalusan pada *foreground* dapat mengakibatkan tekstur *foreground* khususnya area tepi *foreground* menjadi lebih halus. Sehingga hal tersebut dapat mengakibatkan melesetnya hasil segmentasi pada tepi *foreground* seperti yang ditunjukkan pada Gambar 4.5 *décorwall* dan Tabel 4.1. Pada tabel tersebut terlihat bahwa hampir semua citra memiliki nilai FNR yang lebih baik pada sistem tanpa *filtering.* Bahkan pada beberapa citra terlihat bahwa penggunaan *filtering* mengakibatkan *foreground* tersegmentasi dengan buruk atau sebagian area *foreground* terklasifikasi sebagai *background*, seperti citra *decorwall, caterpillar,* dan *boat.* Citra-citra tersebut merupakan citra yang memiliki *background* dengan tekstur yang lebih sederhana daripada tekstur *foreground*. Hal ini membuktikan bahwa *filtering* tidak dibutuhkan pada citra dengan kondisi *background* yang sederhana (tekstur lebih sederhana daripada tekstur *foreground*). Karena walaupun tanpa *filtering,* sistem sudah bisa menghasilkan *transition region* yang mengelilingi *foreground.*



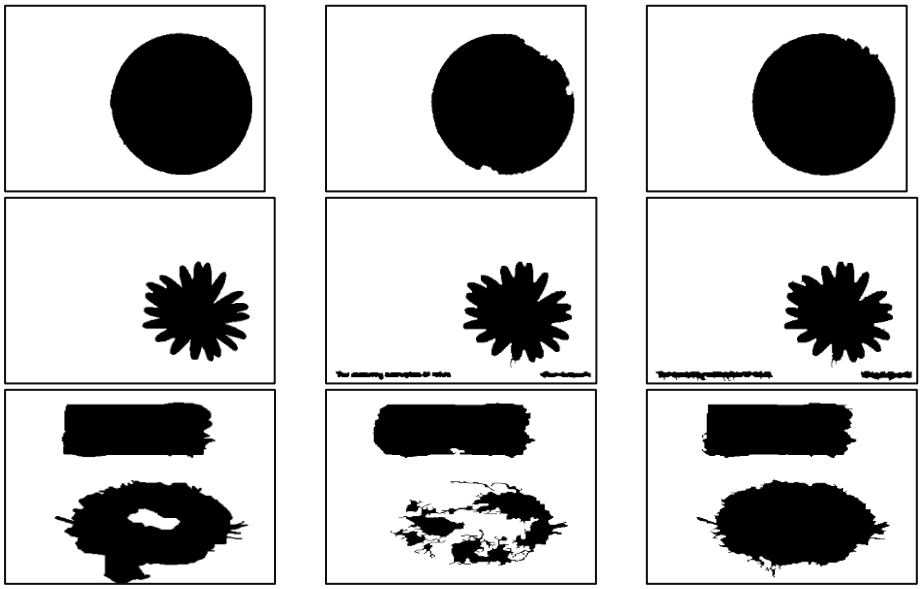
(a) (b) (c)

Gambar 4.3 *Ground Truth* dan *Transition Region* untuk Citra *Simple Background & Simple Foreground: (a) Ground Truth,* (b) Menggunakan *Filtering,* (c) tanpa *Filtering*



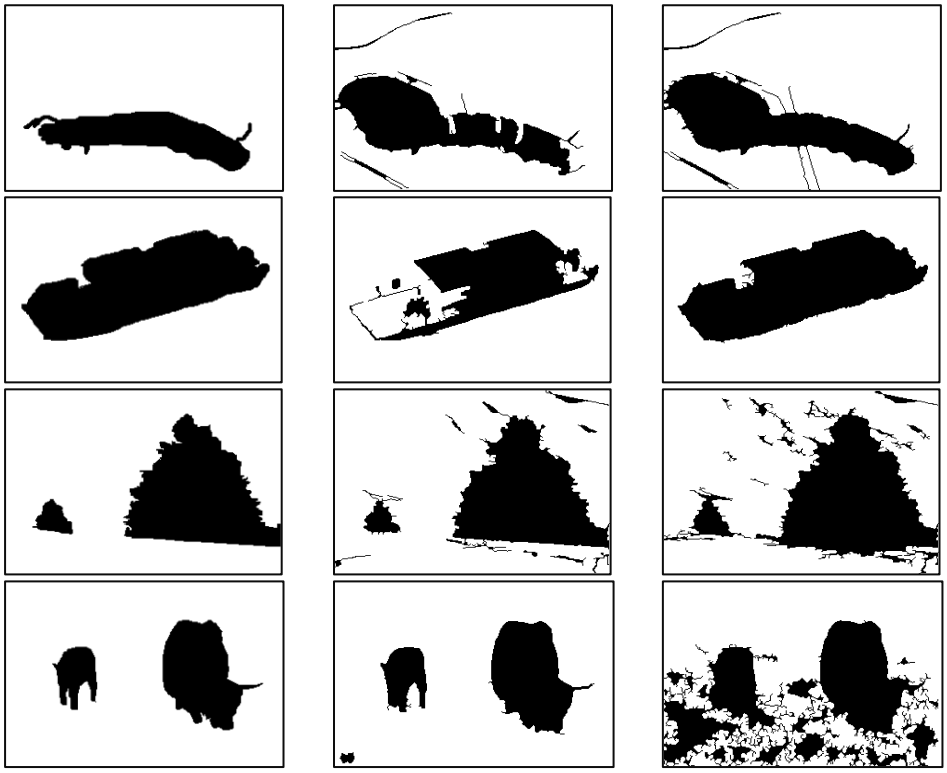
(a) (b) (c)

Gambar 4.4 *Ground Truth* dan *Transition Region* untuk Citra *Textured Background & Simple Foreground: (a) Ground Truth,* (b) Menggunakan *Filtering,* (c) tanpa *Filtering*



(a) (b) (c)

Gambar 4.5 *Ground Truth* dan *Transition Region* untuk Citra *Simple Background & Textured Foreground: (a) Ground Truth,* (b) Menggunakan *Filtering,* (c) tanpa *Filtering*



(a) (b) (c)

Gambar 4.6 *Ground Truth* dan *Transition Region* untuk Citra *Textured Foreground & Textured Background: (a) Ground Truth,* (b) Menggunakan *Filtering,* (c) tanpa *Filtering*

4.2.2. Hasil Pengujian dan Pembahasan Pencarian Nilai Parameter Yang Cocok

Hasil pengujian sebelumnya menunjukkan bahwa penggunaan *filtering* pada sistem ini bersifat kondisional. *Filtering* sangatlah dibutuhkan pada citra dengan kondisi *background* yang kompleks atau bertekstur. Sebaliknya, *filtering* tidak dibutuhkan pada citra dengan kondisi *background* yang sederhana. Untuk menentukan apakah *filtering* perlu dijalankan ataukah tidak, sistem ini menambahkan sebuah metode baru untuk menghitung persentase *transitional pixels* yang dekat.

Filtering akan dijalankan pada citra yang memiliki nilai persentase *transitional pixels* dekat yang melebihi nilai *threshold* normal. Dikarenakan penentuan nilai *threshold* tersebut masih manual, maka perlu dilakukan uji coba untuk menentukan nilai parameter yang cocok. Terdapat empat parameter yang perlu dicari nilai optimalnya: parameter ukuran *window* untuk proses *filtering uw*, parameter jarak maksimal *transitional pixels* yang dianggap berdekatan *jt*, parameter untuk menentukan persentase *transitional pixels* dekat yang normal *TPn*, dan parameter untuk menentukan *transition region* citra asli yang sedikit beririsan dengan *transition region* citra sederhana *Ti*.

Proses pengujian ini dilakukan pada 13 citra dari penelitian Parida & Bhoi (2016) yang memiliki karakteristik yang berbeda-beda. Kandidat nilai *jt* ditentukan, yaitu berjarak 1 sampai 6 piksel. Kandidat nilai *uw* adalah 3 dan 5. Kandidat nilai *TPn* yaitu antara 4 sampai 8%. Dan kandidat nilai *Ti* adalah 10% sampai 90%.

Tabel 4.2 sampai 4.12 berisi data-data dengan nilai ME terbaik pada setiap pengujian. Detail hasil setiap pengujian bisa dilihat pada Lampiran A. Nilai rata-rata FPR paling kecil pada pengujian ini adalah 1.73% yang diperoleh dengan melakukan *filtering* hampir pada seluruh citra, yaitu 11 buah citra *input* serta melakukan penghapusan pada *transition region* dengan irisan kurang dari 60%. Hal tersebut menunjukkan bahwa penggunaan *filtering* pada citra dapat mengurangi kesalahan klasifikasi *background.*

Tabel 4.2 Hasil dari Jarak Kedekatan Maksimal 1 Piksel dengan Ukuran *Window Filtering* 3

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
3	50	4.07%	4.28%	3.39%	0
4	50	4.07%	4.28%	3.39%	0
5	50	4.07%	4.28%	3.39%	0
6	50	4.07%	4.28%	3.39%	0
7	50	4.07%	4.28%	3.39%	0
8	50	4.07%	4.28%	3.39%	0

Tabel 4.3 Hasil dari Jarak Kedekatan Maksimal 2 Piksel dengan Ukuran *Window Filtering* 3

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
3	50	3.11%	**2.01%**	5.99%	4
4	50	**2.37%**	2.11%	3.76%	3
5	50	2.65%	2.59%	3.47%	2
6	50	2.79%	2.74%	**3.43%**	1
7	50	2.79%	2.74%	**3.43%**	1
8	50	2.79%	2.74%	**3.43%**	1

Tabel 4.4 Hasil dari Jarak Kedekatan Maksimal 3 Piksel dengan Ukuran *Window Filtering* 3

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
4	50	3.19%	**1.88%**	7.36%	7
5	50	2.45%	2.08%	4.42%	5
6	50	**2.37%**	2.11%	3.76%	3
7	50	2.65%	2.59%	**3.47%**	2
8	50	2.65%	2.59%	**3.47%**	2

Tabel 4.5 Hasil dari Jarak Kedekatan Maksimal 4 Piksel dengan Ukuran *Window Filtering* 3

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
4	30	3.85%	**1.85%**	9.62%	9
5	30	3.18%	1.87%	7.35%	8
6	50	2.45%	2.08%	4.42%	5
7	50	2.4%	2.1%	4.21%	4
8	50	**2.37%**	2.11%	3.76%	3
9	50	2.65%	2.59%	**3.47%**	2

Tabel 4.6 Hasil dari Jarak Kedekatan Maksimal 5 Piksel dengan Ukuran *Window Filtering* 3

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
4	30	3.8%	**1.8%**	9.76%	11
5	30	3.18%	1.87%	7.35%	8
6	50	2.45%	1.96%	5.14%	7
7	50	2.45%	2.08%	4.42%	5
8	50	2.45%	2.08%	4.42%	5
9	50	**2.37%**	2.11%	3.76%	0
10	50	2.65%	2.59%	**3.47%**	0

Tabel 4.7 Hasil dari Jarak Kedekatan Maksimal 6 Piksel dengan Ukuran *Window Filtering* 3

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
4	30	3.8%	**1.8%**	9.76%	11
5	30	3.84%	1.84%	9.64%	10
6	50	2.45%	1.96%	5.14%	7
7	50	**2.44%**	2.06%	4.43%	6
8	50	2.45%	2.08%	**4.42%**	5

Tabel 4.8 Hasil dari Jarak Kedekatan Maksimal 2 Piksel dengan Ukuran *Window Filtering* 5

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
3	60	3.26%	2.01%	6.94%	4
4	50	**2.64%**	**2%**	5.2%	3
5	50	2.7%	2.6%	3.66%	2
6	50	2.84%	2.78%	**3.53%**	1
7	50	2.84%	2.78%	**3.53%**	1
8	50	2.84%	2.78%	**3.53%**	1

Tabel 4.9 Hasil dari Jarak Kedekatan Maksimal 3 Piksel dengan Ukuran *Window Filtering* 5

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
4	60	3.78%	**1.84%**	11.64%	7
5	50	2.98%	1.95%	7.75%	5
6	50	**2.64%**	2%	5.2%	3
7	50	2.7%	2.6%	**3.66%**	2
8	50	2.7%	2.6%	**3.66%**	2
9	50	2.7%	2.6%	**3.66%**	2

Tabel 4.10 Hasil dari Jarak Kedekatan Maksimal 4 Piksel dengan Ukuran *Window Filtering* 5

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
4	60	4.57%	**1.81%**	14.34%	9
5	60	3.78%	1.83%	11.68%	8
6	50	2.98%	1.95%	7.75%	5
7	50	2.76%	1.98%	6.99%	4
8	50	**2.64%**	2%	5.2%	3
9	50	2.7%	2.6%	**3.66%**	2

Tabel 4.11 Hasil dari Jarak Kedekatan Maksimal 5 Piksel dengan Ukuran *Window Filtering* 5

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
4	60	4.52%	**1.73%**	15%	11
5	60	3.78%	1.83%	11.68%	8
6	50	3.15%	1.81%	9.94%	7
7	50	2.98%	1.95%	7.75%	5
8	50	2.98%	1.95%	7.75%	5
9	50	**2.64%**	2%	**5.2%**	0

Tabel 4.12 Hasil dari Jarak Kedekatan Maksimal 6 Piksel dengan Ukuran *Window Filtering* 5

Persentase transitional pixels dekat yang normal *(TPn)*	***Threshold transition region* yang sedikit beririsan *(Ti)***	**ME**	**FPR**	**FNR**	**Jumlah citra yang difilter**
4	60	4.52%	**1.73%**	15%	11
5	60	4.55%	1.79%	14.49%	10
6	50	3.15%	1.81%	9.94%	7
7	50	**2.97%**	1.94%	7.78%	6
8	50	2.98%	1.95%	**7.75%**	5

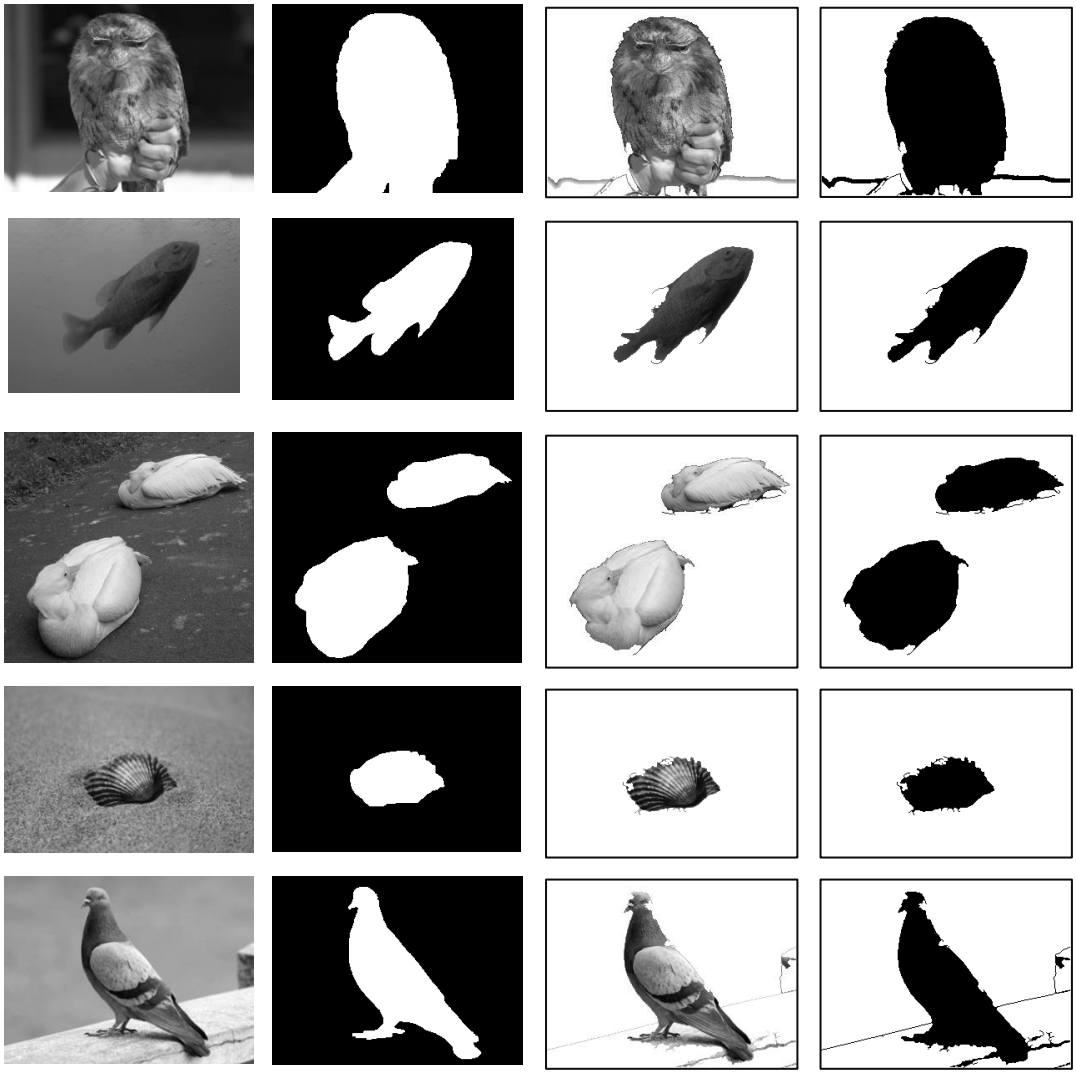
Tetapi, hasil FPR yang kecil ternyata berlawanan dengan FNR dan ME. Pada kejadian ini sistem menghasilkan kesalahan klasifikasi *foreground* FNR dan kesalahan klasifikasi secara keseluruhan ME yang buruk seperti yang terlihat pada Gambar 4.7 *mountain trees*. Hal tersebut terjadi karena proses penghapusan pada *transition region* dengan irisan yang kurang dari 60%. Sehingga, terlalu banyak *transition region* yang terhapus dan mengurangi *transition region* yang akan disambungkan.

Gambar 4.7 Beberapa Citra Hasil Segmentasi dengan Nilai Rata-Rata FPR Terkecil

Dari beberapa tabel di atas dapat dilihat bahwa nilai rata-rata FNR terkecil adalah 3.39% yang merupakan hasil dari beberapa kombinasi nilai *threshold* dengan parameter jarak *transitional pixels* dekat normal yang bernilai 1. FNR dengan nilai rata-rata 3.39% diperoleh dengan tidak melakukan *filtering* pada citra *input*. Seperti halnya nilai FPR, walaupun FNR bernilai kecil tetapi nilai FPR dan ME memiliki nilai yang tidak terlalu baik. Hal itu terjadi karena FNR hanya fokus terhadap *foreground* yang ditampilkan tanpa memperhatikan *background* yang ikut terklasifikasi sebagai *foreground*. Beberapa hasil segmentasi dari kombinasi *threshold* ini memang berhasil menampilkan sebagian besar area *foreground* tetapi banyak juga bagian *background* yang ikut terklasifikasi sebagai *foreground* seperti yang terlihat pada Gambar 4.8. Hal ini mengakibatkan sulitnya menentukan objek yang ada pada citra.

Gambar 4.8 Beberapa Citra Hasil Segmentasi dengan Nilai Rata-Rata FNR Terkecil

Selain nilai FPR dan FNR, pada tabel 4.2 sampai 4.12 juga menampilkan rata-rata nilai ME. Nilai ME dipilih sebagai dasar penentu nilai optimal untuk semua parameter atau *threshold* yang dibutuhkan sistem ini karena menunjukkan kesalahan klasifikasi secara keseluruhan. Nilai rata-rata ME 2.37% merupakan nilai rata-rata ME yang paling bagus. Terdapat empat kombinasi parameter yang menghasilkan rata-rata nilai tersebut seperti yang ditunjukkan pada Tabel 4.13.



(a) (b) (c) (d)

Gambar 4.9 Citra (a) *Input*, (b) *Ground Truth*, (c) Hasil Segmentasi dengan Kombinasi Parameter Paling Optimal, (d) *Transition Region*

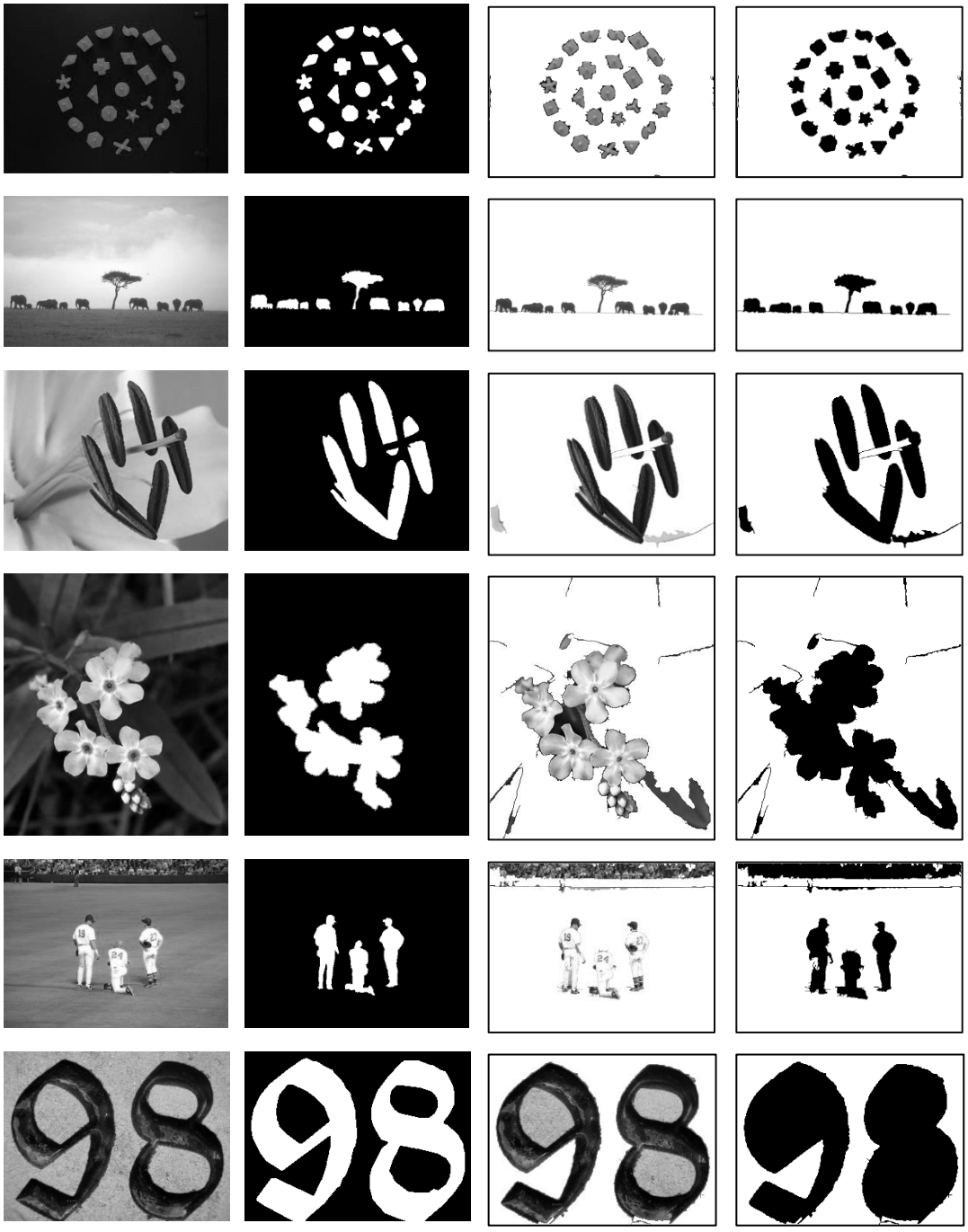
Tabel 4.13 menunjukkan bahwa ukuran *window* untuk *median filter* yang cocok pada sistem ini adalah 3. Hal ini menandakan bahwa ukuran tersebut terbukti mampu untuk menghaluskan citra sambil tetap menjaga bentuk *foreground.*

Proses segmentasi kemudian dilakukan dengan menggunakan kombinasi-kombinasi parameter tersebut terhadap 128 citra untuk mendapatkan kombinasi parameter yang paling optimal. Hasil dari pengujian ini dapat dilihat pada Tabel 4.13, Gambar 4.9, Gambar 4.10, dan Lampiran B. Tabel 4.13 menunjukkan bahwa kombinasi parameter kedua (jt=3, TPn=6%, Ti=50%, uw=3) merupakan

kombinasi paling optimal. Sehingga, dalam penelitian ini kombinasi parameter tersebut digunakan untuk menunjukkan hasil akhir dari sistem ini. Jarak maksimal kedekatan piksel yang bernilai 3 menunjukkan bahwa *transitional pixels* yang bangkit seharusnya memiliki jarak yang lebih dari 3 piksel karena *transitional pixels* yang bangkit dengan berjarak lebih kecil atau sama dengan 3 piksel dari tetangganya maka akan dianggap sebagai *transitional pixels* palsu. Nilai optimal persentase *transitional pixels* dekat yang normal adalah 6%. Nilai tersebut menunjukkan bahwa kemunculan *transitional pixels* yang saling berdekatan pada sebuah citra diharapkan sedikit. Selanjutnya *threshold* irisan *transition region* yang menunjukkan angka 50% menunjukkan bahwa *transition region* yang dianggap penting oleh metode ini harus memiliki irisan yang lebih besar dari setengah jumlah piksel anggotanya. Hasil pengujian ini juga menunjukkan bahwa ukuran *window* untuk *median filter* yang paling cocok adalah 3. Hal ini disebabkan *window* yang berukuran 3 mampu menghaluskan tekstur *background* sambil tetap menjaga tekstur dari *foreground*, sedangkan *window* yang berukuran 5 akan berdampak buruk pada proses segmentasi karena *window* tersebut akan lebih menghaluskan tekstur *foreground.*

Tabel 4.13 Kombinasi Parameter Optimal dan Rata-Rata ME

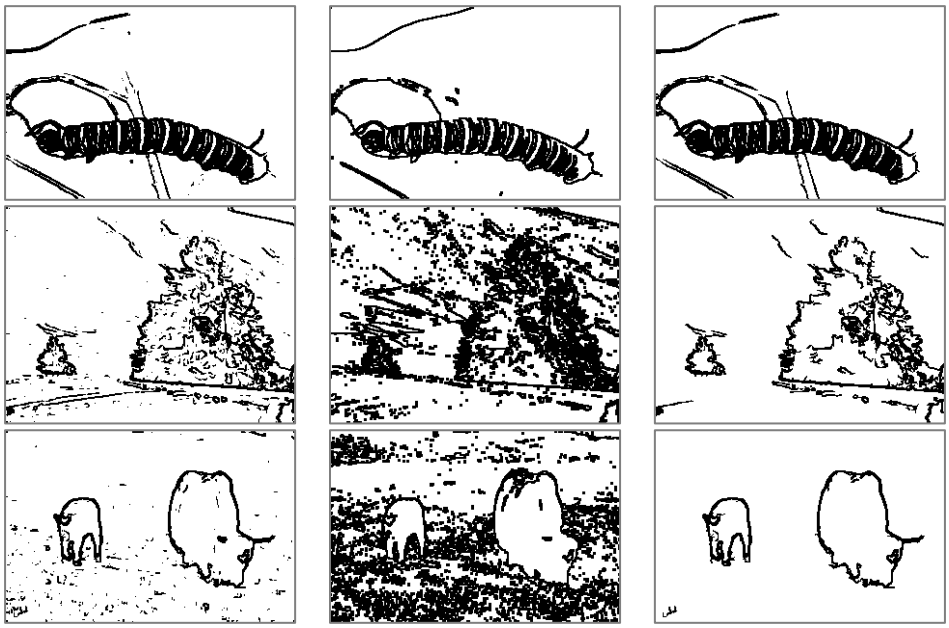
No.	**Jarak transitional pixels saling berdekatan *(jt)***	**Persentase transitional pixels dekat normal *(TPn)***	**Persentase untuk menentukan transition region citra asli yang sedikit beririsan *(Ti)***	**Ukuran window median filter *(uw)***	**Rata-rata nilai ME**
1	2	4%	50%	3	9.597370%
2	3	6%	50%	3	**9.290637%**
3	4	8%	50%	3	9.555582%
4	5	9%	50%	3	9.290657%



(a) (b) (c) (d)

Gambar 4.10 Citra (a) *Input*, (b) *Ground Truth*, (c) Hasil Segmentasi dengan Kombinasi Parameter Paling Optimal, (d) *Transition Region*

4.2.3. Hasil Pengujian dan Pembahasan Penghapusan *Transition Region* Citra Asli yang Sedikit Beririsan



(a) (b) (c)

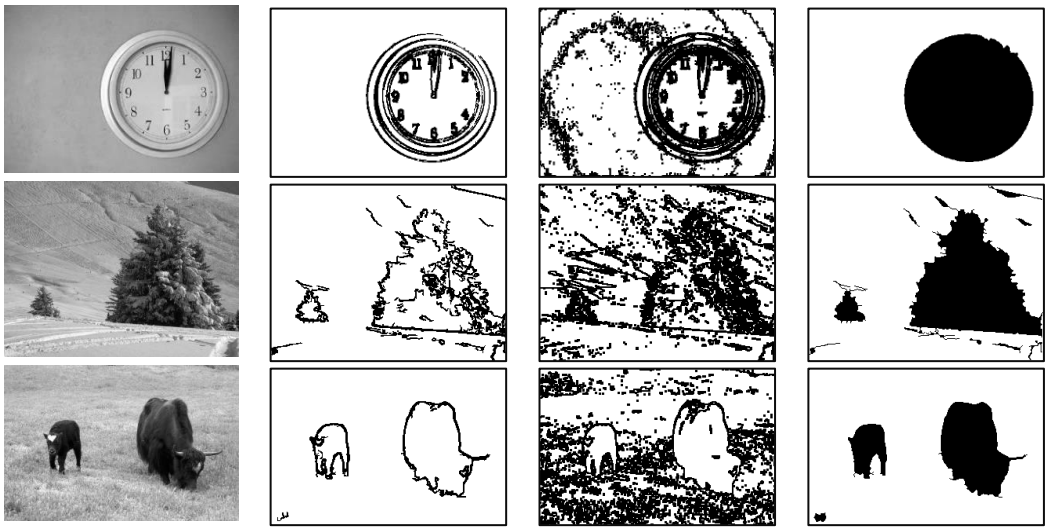
Gambar 4.11 (a) *Transition Region* Citra Asli, (b) *Transition Region* Citra Sederhana, (c) *Transition Region* Setelah Proses Penghapusan

Proses penghapusan *transition region* citra asli yang sedikit beririsan dengan *transition region* citra sederhana bertujuan untuk mengurangi *transition region* yang ada di area *background*. *Threshold* persentase untuk menentukan *transition region* citra asli yang sedikit beririsan menggunakan nilai 50% berdasarkan pada hasil dari pengujian sebelumnya. Gambar 4.11 menunjukkan bahwa proses ini terbukti mampu mengurangi *transition region* yang ada di area *background.*

Meskipun begitu, proses ini masih mengurangi *transition region* yang ada di area *foreground* pada beberapa citra yang mungkin memperburuk hasil segmentasi. Hal itu terjadi dikarenakan *transition region* citra sederhana sangat tergantung pada citra hasil *clustering*. Apabila citra hasil *clustering* dapat mengelompokkan piksel-piksel ke beberapa kelompok yang cocok dengan *background* dan *foreground*, maka *transition region* citra sederhana akan dihasilkan dengan bagus. Tetapi, apabila citra hasil *clustering* tidak dapat

mengelompokkan piksel dengan bagus, maka *transition region* citra sederhana mungkin dihasilkan dengan kurang bagus.

4.2.4. Hasil Pengujian dan Pembahasan Proses *Edge Linking*



(a) (b) (c) (d)

Gambar 4.12 (a) Citra Asli, *Transition Region* (b) Citra Asli, (c) Citra Sederhana, (d) Setelah Proses *Edge Linking* dan *Region Filling*

Proses *edge linking* pada penelitian ini hanya dilakukan pada *transition region* yang belum utuh. *Edge linking* tidak dilakukan pada *transition region* yang sudah utuh karena untuk menghindari adanya sambungan yang tidak perlu antara beberapa *foreground* atau *background* yang disebabkan oleh terdapatnya rute sambungan dari tepi *background* ke *foreground.* Sehingga *transition region* yang sudah utuh akan disimpan sementara dan akan digunakan lagi untuk digabungkan dengan *transition region* hasil *edge linking* setelah proses *edge linking* selesai dijalankan pada semua ujung *transition region.*

Proses *edge linking* yang dilakukan pada *transition region* belum utuh terbukti mampu menghubungkan ujung-ujung *transition region* seperti yang terlihat pada Gambar 4.12. Sistem ini mampu menghubungkan ujung *transition region* yang terpisah dekat maupun jauh selama terdapat rute penyambung dari *transition region* citra sederhana. Rute penyambung dari *transition region* citra sederhana merupakan salah satu komponen penting dalam proses ini, karena dengan itulah sistem akan mampu untuk menentukan apakah dua buah ujung

transition region bisa dihubungkan atau tidak. Seperti yang terlihat pada Gambar 4.12 *mountain trees* yang menunjukkan bahwa terdapat beberapa *transition region* yang masih terpisah, hal itu terjadi karena *transition region* tersebut tidak memiliki rute ke *transition region* lain. Meskipun begitu, sistem ini juga mungkin akan menghasilkan *transition region* yang kurang bagus apabila rute penyambung atau *transition region* citra sederhana terekstraksi dengan kurang bagus (*background* dan *foreground* menyatu) seperti Gambar 4.12 *mountain trees*.

Proses *edge linking* pada penelitian ini sangatlah tergantung pada jumlah *transition region* yang butuh untuk disambungkan dan tingkat kekompleksan rute sambungan dari *transition region* citra sederhana. Semakin sedikit *transition region* citra asli yang perlu disambungkan atau semakin sederhana *transition region* citra sederhana akan mempercepat proses segmentasi. Sebaliknya, semakin banyak ujung *transition region* citra asli yang perlu disambungkan atau semakin kompleks *transition region* citra sederhana akan mengakibatkan semakin lama proses segmentasi oleh sistem ini.

4.2.5. Pembahasan Hasil dari Sistem Secara Umum

Dari hasil-hasil pengujian menunjukkan bahwa sistem yang diusulkan terbukti mampu untuk melakukan segmentasi pada citra multi-objek. Untuk mengukur kinerja dari sistem ini, hasil dari sistem ini juga dibandingkan dengan hasil dari penelitian Parida & Bhoi (2016) seperti yang ditunjukkan pada Tabel 4.14. Selain itu, sistem ini juga dibandingkan dengan penelitian Parida & Bhoi (2016) secara visual seperti yang terlihat pada Gambar 4.14 dan 4.15.

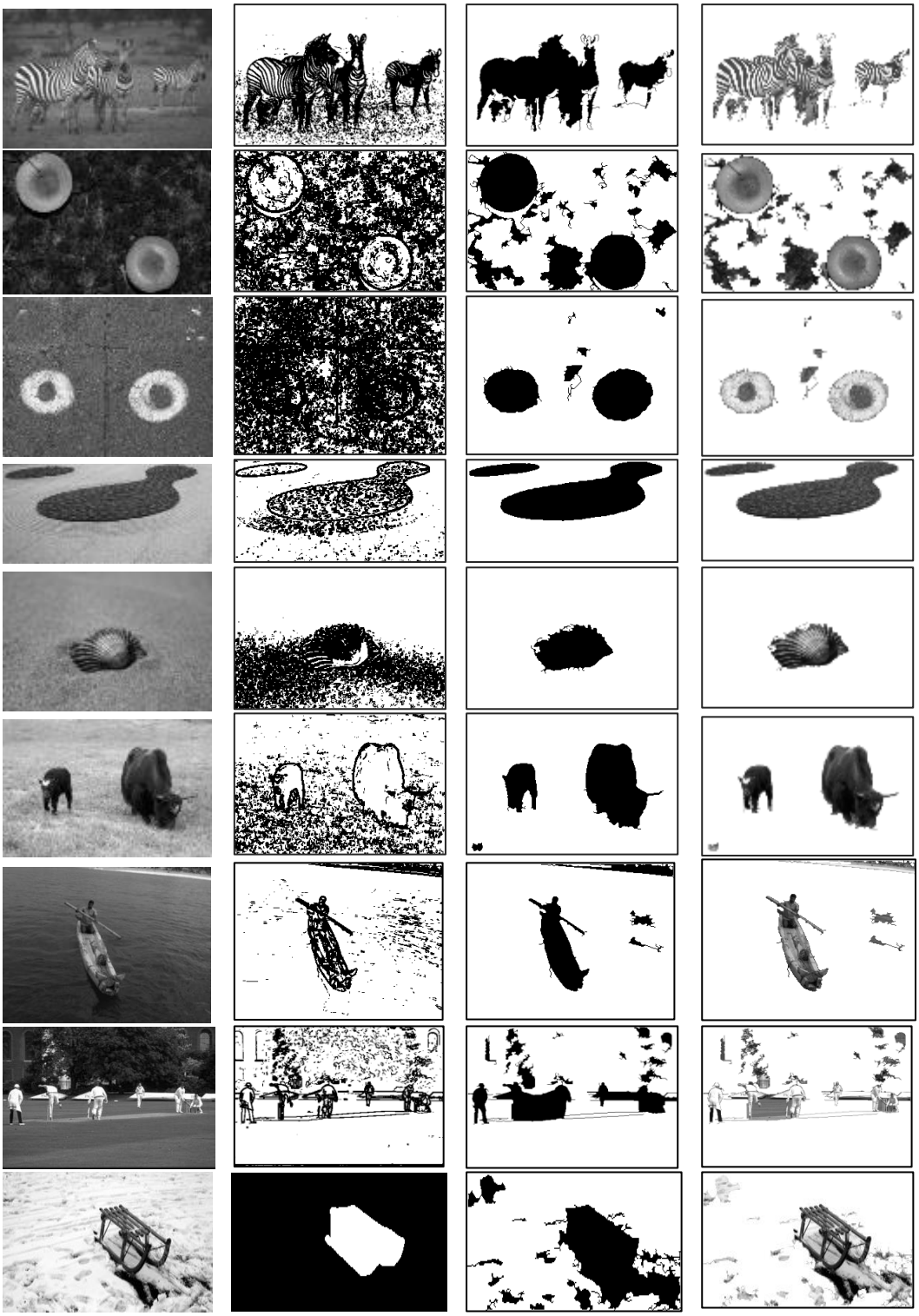
Tabel 4.14 menunjukkan bahwa sistem yang diusulkan mempunyai kinerja yang lebih baik daripada sistem dari penelitian Parida & Bhoi (2016) dalam segi kesalahan klasifikasi secara umum ME, *background* FPR, maupun *foreground* FNR. Sistem yang diusulkan memiliki rata-rata nilai ME yang lebih baik yaitu 2.45%, sedangkan nilai ME yang dihasilkan oleh penelitian Parida & Bhoi (2016) adalah 3.97%. Sistem ini mampu menghasilkan nilai ME lebih bagus karena sistem ini mampu mengurangi kemunculan *transition region* di area *background* dengan tetap memperhatikan *transition region* pada *foreground.*

Selain itu, sistem ini juga handal dalam melakukan segmentasi pada citra dengan *background* kompleks seperti yang terlihat pada Gambar 4.13.

Tabel 4.14 Data Perbandingan Hasil dari Sistem yang Diusulkan dengan Hasil dari Penelitian Parida & Bhoi, 2016

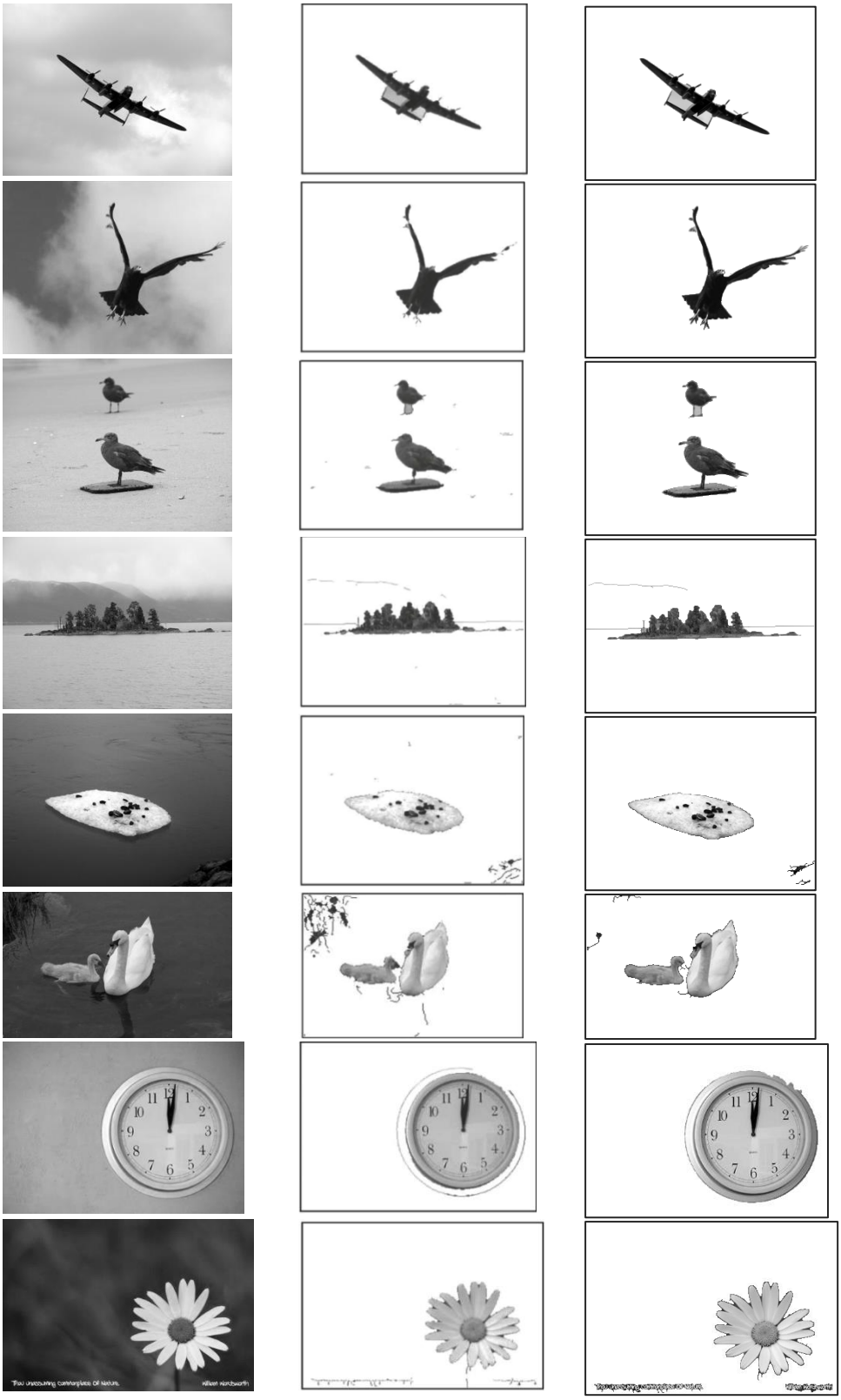
No	Citra	ME		FPR		FNR	
		Parida & Bhoi, 2016	**Proposed**	**Parida & Bhoi, 2016**	**Proposed**	**Parida & Bhoi, 2016**	**Proposed**
1	Airplane	1.79%	**1.2%**	**0.89%**	0.96%	16.7%	**5.24%**
2	Eagle	**0.65%**	1%	**0.47%**	1.05%	3.78%	**0.02%**
3	Bird	2.74%	**2.38%**	**1.92%**	1.98%	16.84%	**9.25%**
4	Island	1.06%	**0.93%**	**0.68%**	**0.68%**	11.68%	**4.16%**
5	Iceberg	0.91%	**0.49%**	0.36%	**0.32%**	6%	**2.09%**
6	Duck	3.27%	**1.34%**	2.39%	**0.58%**	9.57%	**6.82%**
7	Clock	6.15%	**0.51%**	**0.25%**	0.56%	18.09%	**0.41%**
8	Flower	**0.78%**	2.6%	**0.72%**	2.99%	1.18%	**0%**
9	Wall décor	6.28%	**4.03%**	**2.08%**	2.55%	13.89%	**6.71%**
10	Caterpillar	10.52%	**10.14%**	**9.74%**	11.03%	9.48%	**4.38%**
11	Boat	6.22%	**1.24%**	**0.17%**	0.3%	20.23%	**3.42%**
12	Mountain-tree	7.26%	**3.53%**	7.21%	**2.83%**	7.41%	**5.56%**
Rata-rata		3.97%	**2.45%**	2.24%	**2.15%**	11.24%	**4.01%**

Sistem ini juga menghasilkan rata-rata nilai FPR yang lebih bagus daripada penelitian Parida & Bhoi (2016) yaitu 2.15% dibandingkan 2.24%. Hal ini menunjukkan bahwa sistem berhasil mengurangi kemunculan *background* sebagai *foreground* sehingga meminimalisir kesalahan klasifikasi *background* menjadi 2.15%. Sedikitnya kesalahan klasifikasi tersebut diperoleh karena dalam proses penghapusan *transition region*, *transition region* yang dihapus bukan hanya yang berukuran kecil, tetapi juga yang sedikit beririsan dengan *transition region* citra sederhana. Sehingga, *transition region* yang berukuran kecil maupun besar juga akan dihapus apabila jumlah irisannya sedikit.



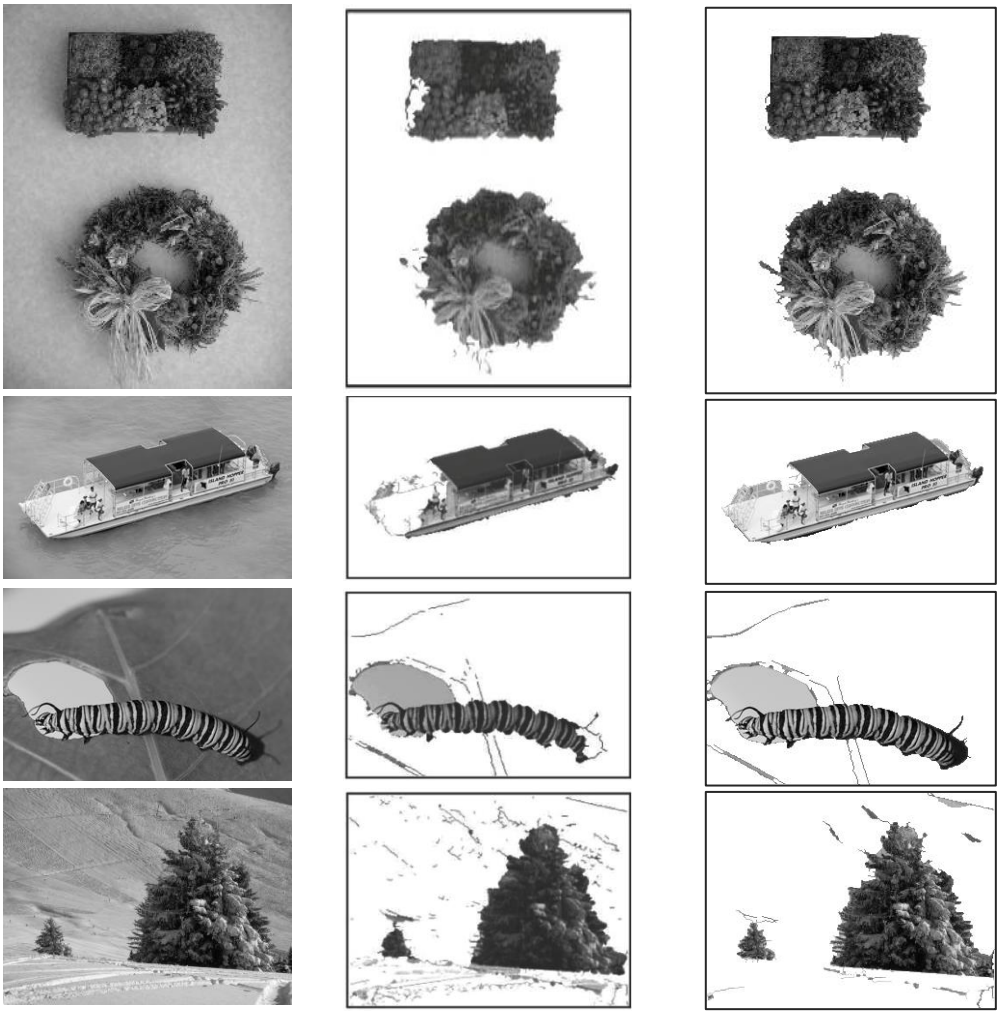
(a) (b) (c) (d)

Gambar 4.13 (a) Citra *Input* dengan *Background* Bertekstur, Hasil (b) *Transition Region* Standar, (c) *Region Filling* oleh Sistem, (d) Segmentasi.



(a) (b) (c)

Gambar 4.14 (a) Citra Asli, Hasil Segmentasi oleh: (b) Penelitian Parida & Bhoi, 2016, (c) Sistem yang Diusulkan



(a) (b) (c)

Gambar 4.15 (a) Citra Asli, Hasil Segmentasi oleh: (b) Penelitian Parida & Bhoi, 2016, (c) Sistem yang Diusulkan

Selain nilai ME dan FPR, rata-rata nilai FNR dari sistem ini juga lebih baik daripada nilai FNR dari penelitian Parida & Bhoi (2016) yaitu 4.01% dibandingkan 11.24%. Sistem yang diusulkan berhasil meminimalisir kesalahan klasifikasi *foreground* sebagai *background* menjadi hanya 4.01% karena *edge linking* dalam sistem ini dilakukan berdasarkan kesamaan visual yang diperoleh dari *transition region* citra sederhana. Sehingga kesalahan penyambungan antar ujung *transition region* dapat diminimalisir karena hanya ujung *transition region* yang memiliki rute sambungan dengannya saja yang akan disambungkan. Selain itu, batasan jarak antar ujung *transition region* yang akan disambung bisa menjadi fleksibel sesuai ukuran *transition region* tersebut.

Meskipun sistem yang diusulkan mampu untuk melakukan penyambungan *transition region* dan segmentasi dengan cukup baik, sistem ini masih terkendala dalam hal waktu untuk melakukan segmentasi seperti yang terlihat pada Tabel 4.15. Tabel 4.15 menunjukkan bahwa waktu tercepat sistem ini melakukan segmentasi adalah 5.2 detik, sedangkan waktu terlamanya adalah 616.5 detik. Lama waktu segmentasi dipengaruhi oleh banyaknya ujung *transition region* yang perlu disambungkan dan tingkat kekompleksan *transition region* citra sederhana. Semakin sedikit *transition region* yang perlu disambungkan atau semakin sederhana *transition region* citra sederhana akan mempercepat proses segmentasi. Sebaliknya, semakin banyak ujung *transition region* yang perlu disambungkan atau semakin kompleks *transition region* citra sederhana akan mengakibatkan semakin lama proses segmentasi oleh sistem ini. Selain itu, proses *region filling* pada sistem ini terkadang kurang optimal apabila terdapat *background* di tengah *foreground* seperti citra *ninetyeight* pada Gambar 4.10 (a). Pada kasus seperti ini sistem akan langsung menganggap piksel-piksel *background* di tengah *foreground* sebagai anggota *foreground*, sehingga piksel-piksel tersebut juga akan ditampilkan sebagai *foreground* hasil segmentasi seperti yang terlihat pada Gambar 4.10 (c) *ninetyeight*. Hal tersebut terjadi karena proses *region filling* pada sistem ini langsung melakukan pengisian pada tepi terluar objek tanpa melakukan pengecekan apakah di dalamnya terdapat area *background* ataukah tidak.

Tabel 4.15 Nilai ME, FPR, FNR, dan Lama Segmentasi oleh Sistem yang Diusulkan

No	**Nama citra input**	**ME**	**FPR**	**FNR**	**Lama (detik)**	**filter**
1	155	1.27%	0.20%	15.09%	7.7968	no
2	0677845-r1-067-32_a	2.48%	0.04%	19.08%	6.7133	no
3	100_0109	11.27%	17.07%	1.48%	9.8664	no
4	100_0497	12.73%	23.94%	0.18%	25.9743	no
5	100_1219	4.48%	0.68%	59.40%	8.194	no
6	1.1E+11	2.21%	1.11%	11.20%	16.2459	no
7	1.12E+11	17.26%	20.71%	0.62%	125.7797	yes
8	1.13E+11	7.33%	8.40%	3.97%	33.8968	yes

No	Nama citra input	ME	FPR	FNR	Lama (detik)	filter
9	20060319_087	51.54%	59.50%	0.87%	8.362	no
10	By_the_North_Sea15	2.74%	0.42%	12.92%	8.0732	no
11	DSC_18501299	0.25%	0.11%	4.20%	11.8192	no
12	_3076180_cropped	1.58%	1.30%	4.90%	9.36	yes
13	_3076180_cropped2	0.74%	0.78%	0.33%	12.8352	no
14	_mg_5707	0.86%	0.92%	0.49%	9.5148	no
15	aaa	4.50%	2.96%	20.92%	19.7376	no
16	alanmort_p1010058	2.66%	1.45%	6.45%	20.0643	yes
17	animal_5_bg_020803	4.15%	2.74%	6.34%	8.6668	no
18	b10mechanics037	0.39%	0.28%	0.74%	11.1079	no
19	b17leon000	0.40%	0.03%	30.87%	9.8765	no
20	b17maartent719	0.87%	0.77%	3.78%	16.3511	no
21	b17paul1444	21.21%	18.65%	24.05%	24.903	yes
22	b19objects118	14.28%	25.78%	0.24%	33.3965	no
23	b1chesnuttame	20.18%	18.81%	21.49%	16.0357	no
24	b1snake001	27.30%	43.65%	1.21%	12.9008	no
25	b20vehicles_water065	2.32%	1.90%	11.90%	26.0449	yes
26	b2chopper008	0.79%	0.42%	8.34%	7.7651	no
27	b2pods001	23.69%	30.48%	16.63%	12.5245	no
28	b8vehicles_water015	1.43%	0.65%	19.88%	8.8834	no
29	b9vehicles_air015	0.57%	0.20%	11.56%	7.5692	no
30	b9vehicles_air016	0.29%	0.25%	2.42%	8.2982	no
31	b9vehicles_air018_cropped	2.73%	0.45%	10.52%	8.9516	no
32	bbmf_lancaster_july_06 airplane	1.20%	0.96%	5.24%	10.0397	no
33	bbmf_lancaster_july_06	1.20%	0.96%	5.24%	9.9026	no
34	beltaine_4_bg_050502 island	0.93%	0.68%	4.16%	7.4824	no
35	beltaine_4_bg_050502	0.93%	0.68%	4.16%	7.3185	no
36	bi_planes	0.21%	0.13%	5.36%	9.3003	no
37	boatscn2174	0.59%	0.24%	4.74%	9.4396	no
38	boy_float_lake	14.27%	14.63%	4.12%	10.6418	no
39	bream_in_basin	3.05%	0.05%	17.32%	9.5563	no
40	carriage	29.70%	40.20%	0.42%	9.6827	no
41	carrigafoyle_castle_ext	3.92%	2.98%	12.75%	7.1634	no
42	caterpiller	10.14%	11.03%	4.38%	7.0178	no
43	chaom38	3.49%	4.10%	0.97%	9.973	yes
44	chaom69	3.47%	3.78%	1.04%	52.3307	yes

No	Nama citra input	ME	FPR	FNR	Lama (detik)	filter
45	clock	0.51%	0.56%	0.41%	18.6402	no
46	crw_0384 boat	1.24%	0.30%	3.42%	7.2646	no
47	crw_0384	1.24%	0.30%	3.42%	7.3251	no
48	decorwall	4.03%	2.55%	6.71%	7.5196	no
49	dsc00103_h	0.45%	0.06%	53.47%	8.8369	no
50	dsc01239_d	5.48%	1.26%	15.15%	15.9434	no
51	dsc04575 iceberg	0.49%	0.32%	2.09%	9.146	no
52	dsc04575	0.49%	0.32%	2.09%	9.118	no
53	dsc_7742	7.16%	0.29%	65.72%	10.8773	yes
54	dscf0034_l	8.82%	9.93%	3.70%	13.0354	no
55	dscf3208a	4.29%	4.74%	1.34%	30.2673	no
56	dscn1205_cropped	0.64%	0.53%	0.94%	5.2397	yes
57	dscn1908	1.21%	0.71%	1.90%	10.68	no
58	dscn2154	30.93%	36.85%	10.53%	8.6331	no
59	dscn5449	1.41%	0.93%	6.41%	13.3304	no
60	dscn6805	5.48%	5.05%	6.21%	9.7337	no
61	dualwindows	43.16%	59.30%	0.28%	11.9482	no
62	duck	1.34%	0.58%	6.82%	8.0851	yes
63	eagle	1.00%	1.05%	0.02%	13.6961	no
64	eagle052607__20_	0.33%	0.26%	3.19%	9.5661	no
65	elke_p9270005	36.89%	5.40%	61.48%	11.4565	yes
66	flower	2.60%	2.99%	0.00%	14.4124	no
67	hot_air_balloons_05	15.31%	18.34%	5.78%	20.8938	no
68	hpim1242	3.28%	3.37%	2.01%	14.0806	no
69	hpim1928	4.60%	0.04%	36.44%	11.9236	yes
70	hpim5083_morguefile	6.52%	5.06%	10.41%	11.717	no
71	img_1516	7.45%	8.05%	2.31%	25.8478	no
72	img_2528	11.59%	13.79%	4.01%	13.6838	no
73	img_2577	1.26%	0.21%	11.55%	7.5516	yes
74	img_3083_modif	4.24%	4.31%	3.93%	21.2813	no
75	img_3733_modif	2.11%	1.89%	8.33%	8.3709	no
76	img_4214	16.66%	7.35%	25.08%	11.6935	no
77	img_4730_modif	28.93%	31.72%	0.40%	179.8699	no
78	istambulminaret	4.26%	0.25%	15.65%	8.6093	no
79	kata_beach_phuket	2.91%	2.99%	1.59%	8.4496	no
80	matsumt060102_dyj08	13.53%	16.23%	2.60%	31.9275	no
81	mexico3	3.40%	1.88%	7.39%	8.0467	no

No	Nama citra input	ME	FPR	FNR	Lama (detik)	filter
82	mf_3379	1.39%	1.16%	6.07%	77.9629	no
83	mont_st_michel	10.06%	13.18%	2.23%	9.1762	no
84	moon_and_a_tree	5.10%	1.22%	12.99%	8.4092	no
85	moth061304_0679	0.74%	0.63%	1.43%	8.2616	yes
86	mountain trees	3.53%	2.83%	5.56%	23.5474	yes
87	ninetyeight	15.57%	26.85%	1.47%	6.9469	yes
88	osaka060102_dyjsn071	8.53%	11.14%	2.63%	18.2828	no
89	pc250176	13.92%	14.43%	0.77%	10.5377	no
90	per__02_cropped	16.01%	17.18%	0.86%	61.1459	yes
91	photo-026	0.48%	0.50%	0.00%	7.63	no
92	photo_sanary_19-03-05_083	27.31%	9.20%	81.15%	36.2271	no
93	pic1077114394	3.33%	0.21%	21.42%	7.839	no
94	pic1080629574	1.35%	0.00%	8.66%	13.3109	no
95	pic108459418417	12.80%	1.43%	50.12%	70.8911	no
96	pic108891392790	11.28%	11.35%	11.12%	42.4436	no
97	pic109250805856	24.75%	30.80%	0.67%	57.4804	no
98	pic1092515922117	6.65%	7.13%	3.95%	26.8656	no
99	pict2272	1.87%	0.13%	7.26%	29.6753	yes
100	picture_458	22.29%	23.99%	0.00%	11.5535	no
101	roman-pot-31-01-07-02842	20.03%	16.66%	24.73%	30.3706	no
102	sg_01_img_1943_tratada	2.54%	3.19%	0.89%	14.9519	no
103	sh-med-bandages-082404-01	9.91%	1.30%	33.71%	7.087	no
104	skookumchuk_starfish1	54.14%	70.16%	0.32%	8.9099	no
105	snow2_004	15.98%	17.68%	3.08%	10.6042	no
106	swan_and_friend	14.70%	19.45%	3.10%	195.2561	no
107	tendrils	22.12%	24.71%	10.41%	26.2924	no
108	toplessmonument	33.77%	35.51%	18.78%	10.5656	no
109	windowcn_0078	9.02%	14.72%	1.27%	16.9759	no
110	yack1	1.51%	1.64%	0.82%	7.764	yes
111	132581	7.19%	8.48%	0.34%	15.651	no
112	135069	0.29%	0.24%	1.10%	31.932	no
113	184112	30.50%	37.77%	1.10%	35.1132	no
114	184313	9.53%	11.20%	0.29%	25.4989	no
115	198720	2.08%	2.29%	0.09%	11.7941	no
116	216041	5.53%	1.06%	39.47%	19.1259	yes
117	245051	15.16%	10.34%	24.42%	14.3557	no

No	Nama citra input	ME	FPR	FNR	Lama (detik)	filter
118	253027	6.79%	1.50%	19.05%	21.6461	yes
119	253036	0.58%	0.49%	2.19%	13.1803	no
120	28046	1.95%	0.92%	14.57%	18.4275	no
121	361084	25.81%	35.37%	6.88%	22.4696	no
122	368016	10.78%	11.21%	3.71%	17.6304	yes
123	37508	42.94%	46.63%	0.21%	16.8412	no
124	38092	33.38%	39.68%	17.32%	616.5414	yes
125	42044	1.93%	1.62%	2.47%	13.1315	no
126	5108	10.19%	10.60%	4.94%	16.4139	no
127	69224	3.23%	3.82%	0.01%	11.9426	yes
128	90072	7.89%	8.16%	5.60%	73.3802	no
Rata-rata		**9.29%**	**9.57%**	**9.80%**	**24.9851**	

BAB 5
PENUTUP

Pada bab ini dijelaskan tentang kesimpulan dan saran dari serangkaian pengujian pada bab sebelumnya.

5.1. Kesimpulan

Berdasarkan penelitian ini, kesimpulan yang dapat diambil yaitu:

a. Sistem yang diusulkan mampu untuk menghasilkan *transition region* yang lebih bersih, utuh, dan mengelilingi objek.
b. Penambahan metode untuk memperbaiki citra yang memiliki *background* kompleks atau bertekstur menggunakan *median filter* mampu menghasilkan *transition region* yang lebih bersih karena tekstur dari *background* menjadi lebih sederhana akibat penghalusan tingkat keabuan antar piksel yang berdekatan.
c. Proses *edge linking* dengan metode A* pada sistem yang diusulkan terbukti mampu menghasilkan *transition region* yang lebih utuh karena dalam sistem ini *edge linking* dilakukan berdasarkan kesamaan visual yang diperoleh dari *transition region* citra sederhana hasil *clustering* menggunakan k-means *clustering*.
d. Proses *region filling* pada sistem ini berjalan dengan baik karena berhasil mengisi area atau lubang dari *transition region*. Tetapi, proses ini terkadang kurang optimal apabila terdapat *background* di tengah *foreground* karena piksel-piksel *background* tersebut akan dianggap sebagai anggota *foreground.*
e. Hasil uji coba menunjukkan bahwa sistem yang diusulkan memiliki kinerja yang lebih baik daripada kinerja dari metode lain karena sukses menghasilkan nilai rata-rata *ME, FPR,* dan *FNR* yang lebih baik.
f. Lama waktu untuk proses segmentasi pada sistem yang diusulkan dipengaruhi oleh banyaknya ujung *transition region* yang perlu disambungkan dan tingkat kekompleksan *transition region* citra sederhana. Semakin sedikit *transition region* yang perlu disambungkan atau semakin sederhana *transition region* citra sederhana akan mempercepat proses segmentasi. Sebaliknya, semakin banyak ujung *transition region* yang perlu disambungkan atau semakin kompleks *transition region* citra sederhana akan mengakibatkan semakin lama proses segmentasi oleh sistem ini.

5.2. Saran

Saran yang dapat diberikan untuk pengembangan lebih lanjut penelitian ini adalah:

a. Perbaikan pada proses *edge linking* karena metode A* terkadang membutuhkan waktu yang lama untuk menyambungkan semua ujung *transition region.*
b. Penambahan proses untuk melakukan pengecekan piksel-piksel *background* yang terdapat di tengah *foreground* supaya piksel –piksel *background* tersebut tidak dianggap sebagai anggota *foreground.*

DAFTAR PUSTAKA

Dhanachandra, N., Manglem, K., & Chanu, Y. J. (2015). Image Segmentation Using K-means Clustering Algorithm and Subtractive Clustering Algorithm. *Procedia Computer Science*, *54*, 764–771. https://doi.org/10.1016/j.procs.2015.06.090

Duo-le, F., & Ming, Z. (2011). A New Fast Region Filling Algorithm Based on Cross Searching Method. *International Conference on Computer Science and Education*, *202*, 380–387.

Felzenszwalb, P. F., & Huttenlocher, D. P. (2004). Efficient Graph-Based Image Segmentation. *International Journal of Computer Vision*, *59*(2), 167–181. https://doi.org/10.1023/B:VISI.0000022288.19776.77

Jaroš, M., Strakoš, P., Karásek, T., Ríha, L., Jarošová, M., & Kozubek, T. (2016). Implementation of K-means segmentation algorithm on Intel Xeon Phi and GPU : Application in medical imaging. *Advances in Engineering Software*, *0*, 1–8. https://doi.org/10.1016/j.advengsoft.2016.05.008

Ji, X., Zhang, X., & Zhang, L. (2013). Sequential edge linking method for segmentation of remotely sensed imagery based on heuristic search. *International Conference on Geoinformatics*, (40730635), 1–5. https://doi.org/10.1109/Geoinformatics.2013.6626164

Kang, C. C., Wang, W. J., & Kang, C. H. (2012). Image segmentation with complicated background by using seeded region growing. *AEU - International Journal of Electronics and Communications*, *66*(9), 767–771. https://doi.org/10.1016/j.aeue.2012.01.011

Li, Z., & Liu, C. (2009). Gray level difference-based transition region extraction and thresholding. *Computers and Electrical Engineering*, *35*(5), 696–704. https://doi.org/10.1016/j.compeleceng.2009.02.001

Li, Z., Liu, G., Zhang, D., & Xu, Y. (2016). Robust single-object image segmentation based on salient transition region. *Pattern Recognition*, *52*, 317–331. https://doi.org/10.1016/j.patcog.2015.10.009

Li, Z., Zhang, D., Xu, Y., & Liu, C. (2011a). Modified local entropy-based transition region extraction and thresholding. *Applied Soft Computing Journal*, *11*(8), 5630–5638. https://doi.org/10.1016/j.asoc.2011.04.001

Li, Z., Zhang, D., Xu, Y., & Liu, C. (2011b). Modified local entropy-based transition region extraction and thresholding. *Applied Soft Computing Journal*, *11*(8), 5630–5638. https://doi.org/10.1016/j.asoc.2011.04.001

Nyma, A., Kang, M., Kwon, Y.-K., Kim, C.-H., & Kim, J.-M. (2012). A Hybrid Technique for Medical Image Segmentation. *Journal of Biomedicine and Biotechnology*, *2012*, 1–7. https://doi.org/10.1155/2012/830252

Parida, P., & Bhoi, N. (2016). Transition region based single and multiple object segmentation of gray scale images. *Engineering Science and Technology, an International Journal*, *19*(3), 1206–1215. https://doi.org/10.1016/j.jestch.2015.12.009

Prasetyo, E. (2011). *Pengolahan Citra Digital dan Aplikasinya menggunakan Matlab*. (F. S. Suyantoro, Ed.). Yogyakarta: CV. Andi Offset.

Pratt, W. K. (2001). *PROCESSING DIGITAL IMAGE PROCESSING* (Vol. 5).

Rosyadi, A. W., Danardono, R., Manek, S. S., & Arifin, A. Z. (2018). A Flexible Sub-Block In Region Based Image Retrieval Based On Transition Region. *Jurnal Ilmu Komputer Dan Informasi*, *11*(1), 42–51.

Salmbier, P., & Marqués, F. (1999). Region-Based representation of images and video: Segmentation tools for multimedia services, *9*(8), 1147–1167.

Shakeri, M., Dezfoulian, M. H., Khotanlou, H., Barati, A. H., & Masoumi, Y. (2017). Image contrast enhancement using fuzzy clustering with adaptive cluster parameter and sub-histogram equalization. *Digital Signal Processing: A Review Journal*, *62*, 224–237. https://doi.org/10.1016/j.dsp.2016.10.013

Strasters, K., & Gerbrands, J. (1991). Three-dimensional image segmentation using a split, merge and group approach. *Pattern Recognition Letters*, *12*(May), 307–325. Retrieved from http://www.sciencedirect.com/science/article/pii/016786559190414H

Suyanto. (2011). *Artificial Intelligence (Edisi Revisi)*. Bandung: INFORMATIKA.

Wang, S., Chung, F. lai, & Xiong, F. (2008). A novel image thresholding method based on Parzen window estimate. *Pattern Recognition*, *41*(1), 117–129. https://doi.org/10.1016/j.patcog.2007.03.029

Yan, C., Sang, N., & Zhang, T. (2003). Local entropy-based transition region extraction and thresholding. *Pattern Recognition Letters*, *24*(16), 2935–2941. https://doi.org/10.1016/S0167-8655(03)00154-5

Yousefi, S., Azmi, R., & Zahedi, M. (2012). Brain tissue segmentation in MR images based on a hybrid of MRF and social algorithms. *Medical Image Analysis*, *16*(4), 840–848. https://doi.org/10.1016/j.media.2012.01.001

Zhang, Y. J., & Gerbrands, J. J. (1991). Transition region determination based thresholding. *Pattern Recognition Letters*, *12*(1), 13–23. https://doi.org/10.1016/0167-8655(91)90023-F

Lampiran A. Hasil Uji Coba

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 1 piksel dan persentase *transitional pixels* dekat normal bernilai 3%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**Waktu (detik)**	**filter**
10	4.21%	4.45%	**3.39%**	14.02482	0
20	4.19%	4.43%	**3.39%**	14.29174	0
30	4.19%	4.42%	**3.39%**	13.95762	0
40	4.19%	4.42%	**3.39%**	14.14632	0
50	**4.07%**	4.28%	**3.39%**	14.08283	0
60	6.40%	3.96%	10.57%	13.46122	0
70	6.27%	3.81%	10.57%	14.08817	0
80	6.89%	3.44%	16.52%	15.98389	0
90	8.83%	**0.85%**	40.35%	17.50377	0

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 1 piksel dan persentase *transitional pixels* dekat normal bernilai 4%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	4.21%	4.45%	**3.39%**	10.08746	0
20	4.19%	4.43%	**3.39%**	9.140301	0
30	4.19%	4.42%	**3.39%**	9.142273	0
40	4.19%	4.42%	**3.39%**	9.424539	0
50	**4.07%**	4.28%	**3.39%**	9.101898	0
60	6.40%	3.96%	10.57%	8.977843	0
70	6.27%	3.81%	10.57%	9.023438	0
80	6.89%	3.44%	16.52%	10.45099	0
90	8.83%	**0.85%**	40.35%	11.53294	0

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 1 piksel dan persentase *transitional pixels* dekat normal bernilai 5%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	4.21%	4.45%	**3.39%**	9.59351	0
20	4.19%	4.43%	**3.39%**	12.95581	0

30	4.19%	4.42%	**3.39%**	13.60737	0
40	4.19%	4.42%	**3.39%**	13.78888	0
50	**4.07%**	4.28%	**3.39%**	13.82865	0
60	6.40%	3.96%	10.57%	13.50051	0
70	6.27%	3.81%	10.57%	12.40486	0
80	6.89%	3.44%	16.52%	13.57322	0
90	8.83%	**0.85%**	40.35%	13.67036	0

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 1 piksel dan persentase *transitional pixels* dekat normal bernilai 6%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	4.21%	4.45%	**3.39%**	10.57628	0
20	4.19%	4.43%	**3.39%**	10.48855	0
30	4.19%	4.42%	**3.39%**	10.28325	0
40	4.19%	4.42%	**3.39%**	12.54688	0
50	**4.07%**	4.28%	**3.39%**	12.83324	0
60	6.40%	3.96%	10.57%	12.21647	0
70	6.27%	3.81%	10.57%	12.36306	0
80	6.89%	3.44%	16.52%	13.94773	0
90	8.83%	**0.85%**	40.35%	14.99097	0

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 1 piksel dan persentase *transitional pixels* dekat normal bernilai 7%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	4.21%	4.45%	**3.39%**	14.02845	0
20	4.19%	4.43%	**3.39%**	13.76598	0
30	4.19%	4.42%	**3.39%**	14.11618	0
40	4.19%	4.42%	**3.39%**	13.9614	0
50	**4.07%**	4.28%	**3.39%**	14.16186	0
60	6.40%	3.96%	10.57%	13.80398	0
70	6.27%	3.81%	10.57%	13.49044	0
80	6.89%	3.44%	16.52%	16.25354	0
90	8.83%	**0.85%**	40.35%	18.0771	0

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling

berdekatan 1 piksel dan persentase *transitional pixels* dekat normal bernilai 8%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	4.21%	4.45%	**3.39%**	13.67282	0
20	4.19%	4.43%	**3.39%**	13.77878	0
30	4.19%	4.42%	**3.39%**	14.2562	0
40	4.19%	4.42%	**3.39%**	13.64414	0
50	**4.07%**	4.28%	**3.39%**	13.89657	0
60	6.40%	3.96%	10.57%	13.99908	0
70	6.27%	3.81%	10.57%	14.44717	0
80	6.89%	3.44%	16.52%	16.61475	0
90	8.83%	**0.85%**	40.35%	18.12202	0

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 2 piksel dan persentase *transitional pixels* dekat normal bernilai 3%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	3.14%	2.07%	**5.96%**	16.70538	4
20	3.12%	2.04%	**5.96%**	16.35411	4
30	3.12%	2.04%	**5.96%**	16.9328	4
40	3.13%	2.04%	5.99%	16.37943	4
50	**3.11%**	2.01%	5.99%	16.027	4
60	3.94%	1.92%	8.51%	15.00555	4
70	4.07%	1.86%	9.01%	14.7558	4
80	4.68%	1.77%	10.94%	14.75899	4
90	4.96%	**0.95%**	18.29%	18.05749	4

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 2 piksel dan persentase *transitional pixels* dekat normal bernilai 4%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.42%	2.16%	**3.76%**	19.2007	3
20	2.40%	2.13%	**3.76%**	19.95012	3
30	2.40%	2.13%	**3.76%**	17.1595	3
40	2.40%	2.13%	**3.76%**	17.04231	3
50	**2.37%**	2.11%	**3.76%**	17.50941	3

60	4.79%	1.90%	10.94%	17.13822	3
70	4.74%	1.85%	10.94%	16.93202	3
80	4.68%	1.77%	10.94%	16.87628	3
90	4.96%	**0.95%**	18.29%	19.92339	3

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 2 piksel dan persentase *transitional pixels* dekat normal bernilai 5%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.79%	2.76%	**3.47%**	13.46598	2
20	2.77%	2.74%	**3.47%**	13.5903	2
30	2.77%	2.73%	**3.47%**	13.79541	2
40	2.77%	2.73%	**3.47%**	14.82867	2
50	**2.65%**	2.59%	**3.47%**	13.89716	2
60	5.07%	2.37%	10.64%	13.83242	2
70	5.03%	2.33%	10.64%	14.12093	2
80	4.96%	2.25%	10.64%	14.35162	2
90	6.76%	**0.86%**	25.55%	14.43585	2

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 2 piksel dan persentase *transitional pixels* dekat normal bernilai 6%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.92%	2.92%	**3.43%**	13.4899	1
20	2.90%	2.89%	**3.43%**	13.54698	1
30	2.90%	2.89%	**3.43%**	13.36103	1
40	2.90%	2.89%	**3.43%**	14.26666	1
50	**2.79%**	2.74%	**3.43%**	13.94513	1
60	5.20%	2.53%	10.60%	13.743	1
70	5.08%	2.39%	10.60%	14.27866	1
80	5.01%	2.31%	10.60%	13.53222	1
90	7.61%	**0.82%**	32.72%	13.26688	1

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 2 piksel dan persentase *transitional pixels* dekat normal bernilai 7%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.92%	2.92%	**3.43%**	15.21201	1
20	2.90%	2.89%	**3.43%**	14.06957	1
30	2.90%	2.89%	**3.43%**	13.92701	1
40	2.90%	2.89%	**3.43%**	13.91293	1
50	**2.79%**	2.74%	**3.43%**	13.91423	1
60	5.20%	2.53%	10.60%	13.81454	1
70	5.08%	2.39%	10.60%	13.82523	1
80	5.01%	2.31%	10.60%	13.67542	1
90	7.61%	**0.82%**	32.72%	13.89049	1

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 2 piksel dan persentase *transitional pixels* dekat normal bernilai 8%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.92%	2.92%	**3.43%**	13.89858	1
20	2.90%	2.89%	**3.43%**	424.127	1
30	2.90%	2.89%	**3.43%**	13.30981	1
40	2.90%	2.89%	**3.43%**	13.7479	1
50	**2.79%**	2.74%	**3.43%**	15.26657	1
60	5.20%	2.53%	10.60%	17.35888	1
70	5.08%	2.39%	10.60%	14.91343	1
80	5.01%	2.31%	10.60%	14.26726	1
90	7.61%	**0.82%**	32.72%	13.88093	1

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 3 piksel dan persentase *transitional pixels* dekat normal bernilai 4%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	3.23%	1.93%	**7.33%**	16.18329	7
20	3.21%	1.91%	**7.33%**	16.20166	7
30	3.21%	1.91%	**7.33%**	16.23004	7
40	3.21%	1.90%	7.36%	16.06535	7

50	**3.19%**	1.88%	7.36%	16.05798	7
60	4.02%	1.78%	9.88%	15.06896	7
70	4.15%	1.72%	10.38%	15.1774	7
80	4.77%	1.64%	12.31%	14.24824	7
90	4.72%	**1.58%**	12.31%	15.90408	7

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 3 piksel dan persentase *transitional pixels* dekat normal bernilai 5%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.49%	2.13%	**4.42%**	14.10765	5
20	2.48%	2.10%	**4.42%**	14.10068	5
30	2.48%	2.10%	**4.42%**	14.09738	5
40	2.48%	2.10%	**4.42%**	14.10223	5
50	**2.45%**	2.08%	**4.42%**	14.11629	5
60	4.86%	1.86%	11.60%	14.095	5
70	4.82%	1.81%	11.60%	14.03411	5
80	4.76%	1.73%	11.60%	14.06595	5
90	5.04%	**0.92%**	18.95%	15.94428	5

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 3 piksel dan persentase *transitional pixels* dekat normal bernilai 6%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.42%	2.16%	**3.76%**	10.51909	3
20	2.40%	2.13%	**3.76%**	11.70727	3
30	2.40%	2.13%	**3.76%**	12.55705	3
40	2.40%	2.13%	**3.76%**	13.41659	3
50	**2.37%**	2.11%	**3.76%**	11.82588	3
60	4.79%	1.90%	10.94%	11.61562	3
70	4.74%	1.85%	10.94%	11.69318	3
80	4.68%	1.77%	10.94%	11.43401	3
90	4.96%	**0.95%**	18.29%	13.52381	3

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 3 piksel dan persentase *transitional pixels* dekat normal bernilai 7%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.79%	2.76%	**3.47%**	11.09941	2
20	2.77%	2.74%	**3.47%**	10.60732	2
30	2.77%	2.73%	**3.47%**	10.4842	2
40	2.77%	2.73%	**3.47%**	11.48749	2
50	**2.65%**	2.59%	**3.47%**	10.58767	2
60	5.07%	2.37%	10.64%	10.54965	2
70	5.03%	2.33%	10.64%	10.63199	2
80	4.96%	2.25%	10.64%	10.64723	2
90	6.76%	**0.86%**	25.55%	9.689031	2

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 3 piksel dan persentase *transitional pixels* dekat normal bernilai 8%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.79%	2.76%	**3.47%**	9.473891	2
20	2.77%	2.74%	**3.47%**	9.462725	2
30	2.77%	2.73%	**3.47%**	9.494648	2
40	2.77%	2.73%	**3.47%**	9.496103	2
50	**2.65%**	2.59%	**3.47%**	9.50991	2
60	5.07%	2.37%	10.64%	9.44454	2
70	5.03%	2.33%	10.64%	9.460081	2
80	4.96%	2.25%	10.64%	9.452241	2
90	6.76%	**0.86%**	25.55%	9.47505	2

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 3 piksel dan persentase *transitional pixels* dekat normal bernilai 9%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.79%	2.76%	**3.47%**	9.501247	2
20	2.77%	2.74%	**3.47%**	9.496448	2
30	2.77%	2.73%	**3.47%**	9.541643	2
40	2.77%	2.73%	**3.47%**	9.555509	2

50	**2.65%**	2.59%	**3.47%**	9.576489	2
60	5.07%	2.37%	10.64%	9.52043	2
70	5.03%	2.33%	10.64%	9.494086	2
80	4.96%	2.25%	10.64%	9.45621	2
90	6.76%	**0.86%**	25.55%	9.557929	2

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 4 piksel dan persentase *transitional pixels* dekat normal bernilai 4%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	3.87%	1.88%	**9.62%**	14.52765	9
20	3.85%	1.86%	**9.62%**	15.10524	9
30	**3.85%**	1.85%	**9.62%**	14.52334	9
40	3.85%	1.85%	9.65%	14.52789	9
50	3.85%	1.85%	9.65%	14.54849	9
60	4.69%	1.76%	12.17%	13.85583	9
70	4.82%	1.70%	12.67%	15.3268	9
80	5.44%	1.62%	14.60%	15.71579	9
90	5.40%	**1.56%**	14.60%	17.36236	9

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 4 piksel dan persentase *transitional pixels* dekat normal bernilai 5%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	3.19%	1.89%	**7.35%**	26.55867	8
20	3.18%	1.88%	**7.35%**	23.80555	8
30	**3.18%**	1.87%	**7.35%**	20.99865	8
40	3.18%	1.87%	7.37%	20.48555	8
50	3.18%	1.86%	7.37%	16.72568	8
60	4.02%	1.78%	9.89%	15.7982	8
70	4.15%	1.72%	10.39%	15.66027	8
80	4.77%	1.64%	12.32%	15.24509	8
90	4.73%	**1.58%**	12.32%	16.61057	8

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 4 piksel dan persentase *transitional pixels* dekat normal bernilai 6%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.49%	2.13%	**4.42%**	13.90801	5
20	2.48%	2.10%	**4.42%**	13.87102	5
30	2.48%	2.10%	**4.42%**	13.92334	5
40	2.48%	2.10%	**4.42%**	14.0568	5
50	**2.45%**	2.08%	**4.42%**	13.92277	5
60	4.86%	1.86%	11.60%	13.8727	5
70	4.82%	1.81%	11.60%	13.83948	5
80	4.76%	1.73%	11.60%	13.82089	5
90	5.04%	**0.92%**	18.95%	15.71649	5

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 4 piksel dan persentase *transitional pixels* dekat normal bernilai 7%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.44%	2.15%	**4.21%**	10.69758	4
20	2.42%	2.13%	**4.21%**	10.67249	4
30	2.42%	2.13%	**4.21%**	10.67389	4
40	2.42%	2.13%	**4.21%**	10.6567	4
50	**2.40%**	2.10%	**4.21%**	10.67177	4
60	4.81%	1.89%	11.38%	10.6679	4
70	4.77%	1.84%	11.38%	10.63677	4
80	4.70%	1.76%	11.38%	10.5967	4
90	4.99%	**0.94%**	18.74%	12.50011	4

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 4 piksel dan persentase *transitional pixels* dekat normal bernilai 8%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.42%	2.16%	**3.76%**	10.40788	3
20	2.40%	2.13%	**3.76%**	10.50574	3
30	2.40%	2.13%	**3.76%**	10.44322	3
40	2.40%	2.13%	**3.76%**	10.40397	3

50	**2.37%**	2.11%	**3.76%**	10.42853	3
60	4.79%	1.90%	10.94%	10.4306	3
70	4.74%	1.85%	10.94%	10.32022	3
80	4.68%	1.77%	10.94%	10.32326	3
90	4.96%	**0.95%**	18.29%	12.22625	3

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 4 piksel dan persentase *transitional pixels* dekat normal bernilai 9%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.79%	2.76%	**3.47%**	9.334255	2
20	2.77%	2.74%	**3.47%**	9.331657	2
30	2.77%	2.73%	**3.47%**	9.34123	2
40	2.77%	2.73%	**3.47%**	9.335232	2
50	**2.65%**	2.59%	**3.47%**	9.344374	2
60	5.07%	2.37%	10.64%	9.362524	2
70	5.03%	2.33%	10.64%	9.303983	2
80	4.96%	2.25%	10.64%	9.281978	2
90	6.76%	**0.86%**	25.55%	9.343833	2

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 5 piksel dan persentase *transitional pixels* dekat normal bernilai 4%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	3.82%	1.82%	**9.76%**	15.56981	11
20	3.80%	1.80%	**9.76%**	14.82286	11
30	**3.80%**	1.80%	**9.76%**	14.67033	11
40	3.81%	1.79%	9.79%	16.38508	11
50	3.81%	1.79%	9.79%	14.68016	11
60	4.64%	1.70%	12.30%	14.00967	11
70	4.77%	1.64%	12.80%	15.07502	11
80	5.40%	1.57%	14.73%	15.85956	11
90	5.35%	**1.51%**	14.73%	16.2448	11

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 5 piksel dan persentase *transitional pixels* dekat normal bernilai 5%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	3.19%	1.89%	**7.35%**	14.83658	8
20	3.18%	1.88%	**7.35%**	14.97011	8
30	**3.18%**	1.87%	**7.35%**	15.13704	8
40	3.18%	1.87%	7.37%	14.78224	8
50	3.18%	1.86%	7.37%	14.8407	8
60	4.02%	1.78%	9.89%	14.1248	8
70	4.15%	1.72%	10.39%	14.54204	8
80	4.77%	1.64%	12.32%	14.49257	8
90	4.73%	**1.58%**	12.32%	16.03375	8

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 5 piksel dan persentase *transitional pixels* dekat normal bernilai 6%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.47%	1.99%	**5.14%**	14.46242	7
20	2.45%	1.97%	**5.14%**	14.20884	7
30	2.45%	1.96%	**5.14%**	14.16638	7
40	2.45%	1.96%	**5.14%**	14.04005	7
50	**2.45%**	1.96%	**5.14%**	14.03572	7
60	4.87%	1.76%	12.32%	14.07769	7
70	4.83%	1.70%	12.32%	14.17311	7
80	4.77%	1.64%	12.32%	14.11454	7
90	4.73%	**1.58%**	12.32%	16.03527	7

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 5 piksel dan persentase *transitional pixels* dekat normal bernilai 7%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.49%	2.13%	**4.42%**	14.16145	5
20	2.48%	2.10%	**4.42%**	18.68567	5
30	2.48%	2.10%	**4.42%**	14.56564	5
40	2.48%	2.10%	**4.42%**	14.09585	5

50	**2.45%**	2.08%	**4.42%**	14.10854	5
60	4.86%	1.86%	11.60%	14.14088	5
70	4.82%	1.81%	11.60%	14.74902	5
80	4.76%	1.73%	11.60%	14.27814	5
90	5.04%	**0.92%**	18.95%	16.1606	5

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 5 piksel dan persentase *transitional pixels* dekat normal bernilai 8%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.49%	2.13%	**4.42%**	14.18788	5
20	2.48%	2.10%	**4.42%**	14.1548	5
30	2.48%	2.10%	**4.42%**	14.18374	5
40	2.48%	2.10%	**4.42%**	14.10743	5
50	**2.45%**	2.08%	**4.42%**	14.11157	5
60	4.86%	1.86%	11.60%	14.09393	5
70	4.82%	1.81%	11.60%	14.05636	5
80	4.76%	1.73%	11.60%	14.20117	5
90	5.04%	**0.92%**	18.95%	16.01807	5

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 5 piksel dan persentase *transitional pixels* dekat normal bernilai 9%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.42%	2.16%	**3.76%**	15.27737	0
20	2.40%	2.13%	**3.76%**	15.5176	0
30	2.40%	2.13%	**3.76%**	15.58295	0
40	2.40%	2.13%	**3.76%**	14.91307	0
50	**2.37%**	2.11%	**3.76%**	15.53564	0
60	4.79%	1.90%	10.94%	14.88901	0
70	4.74%	1.85%	10.94%	14.74167	0
80	4.68%	1.77%	10.94%	16.09335	0
90	4.96%	**0.95%**	18.29%	21.57449	0

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 6 piksel dan persentase *transitional pixels* dekat normal bernilai 4%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	3.82%	1.82%	**9.76%**	15.12002	11
20	3.80%	1.80%	**9.76%**	14.99791	11
30	**3.80%**	1.80%	**9.76%**	14.79324	11
40	3.81%	1.79%	9.79%	14.96318	11
50	3.81%	1.79%	9.79%	15.21411	11
60	4.64%	1.70%	12.30%	14.46923	11
70	4.77%	1.64%	12.80%	14.31741	11
80	5.40%	1.57%	14.73%	13.68901	11
90	5.35%	**1.51%**	14.73%	19.03576	11

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 6 piksel dan persentase *transitional pixels* dekat normal bernilai 5%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	3.85%	1.86%	**9.65%**	14.684	10
20	3.84%	1.84%	**9.65%**	14.70625	10
30	**3.84%**	1.84%	**9.65%**	14.67411	10
40	3.84%	1.83%	9.67%	15.06793	10
50	3.84%	1.83%	9.67%	16.72428	10
60	4.68%	1.74%	12.19%	18.66368	10
70	4.81%	1.69%	12.69%	13.82247	10
80	5.43%	1.61%	14.62%	13.73758	10
90	5.39%	**1.55%**	14.62%	15.60904	10

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 6 piksel dan persentase *transitional pixels* dekat normal bernilai 6%.

***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.47%	1.99%	**5.14%**	13.89841	7
20	2.45%	1.97%	**5.14%**	13.89651	7
30	2.45%	1.96%	**5.14%**	13.88005	7
40	2.45%	1.96%	**5.14%**	17.84393	7

50	**2.45%**	1.96%	**5.14%**	17.50438	7
60	4.87%	1.76%	12.32%	17.66611	7
70	4.83%	1.70%	12.32%	13.74752	7
80	4.77%	1.64%	12.32%	13.75196	7
90	4.73%	**1.58%**	12.32%	15.6116	7

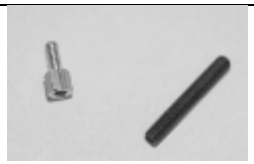
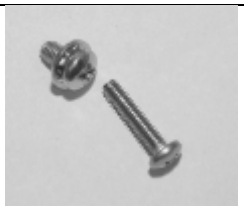
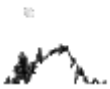
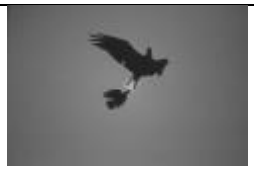
Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 6 piksel dan persentase *transitional pixels* dekat normal bernilai 7%.

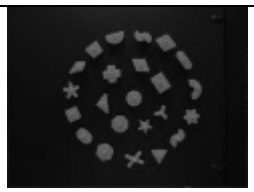
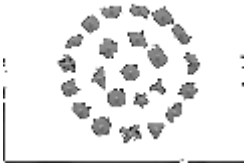
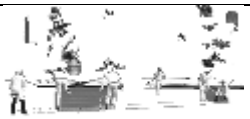
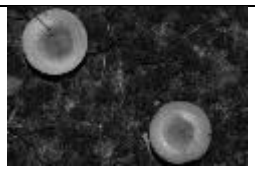
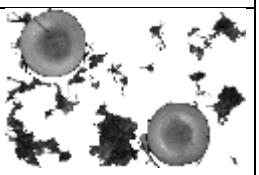
***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.46%	2.09%	**4.43%**	17.71234	6
20	2.45%	2.07%	**4.43%**	17.43279	6
30	2.45%	2.07%	**4.43%**	17.44117	6
40	2.45%	2.07%	**4.43%**	17.24633	6
50	**2.44%**	2.06%	**4.43%**	17.21107	6
60	4.86%	1.86%	11.61%	16.21048	6
70	4.82%	1.81%	11.61%	16.25999	6
80	4.76%	1.73%	11.61%	15.24184	6
90	5.04%	**0.92%**	18.97%	17.221	6

Tabel Hasil segmentasi dengan parameter maksimal jarak *transitional pixels* saling berdekatan 6 piksel dan persentase *transitional pixels* dekat normal bernilai 8%.

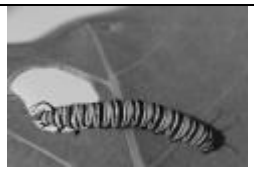
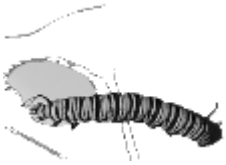
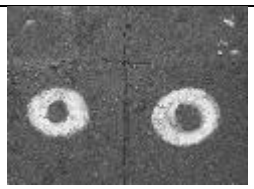
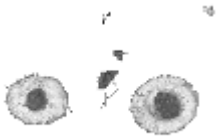
***Threshold transition region* yang sedikit beririsan**	**Rata-rata ME**	**Rata-rata FPR**	**Rata-rata FNR**	**waktu (detik)**	**filter**
10	2.49%	2.13%	**4.42%**	15.11294	5
20	2.48%	2.10%	**4.42%**	15.0613	5
30	2.48%	2.10%	**4.42%**	15.12766	5
40	2.48%	2.10%	**4.42%**	15.23792	5
50	**2.45%**	2.08%	**4.42%**	15.3164	5
60	4.86%	1.86%	11.60%	15.1969	5
70	4.82%	1.81%	11.60%	15.1416	5
80	4.76%	1.73%	11.60%	15.18385	5
90	5.04%	**0.92%**	18.95%	17.16366	5

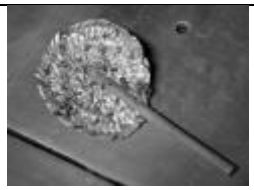
Lampiran B. Citra Hasil Segmentasi

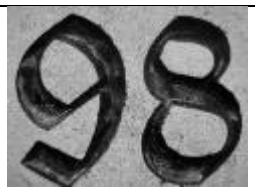
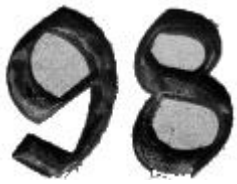
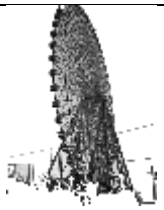
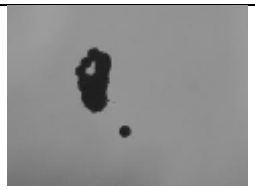
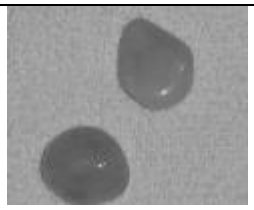
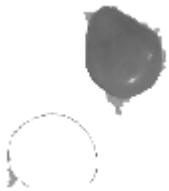
Citra asli	Hasil segmentasi	Citra asli	Hasil segmentasi
			
			
			

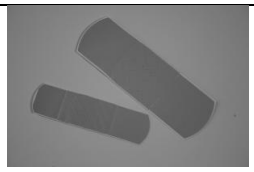
Citra asli	Hasil segmentasi	Citra asli	Hasil segmentasi
			
			
			

Citra asli	Hasil segmentasi	Citra asli	Hasil segmentasi

Citra asli	Hasil segmentasi	Citra asli	Hasil segmentasi
			
			

Citra asli	Hasil segmentasi	Citra asli	Hasil segmentasi
			

Citra asli	Hasil segmentasi	Citra asli	Hasil segmentasi
			
			
			

Citra asli	Hasil segmentasi	Citra asli	Hasil segmentasi
			

Biografi Penulis

Penulis, Ahmad Wahyu Rosyadi, lahir di Kota Gresik pada tanggal 05 Agustus 1992. Penulis adalah anak pertama dari tiga bersaudara. Penulis menghabiskan masa kanak-kanak hingga remaja di kota Gresik.

Penulis menempuh pendidikan formal di MI Assa'adah Sukowati, Gresik (1998-2004), MTs Ma'arif NU Assa'adah 1 Sampurnan Bungah, Gresik (2004-2007), dan SMA Ma'arif NU Assa'adah Sampurnan Bungah, Gresik (2007-2010). Pada tahun 2010-2014, penulis melanjutkan pendidikan S1 di Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang, Jawa Timur. Pada tahun 2016-2018, penulis melanjutkan pendidikan Magister S2 pada Departemen Informatika, Fakultas Teknologi Informasi Dan Komunikasi, Institut Teknologi Sepuluh Nopember Surabaya, Jawa Timur.

Penulis mengambil bidang minat Komputasi Cerdas Dan Visi pada jurusan Teknik Informatika. Selama dalam pendidikan S1, penulis pernah menjadi asisten laboratorium yang mengajar kelas praktikum Dasar-Dasar Pemrograman, Grafika Komputer dan Sistem Terdistribusi. Penulis dapat dihubungi melalui alamat email wahyu.rosyad@gmail.com.