

UNDERGRADUATE THESIS

**EDGE DETECTION IN COLORED IMAGES WITH
BACTERIAL FORAGING ALGORITHM**



Sutyoso

NPM: 2015730045

**DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES
PARAHYANGAN CATHOLIC UNIVERSITY
2019**

LEMBAR PENGESAHAN

DETEKSI TEPI PADA CITRA BERWARNA DENGAN BACTERIAL FORAGING ALGORITHM

Sutyoso

NPM: 2015730045

Bandung, 22 Mei 2019

Menyetujui,

Pembimbing

Kristopher David Harjono, M.T.

Ketua Tim Penguji

Anggota Tim Penguji

Vania Natali, M.T.

Natalia, M.Si.

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng

PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

DETEKSI TEPI PADA CITRA BERWARNA DENGAN BACTERIAL FORAGING ALGORITHM

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,
Tanggal 22 Mei 2019

Meterai Rp. 6000

Sutyoso
NPM: 2015730045

ABSTRAK

Dengan semakin meningkatnya citra digital yang beredar di masyarakat, kebutuhan pengolahan citra digital juga ikut meningkat. Salah satu pengolahan tersebut ialah segmentasi citra, yaitu membagi suatu citra ke dalam beberapa segmen dengan suatu kriteria tertentu. Jenis operasi ini berkaitan erat dengan pengenalan pola. Pola suatu objek pada citra dapat dikenali jika tepi objek tersebut telah diketahui. Untuk mengenali tepi suatu objek pada citra, terdapat beberapa teknik tradisional seperti teknik Sobel, Canny, Roberts, Kirsch, dan Morphological Gradient. Teknik-teknik tradisional tersebut cenderung tidak mengatasi masalah *vagueness*. Selain dengan menggunakan teknik tradisional tersebut, deteksi tepi juga dapat dilakukan dengan menggunakan teknik yang lain. Dengan menggunakan *Bacterial Foraging Optimization Algorithm* (BFOA) adalah salah satu teknik untuk melakukan proses deteksi tepi. Teknik ini akan diimplementasikan pada penelitian yang akan dilakukan.

Proses deteksi tepi dengan menggunakan BFOA memerlukan metode USAN dan *thresholding*. Cara kerja deteksi tepi dengan menggunakan BFOA terlebih dahulu menghitung USAN area dari citra masukkan yang selanjutnya akan dilakukan optimasi dengan BFOA. Setelah dioptimasi, akan dilakukan *thresholding* untuk mendapatkan citra biner sebagai citra hasil. Pada implementasi yang dilakukan terdapat metode *thresholding* mean, metode *thresholding* median, metode *thresholding* mean & median (non-variance), metode *thresholding* mean & median (variance), metode *thresholding* mean & median (variance) & otsu, dan metode *thresholding* Otsu.

Pada perangkat lunak yang dibangun, pengujian kualitas deteksi tepi akan menggunakan basis data gambar Berkeley. Pengujian perangkat lunak dilakukan dengan menggunakan metode *Mean Square Error* (MSE) dan koefisien Jaccard. Dari penelitian yang telah dilakukan, perangkat lunak yang dibangun telah mampu untuk mendeteksi tepi objek pada citra. Hasil dari pengujian perangkat lunak ialah metode *thresholding* yang digunakan akan mempengaruhi citra keluaran. Begitu juga dengan nilai parameter pada algoritma BFOA. Parameter BFOA selain mempengaruhi citra hasil, juga mempengaruhi waktu eksekusi yang dibutuhkan.

Kata-kata kunci: USAN, *Bacterial Foraging Optimization Algorithm*, Fuzzy, *Fuzzy Intensifier*, *Edge quality and sharpness*, *Mean Square Error* (MSE), koefisien Jaccard.

ABSTRACT

With the increasing number of digital images circulating in the community, digital image processing needs have also increased. One such processing is image segmentation, which is dividing an image into several segments with certain criteria. This type of operation is closely related to pattern recognition. The pattern of an object in an image can be identified if the object's edge is known. To recognize the edges of an object in an image, there are several traditional techniques such as the Sobel, Canny, Roberts, Kirsch, and Morphological Gradient techniques. These traditional techniques tend not to overcome vagueness problems. In addition to using these traditional techniques, edge detection can also be done using other techniques. By using Bacterial Foraging Optimization Algorithm (BFOA) is one technique for conducting edge detection processes. This technique will be implemented in the research that will be carried out.

The process of edge detection using BFOA requires the USAN method and thresholding. The workings of edge detection using BFOA first calculate the USAN area from the input image which will then be optimized with BFOA. After optimization, thresholding will be done to get a binary image as a result image. In the implementation carried out there are thresholding mean methods, median thresholding method, mean & median (non-variance) thresholding method, mean & median thresholding method, mean & median (variance) thresholding method, and Otsu thresholding method.

In software built, edge detection quality testing will use the Berkeley image database. Software testing is done using the Mean Square Error (MSE) and Jaccard coefficients. From the research that has been done, the software built has been able to detect the edges of objects in the image. The result of software testing is that the thresholding method used will affect the output image. Likewise with the parameter values on the BFOA algorithm. The BFOA parameter besides influencing the result image, also affects the execution time needed

Keywords: USAN, Bacterial Foraging Optimization Algorithm, Fuzzy, Fuzzy Intensifier, Edge quality and sharpness, Mean Square Error (MSE), Jaccard coefficients.

Dipersembahkan kepada pembimbing, orang tua, teman-teman dan diri sendiri.

KATA PENGANTAR

Puji dan syukur ke hadirat Tuhan Yang Maha Kuasa atas segala rahmat yang diberikan-Nya sehingga penulis bisa menyelesaikan skripsi dengan tepat waktu, yang kami berjudul "Deteksi Tepi pada Citra Berwarna dengan Bacterial Foraging Algorithm". Didalam pengerjaan skripsi ini telah melibatkan banyak pihak yang sangat membantu dalam banyak hal. Oleh sebab itu, disini penulis sampaikan rasa terima kasih kepada:

1. Bapak Husnul selaku pembimbing yang telah memberikan bimbingan dan masukan selama pengerjaan skripsi.
2. Buk Vania dan Buk Natalia selaku dosen penguji yang sudah banyak membantu memberikan kritik dan saran terhadap perbaikan skripsi ini.
3. Orang tua yang telah memberikan doa dan dukungan kepada penulis secara moril maupun materil hingga skripsi ini dapat selesai.
4. Sahabat dan rekan seperjuangan yang memberi bantuan, dukungan dan motivasi kepada penulis.
5. Semua pihak yang telah banyak membantu dalam penyusunan skripsi ini yang tidak bisa penulis sebutkan semuanya.

Penulis berharap agar skripsi ini dapat bermanfaat bagi masyarakat umum.

Bandung, Mei 2019

Penulis

DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
1 PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Tujuan	2
1.4 Batasan Masalah	2
1.5 Metodologi	2
1.6 Sistematika Pembahasan	3
2 LANDASAN TEORI	5
2.1 Pengolahan Citra [1]	5
2.2 Color Model[2]	5
2.2.1 RGB Color Model	5
2.2.2 HSI Color Model	6
2.2.3 RGB to HSI	7
2.3 Pendeteksian Tepi[1]	7
2.3.1 Definisi Pendeteksian Tepi	7
2.3.2 Tujuan Pedeteksian Tepi	9
2.4 Univalve Segment Assimilating Nucleus(USAN)[3]	9
2.5 Logika Fuzzy[4][5]	11
2.6 Fuzzification[3]	13
2.7 Bacterial Foraging Optimization Algorithm (BFOA)[6][3][7][8][9]	15
2.8 Fuzzy measures and optimization[3]	18
2.9 Thresholding[10][11]	19
2.10 Metode Pengukuran Kualitas	22
3 ANALISIS	23
3.1 Analisis Masalah	23
3.1.1 USAN	23
3.1.2 Fuzzify USAN area	27
3.1.3 Bacterial Foraging Algorithm & Fuzzy measures and optimization	28
3.1.4 De-fuzzification & Thresholding	37
3.1.5 Metode Pengukuran Kualitas	45
3.2 Analisis Kebutuhan Perangkat Lunak	47
3.2.1 Deskripsi Perangkat Lunak	47
3.2.2 Kebutuhan Fungsional Perangkat Lunak	47

3.2.3	Data uji dan data validasi	50
3.2.4	Diagram Kelas Sederhana	51
4	PERANCANGAN	55
4.1	Kebutuhan masukan	55
4.2	Perancangan Antarmuka	55
4.3	Diagram Kelas Rinci Perangkat Lunak	56
5	IMPLEMENTASI DAN PENGUJIAN	65
5.1	Implementasi	65
5.1.1	Lingkungan Perangkat Keras	65
5.1.2	Lingkungan Perangkat Lunak	65
5.1.3	Implementasi Antarmuka Perangkat Lunak	65
5.1.4	Implementasi Perangkat Lunak	66
5.2	Pengujian	72
5.2.1	Pengujian Fungsional	72
5.2.2	Pengujian Eksperimental	72
6	KESIMPULAN DAN SARAN	99
6.1	Kesimpulan	99
6.2	Saran	99
	DAFTAR REFERENSI	101
	A KODE PROGRAM	103
	B HASIL EKSPERIMEN	117
B.1	Hasil Pengujian Metode Thresholding	117
B.2	Hasil Pengujian Parameter BFOA	125

DAFTAR GAMBAR

2.1	RGB	6
2.2	Nilai Hue dan Saturation pada warna	6
2.3	HSI	7
2.4	Model tepi	8
2.5	Piksel tetangga	8
2.6	Jenis-jenis tepi	9
2.7	Contoh perubahan arah tepi landai	9
2.8	Matriks USAN	10
2.9	USAN <i>mask</i> dengan radius = 3[3]	10
2.10	Kurva linear naik	12
2.11	Kurva linear turun	12
2.12	Kurva segitiga	13
2.13	Kurva trapesium	13
2.14	Kurva Gaussian	14
2.15	Proses <i>swim</i> dan <i>tumble</i> bakteri[9]	16
2.16	Sekelompok bakteri yang berada di permukaan dimensi pencarian[9]	16
2.17	Contoh histogram bimodal dengan nilai <i>threshold</i> T	20
2.18	Jaccard	22
3.2	Citra contoh 10 × 10 piksel	23
3.1	<i>Flow chart</i> penyelesaian masalah	24
3.3	<i>Flowchart</i> BFOA	31
3.4	<i>Flowchart</i> BFOA	32
3.5	Citra hasil metode <i>thresholding</i> mean	41
3.6	Citra hasil metode <i>thresholding</i> median	41
3.7	Citra hasil metode <i>thresholding</i> median	42
3.8	Citra hasil metode <i>thresholding</i> median	43
3.9	Citra hasil metode <i>thresholding</i> mean & median (variance) & Otsu	43
3.10	Citra hasil metode <i>thresholding</i> Otsu	45
3.11	Citra tepi seharusnya	45
3.12	Use Case Diagram	48
3.13	<i>Diagram Aktivitas</i>	51
3.14	Citra uji berwarna	52
3.15	Citra tepi atau citra validasi	53
3.16	Kelas Diagram Sederhana	53
4.1	Rancangan Antarmuka Perangkat Lunak	55
4.2	Kelas Diagram USAN	56
4.3	Kelas Diagram Bakteri	58
4.4	Kelas Diagram Koloni	58
4.5	Kelas Diagram BFOA	59
4.6	Kelas Diagram Edge_Detection	62

5.1	Antarmuka Perangkat Lunak	66
5.2	Citra hasil dengan metode <i>thresholding</i> mean	74
5.3	Citra hasil dengan metode <i>thresholding</i> median	75
5.4	Citra hasil dengan metode <i>thresholding</i> mean & median (non-variance)	75
5.5	Citra hasil dengan metode <i>thresholding</i> mean & median (non-variance)	76
5.6	Citra hasil dengan metode <i>thresholding</i> mean & median (variance) & Otsu	77
5.7	Citra hasil dengan metode <i>thresholding</i> Otsu	78
5.8	Grafik waktu eksekusi pada masing-masing metode <i>thresholding</i>	78
5.9	Grafik nilai koefisien Jaccard pada masing-masing metode <i>thresholding</i>	79
5.10	Grafik nilai MSE pada masing-masing metode <i>thresholding</i>	79
5.11	Citra 101085.png pada pengujian parameter S dengan metode mean	81
5.12	Grafik waktu eksekusi rata-rata pada masing-masing nilai S	81
5.13	Grafik nilai koefisien Jaccard pada masing-masing nilai S dan metode <i>thresholding</i>	82
5.14	Grafik nilai MSE pada masing-masing nilai S dan metode <i>thresholding</i>	82
5.15	Citra 101085.png pada pengujian parameter N_c dengan metode mean	83
5.16	Grafik waktu eksekusi rata-rata pada masing-masing nilai N_c	84
5.17	Grafik nilai koefisien Jaccard pada masing-masing nilai N_c dan metode <i>thresholding</i>	84
5.18	Grafik nilai MSE pada masing-masing nilai N_c dan metode <i>thresholding</i>	85
5.19	Citra 101085.png pada pengujian parameter N_s dengan metode mean	86
5.20	Grafik waktu eksekusi rata-rata pada masing-masing nilai N_s	86
5.21	Grafik nilai koefisien Jaccard pada masing-masing nilai N_s dan metode <i>thresholding</i>	87
5.22	Grafik nilai MSE pada masing-masing nilai N_s dan metode <i>thresholding</i>	87
5.23	Citra 101085.png pada pengujian parameter N_{re} dengan metode mean	88
5.24	Grafik waktu eksekusi rata-rata pada masing-masing nilai N_{re}	89
5.25	Grafik nilai koefisien Jaccard pada masing-masing nilai N_{re} dan metode <i>thresholding</i>	89
5.26	Grafik nilai MSE pada masing-masing nilai N_{re} dan metode <i>thresholding</i>	90
5.27	Citra 101085.png pada pengujian parameter N_{ed} dengan metode mean	91
5.28	Grafik waktu eksekusi rata-rata pada masing-masing nilai N_{ed}	91
5.29	Grafik nilai koefisien Jaccard pada masing-masing nilai N_{ed} dan metode <i>thresholding</i>	92
5.30	Grafik nilai MSE pada masing-masing nilai N_{ed} dan metode <i>thresholding</i>	92
5.31	Citra 101085.png pada pengujian parameter P_{ed} dengan metode mean	93
5.32	Grafik waktu eksekusi rata-rata pada masing-masing nilai P_{ed}	93
5.33	Grafik nilai koefisien Jaccard pada masing-masing nilai P_{ed} dan metode <i>thresholding</i>	94
5.34	Grafik nilai MSE pada masing-masing nilai P_{ed} dan metode <i>thresholding</i>	94
5.35	Citra 101085.png pada pengujian parameter c_i dengan metode mean	95
5.36	Grafik waktu eksekusi rata-rata pada masing-masing nilai c_i	95
5.37	Grafik nilai koefisien Jaccard pada masing-masing nilai c_i dan metode <i>thresholding</i>	96
5.38	Grafik nilai MSE pada masing-masing nilai c_i dan metode <i>thresholding</i>	96

DAFTAR TABEL

2.1	Nilai inisial operator FINT	15
3.1	Tabel warna Red (R) pada RGB	25
3.2	Tabel warna <i>Green</i> (G) pada RGB	25
3.3	Tabel warna <i>Blue</i> (B) pada RGB	25
3.4	Tabel <i>grayscale</i>	26
3.5	Tabel grayscale hasil perluasan	26
3.6	Tabel perhitungan nilai SUSAN	26
3.7	Tabel hasil perhitungan nilai SUSAN	27
3.8	Tabel USAN	27
3.9	Tabel k	27
3.10	Tabel nilai <i>membership</i> 1	28
3.11	Tabel parameter awal	29
3.12	Tabel posisi bakteri	29
3.13	Tabel nilai <i>membership</i> 2	30
3.14	Tabel nilai <i>fuzzy edge sharpness factor</i>	30
3.15	Tabel <i>fuzzy edge quality</i>	33
3.16	Nilai <i>objective function</i> bakteri	33
3.17	Tabel matriks Δ bakteri ke-0	34
3.18	Posisi β, γ , dan f_h	34
3.19	Posisi β, γ , dan f_h	34
3.20	Posisi β, γ , dan f_h	35
3.21	Posisi α, β, γ , dan f_h	35
3.22	Tabel Posisi bakteri setelah melalui seluruh tahapan <i>chemotaxis</i>	35
3.23	Tabel nilai <i>objective function</i> dan <i>health</i>	36
3.24	Tabel bakteri setelah diurutkan berdasarkan nilai <i>health</i>	36
3.25	Tabel bakteri setelah diurutkan berdasarkan nilai <i>health</i>	37
3.26	Tabel bakteri setelah menghabiskan tahap <i>chemotaxis</i> dan <i>reproduction</i>	37
3.27	Tabel bakteri setelah menghabiskan tahap <i>chemotaxis</i> dan <i>reproduction</i>	38
3.28	Posisi optimal bakteri	38
3.29	Tabel nilai <i>membership</i> 1	38
3.30	Tabel nilai <i>membership</i> 2	39
3.31	Hasil substitusi nilai keanggotaan	39
3.32	Tabel hasil <i>de-fuzzification</i>	40
3.33	<i>Mask</i> untuk menghitung nilai <i>local thresholding</i> pada piksel (0,0)	40
3.34	<i>Mask</i> untuk menghitung nilai <i>local thresholding</i> pada piksel (2,2)	40
3.35	Tabel histogram	43
3.36	Tabel probabilitas	44
3.37	Nilai w_0 dan w_1 untuk nilai <i>threshold</i> 0	44
3.38	Nilai μ_0, μ_1 dan μ_T untuk nilai <i>threshold</i> 0	44
3.39	Nilai $\sigma_0^2, \sigma_1^2, \sigma_W^2$ dan σ_B^2 untuk nilai <i>threshold</i> 0	44
3.40	Tabel intensitas citra validasi	46

3.41	Tabel intensitas citra uji	46
3.42	Tabel nilai selisih kuadrat	46
3.43	Tabel nilai selisih kuadrat	47
3.44	Skenario memasukan citra	48
3.45	Skenario memilih metode <i>thresholding</i>	49
3.46	Skenario mengganti nilai parameter masukkan BFOA	49
3.47	Skenario memproses citra	49
3.48	Skenario reset perangkat lunak	50
3.49	Skenario menyimpan citra hasil	50
4.1	Tabel fungsi elemen yang terdapat pada rancangan antarmuka	56
5.1	Tabel Pengujian Fungsional Perangkat Lunak	72
B.1	Tabel Posisi Bakteri Pada Eksperimen Thresholding	117
B.2	Tabel Hasil Perhitungan Nilai Mean Square Error Pada Eksperimen Thresholding	119
B.3	Tabel Hasil Perhitungan Koefisien Jaccard Pada Eksperimen Thresholding	121
B.4	Tabel Running Time Pengujian Metode Thresholding Dalam Milisecond	123
B.5	Tabel Hasil Eksperimen Parameter S	126
B.6	Tabel Hasil Eksperimen Parameter N_c	128
B.7	Tabel Hasil Eksperimen Parameter N_s	130
B.8	Tabel Hasil Eksperimen Parameter N_{re}	131
B.9	Tabel Hasil Eksperimen Parameter N_{ed}	132
B.10	Tabel Hasil Eksperimen Parameter p_{ed}	133
B.11	Tabel Hasil Eksperimen Parameter c_i	134

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Citra adalah kombinasi antara titik, garis, bidang, dan warna untuk menciptakan suatu imitasi dari suatu objek fisik. Citra dapat dikelompokkan menjadi 2 kelompok, yaitu citra analog dan citra digital. Citra analog adalah citra yang terbentuk dari sinyal kontinyu seperti gambar pada monitor televisi, foto sinar X atau hasil CT scan. Sedangkan citra digital adalah citra yang dapat disimpan pada media penyimpanan dan dapat diolah dengan menggunakan komputer.

Sebuah citra digital dapat diwakili oleh sebuah matriks dengan ukuran L kolom dan P baris di mana sel ke L_i, P_j merupakan piksel dari sebuah citra. Piksel merupakan ukuran terkecil dari sebuah citra. Citra digital terbagi menjadi 3 jenis, yaitu citra biner, citra *grayscale*, dan citra berwarna. Citra biner merupakan citra yang hanya memiliki 2 warna, yaitu hitam dan putih. Citra *grayscale* merupakan citra yang memiliki 256 derajat keabuan. Citra berwarna memiliki lebih dari 16 juta warna.

Pada citra berwarna, terdapat beberapa model representasi warna, yaitu dengan model warna CMYK (*Cyan, Magenta, Yellow, Black*), model warna HSB (*Hue, Saturation, Brightness*), model warna RGB (*Red, Green, Blue*). Nilai yang terdapat pada suatu piksel pada citra merupakan gabungan nilai dari warna yang menyusunnya. Contohnya pada model RGB, nilai suatu piksel pada citra tersusun dari nilai warna merah, nilai warna hijau, dan nilai warna biru. Susunan nilai yang berbeda pada suatu piksel akan menghasilkan warna yang berbeda pula.

Sebuah citra umumnya memiliki beberapa objek yang terdapat di dalam citra tersebut. Tepi dari objek-objek tersebut seringkali perlu untuk diketahui atau dideteksi. Proses untuk mengetahui tepi dari objek-objek yang terdapat pada suatu citra disebut juga dengan proses deteksi tepi. Tujuan dari proses tersebut adalah untuk meningkatkan penampakan garis batas suatu daerah atau objek di dalam citra[1]. Proses deteksi tepi merupakan proses yang seringkali merupakan langkah pertama dalam melakukan segmentasi citra. Segmentasi citra bertujuan untuk memecah suatu citra ke dalam beberapa segmen dengan suatu kriteria tertentu.

Tepi pada suatu objek yang terdapat dalam citra dapat dilihat sebagai lokasi piksel yang memiliki perbedaan nilai intensitas secara ekstrem. Terdapat beberapa teknik yang dapat digunakan untuk melakukan deteksi tepi pada objek yang terdapat dalam suatu citra. Teknik-teknik untuk melakukan deteksi tepi secara tradisional ialah dengan menggunakan teknik Sobel, Canny, Roberts, Kirsch, dan Morphological Gradien. Teknik-teknik tradisional tersebut cenderung tidak mengatasi masalah *vagueness* atau keambiguan dalam menentukan apakah suatu piksel menjadi tepi dari objek atau bukan. Oleh karena itu, telah banyak dikembangkan pendeteksian tepi dengan menggunakan metode *fuzzy*. Teknik-teknik tradisional tersebut selain tidak mengatasi masalah *vagueness*, teknik tersebut juga memerlukan waktu eksekusi yang lama. Begitu juga dengan menggunakan teknik *fuzzy*, teknik *fuzzy* juga memerlukan waktu eksekusi yang lama. Selain dengan menggunakan teknik tradisional, deteksi tepi juga dapat dilakukan dengan menggunakan teknik yang lain. Teknik *Bacterial Foraging Optimization Algorithm* (BFOA) merupakan salah satu teknik yang dapat digunakan. Teknik BFOA akan digunakan bersamaan dengan teknik *fuzzy*. Namun gabungan dari BFOA dan *fuzzy* memiliki kompleksitas waktu eksekusi yang tinggi.

Selain menggunakan teknik BFOA dan metode *fuzzy*, juga diperlukan teknik *thresholding* untuk mendapatkan citra biner sebagai citra hasil. Untuk mendapatkan hasil tersebut, akan digunakan metode *thresholding*. Metode *thresholding* yang tersedia sangat banyak dan memiliki kompleksitas waktu eksekusi dan hasil yang berbeda-beda. Pada skripsi ini, akan dibuat sebuah perangkat lunak untuk mendeteksi tepi pada citra berwarna dengan menggunakan algoritma *bacterial foraging*. Dengan menggunakan perangkat lunak tersebut, pengguna akan mendapatkan hasil berupa tepi dari citra yang dimasukan oleh pengguna. Hasil deteksi tepi suatu citra dapat disimpan oleh pengguna dalam bentuk *file*. Hasil tersebut dapat digunakan untuk melakukan segmentasi citra ataupun untuk keperluan lainnya.

1.2 Rumusan Masalah

Berdasarkan latar belakang yang telah disebutkan di atas, maka rumusan masalah yang akan dibangun adalah:

1. Bagaimana deteksi tepi dengan menggunakan algoritma *bacterial foraging*?
2. Bagaimana membangun perangkat lunak untuk mendeteksi tepi dengan menggunakan algoritma *bacterial foraging*?

1.3 Tujuan

Tujuan penelitian dari makalah ini adalah:

1. Mempelajari deteksi tepi dengan menggunakan algoritma *bacterial foraging*.
2. Membangun perangkat lunak untuk mendeteksi tepi dengan menggunakan algoritma *bacterial foraging*.

1.4 Batasan Masalah

Batasan masalah pada penelitian ini adalah sebagai berikut

- Perangkat lunak yang dibangun memiliki citra masukan maksimum 512×512 piksel.
- Metode *thresholding* yang digunakan akan digunakan ialah metode *thresholding* mean, metode *thresholding* median, metode *thresholding* mean & median (non-variance), metode *thresholding* mean & median (variance), metode *thresholding* mean & median (variance) & Otsu, dan metode *thresholding* Otsu

1.5 Metodologi

Metodologi penelitian yang digunakan untuk menyusun penelitian ini adalah sebagai berikut:

- Melakukan studi pustaka mengenai pengolahan citra
- Melakukan studi pustaka mengenai pendeteksian tepi
- Melakukan studi pustaka mengenai USAN
- Melakukan studi pustaka mengenai model *fuzzy*
- Melakukan studi pustaka dan eksplorasi mengenai algoritma *bacterial foraging*.

- Melakukan analisis perangkat lunak untuk mendeteksi tepi pada citra berwarna dengan *bacterial foraging*.
- Melakukan perancangan perangkat lunak untuk mendeteksi tepi pada citra berwarna dengan *bacterial foraging*.
- Mengimplementasikan perangkat lunak untuk mendeteksi tepi pada citra berwarna dengan *bacterial foraging*.
- Melakukan pengujian dengan menggunakan beberapa citra yang beragam.
- Membuat kesimpulan terhadap penelitian yang telah dilakukan.

1.6 Sistematika Pembahasan

Untuk penulisan skripsi ini akan dibagi dalam enam bagian sebagai berikut:

- Bab 1 Pendahuluan
Bab ini berisi penjelasan mengenai latar belakang permasalahan, rumusan masalah, tujuan, batasan masalah, metodologi penelitian dan sistematika pembahasan yang digunakan untuk menyusun tugas akhir ini.
- Bab 2 Landasan Teori
Bab ini berisi teori-teori yang menunjang penelitian yang tengah dilakukan. Teori tersebut di antaranya adalah teori mengenai pengolahan citra, pendeteksian tepi, *Univalve Segment Assimilating Nucleus (USAN)*, logika *fuzzy*, *fuzzification*, *fuzzy measures and optimization*, *Bacterial Foraging Optimization Algorithm (BFOA)*, *Thresholding*, dan metode pengukuran kualitas.
- Bab 3 Analisis
Bab ini berisi analisis perangkat lunak yang akan dibangun, asumsi yang digunakan, data uji dan data validasi, *use case diagram*, skenario, dan diagram kelas sederhana.
- Bab 4 Perancangan
Bab ini berisi kebutuhan masukan perangkat lunak, perancangan perangkat lunak yang akan dibangun meliputi perancangan antarmuka dan diagram kelas rinci.
- Bab 5 Implementasi dan Pengujian
Bab ini berisi implementasi, pengujian fungsional, dan pengujian eksperimental perangkat lunak.
- Bab 6 Kesimpulan dan Saran
Bab ini berisi kesimpulan dari awal hingga akhir penelitian serta saran untuk pengembangan selanjutnya.