

Modifikasi Ant Colony Optimization Berdasarkan Gradient Untuk Deteksi Tepi Citra

Febri Liantoni¹, Nanik Suciati², Chastine Fatichah³

Program Studi Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember

Kampus ITS Keputih, Sukolilo, Surabaya, 60111, Jawa Timur

E-mail: ¹febri13@mhs.if.its.ac.id, ²nanik@if.its.ac.id, ³chastine@cs.its.ac.id

Masuk: 6 Januari 2015; Direvisi: 17 Januari 2015; Diterima: 20 Januari 2015

Abstract. Ant Colony Optimization (ACO) is an optimization algorithm which can be used for image edge detection. In traditional ACO, the initial ant are randomly distributed. This condition can cause an imbalance ants distribution. Based on this problem, a modified ant distribution in ACO is proposed to optimize the deployment of ant based gradient. Gradient value is used to determine the placement of the ants. Ants are not distributed randomly, but are placed in the highest gradient. This method is expected to be used to optimize the path discovery. Based on the test results, the use of the proposed ACO modification can obtain an average value of the Peak Signal to Noise Ratio (PSNR) of 12.724. Meanwhile, the use of the traditional ACO can obtain an average value of PSNR of 12.268. These results indicate that the ACO modification is capable of generating output image better than traditional ACO in which ants are initially distributed randomly.

Keywords: Ant Colony Optimization, gradient, Edge Detection, Peak Signal to Noise Ratio

Abstrak. Ant Colony Optimization (ACO) merupakan algoritma optimasi, yang dapat digunakan untuk deteksi tepi pada citra Pada ACO tradisional, semut awal disebarkan secara acak. Kondisi ini dapat menyebabkan ketidakseimbangan distribusi semut. Berdasarkan permasalahan tersebut, modifikasi distribusi semut pada ACO diusulkan untuk mengoptimalkan penempatan semut berdasarkan gradient. Nilai gradient digunakan untuk menentukan penempatan semut. Semut tidak disebarkan secara acak akan tetapi ditempatkan di gradient tertinggi. Cara ini diharapkan dapat digunakan untuk optimasi penemuan jalur. Berdasarkan hasil uji coba, dengan menggunakan ACO modifikasi yang diusulkan dapat diperoleh nilai rata-rata Peak Signal to Noise Ratio (PSNR) 12,724. Sedangkan, menggunakan ACO tradisional diperoleh nilai rata-rata PSNR 12,268. Hasil ini menunjukkan bahwa ACO modifikasi mampu menghasilkan citra keluaran yang lebih baik dibandingkan ACO tradisional yang sebaran semut awalnya dilakukan secara acak.

Kata Kunci: Ant Colony Optimization, gradient, deteksi tepi, Peak Signal to Noise Ratio

1. Pendahuluan

Perkembangan teknologi untuk teknik pengolahan citra telah berkembang pesat. Berbagai teknik dikembangkan untuk mempermudah pekerjaan manusia, baik sebagai pengolah citra, analisis citra maupun pengguna citra untuk berbagai tujuan dan keperluan. Seringkali citra yang digunakan tidak dalam kondisi yang ideal. Kondisi ini berupa gangguan bayangan, foto dan gambar kabur. Berbagai masalah tentang analisa dan perencanaan dapat berpengaruh pada hasil interpolasi, oleh karena itu diperlukan teknik pengolahan citra untuk memperoleh citra yang ideal.

Salah satu teknik pengolahan citra yang biasa dilakukan adalah deteksi tepi. Deteksi tepi merupakan proses penggalian informasi tepi dari sebuah gambar. Hal ini dianggap sebagai langkah dasar yang digunakan dalam sebagian besar aplikasi pengolahan citra (Verma, dkk, 2011). Tepi dalam sebuah gambar dapat dianggap sebagai batas antara dua daerah yang berbeda.

Banyak pendekatan telah digunakan untuk melakukan deteksi tepi pada sebuah gambar. Beberapa metode yang umum digunakan adalah sobel, prewitt, dan canny (Verma dkk, 2010), (Charu & Sunanda, 2013). Penelitian terbaru menggunakan Ant Colony Optimization (ACO) untuk melakukan deteksi tepi pada gambar. ACO merupakan metode heuristik yang meniru perilaku semut untuk memecahkan masalah optimasi diskrit (Dorigo, dkk, 2006). Semut menggunakan senyawa kimia khusus

yang disebut feromon untuk menandai jalur antara sumber makanan dan koloni mereka. Jalur feromon digunakan oleh semut berikutnya sebagai referensi untuk mencari makanan karena feromon meningkatkan kemungkinan jalan untuk dipilih.

Ada beberapa keuntungan dari ACO yang bisa digunakan untuk memecahkan berbagai masalah Non Polinomial (NP) (Dorigo. dkk, 2006), seperti *traveling salesman problem* (TSP), deteksi tepi, *network packet routing*, *vehicular routing*, *quadratic assignment problem*, dan sebagainya. Pada penelitian ini ACO digunakan untuk tujuan deteksi tepi pada sebuah gambar.

Pada ACO tradisional, proses penyebaran atau penempatan semut awal dilakukan secara acak. Kondisi ini dapat menyebabkan ketidakseimbangan distribusi semut yang kemudian mempengaruhi proses penemuan jalur. Ada beberapa penelitian yang membahas mengenai penyebaran semut awal. Seperti yang telah dilakukan Rahebi. dkk. (2010) yang melakukan penelitian dengan menggabungkan ACO dan algoritma genetika dalam memperbaiki penyebaran semut untuk meningkatkan konvergensi. Verma. dkk (2010) melakukan pendekatan Fuzzy-ACO. Jumlah semut dihitung dan ditempatkan di titik akhir pada gambar menggunakan detektor sobel. Teknik Derivatif Fuzzy mengimplementasikan faktor probabilitas fuzzy untuk menentukan piksel berikutnya yang paling mungkin untuk menjadi tepi. Deteksi tepi menggunakan *Ant Colony System* (ACS) dilakukan oleh Tian. dkk (2008). Perhitungan yang dilakukan dengan menghitung jumlah semut dan membandingkan dengan ACO.

Berdasarkan permasalahan tersebut, modifikasi distribusi semut pada ACO diusulkan untuk mengoptimalkan penempatan semut berdasarkan *gradient*. Nilai *gradient* pada suatu area menunjukkan adanya keberadaan tepi pada area tersebut. Pada penelitian ini, nilai *gradient* digunakan untuk menentukan penempatan semut awal. Semut tidak disebar secara acak, akan tetapi ditempatkan di *gradient* yang tinggi. Pada *gradient* tertinggi, perjalanan semut dilakukan dahulu untuk menentukan jalur. Cara ini diharapkan dapat digunakan untuk optimasi penemuan jalur.

2. Tinjauan Pustaka

2.1. *Ant Colony Optimization* (ACO)

ACO merupakan metode heuristik yang meniru perilaku semut untuk memecahkan masalah optimasi diskrit (Dorigo. dkk, 2006). Semut menggunakan senyawa kimia khusus yang disebut feromon untuk menandai jalur antara sumber makanan dan koloni mereka. Jalur feromon digunakan oleh semut berikutnya sebagai referensi untuk mencari makanan karena feromon meningkatkan kemungkinan jalan untuk dipilih.

Ada beberapa jenis ACO yaitu *Ant System* (AS), *Ant Colony System* (ACS), *Min-Max Ant System* (MMAS), *Elitist Ant System* (EAS), *Rank-Based Ant System* (ASRank), *Approximate Nondeterministic Tree Search* (ANTS). (Etemad dkk, 2011) Dari jenis-jenis tersebut AS dan ACS merupakan jenis ACO yang paling populer digunakan. (Charu. dkk, 2013). AS merupakan algoritma versi pertama ACO yang diusulkan pada tahun 1992. Kemudian metode ini berkembang menjadi ACS. (Dorigo. dkk, 2006).

Pada penelitian ini dipilih jenis ACS karena algoritma ini telah menerapkan penurunan konsentrasi feromon, sehingga kemungkinan semut terjebak dalam satu daerah eksplorasi tertentu menjadi lebih kecil. (Verma. dkk, 2011). Secara umum, *pseudocode* ditunjukkan pada Gambar 1.

```

Initialize
SCHEDULE_ACTIVITIES
    ConstructAntSolutions
    DoDaemonActions (optional)
    UpdateFeromons
END SCHEDULE_ACTIVITIES

```

Gambar 1. Pseudocode ACO

Initialize merupakan langkah yang dilakukan di awal proses. Dalam langkah ini dilakukan prosedur inialisasi, seperti pengaturan parameter dan penempatan nilai feromon awal. Adapun parameter yang digunakan yaitu jumlah semut (K), jumlah step konstruksi (L), iterasi (N), *feromon evaporation rate* (ρ), *feromon decay* (φ), faktor pembobot feromon (α), dan faktor bobot informasi heuristik (β).

ConstructAntSolutions merupakan kegiatan perjalanan semut. Proses konstruksi berisi sejumlah langkah-langkah konstruksi. Semut akan bergerak dalam suatu gambar sampai jumlah target langkah konstruksi terbentuk. Pada proses konstruksi ke (n^{th}) jumlah semut (k^{th}) akan berpindah dari node (i) ke node (j) yang mengikuti perpindahan probabilitas ($P_{i,j}^{(n)}$). Aturan *proportional pseudorandom* ini ditunjukkan pada Persamaan 1. Dimana (Ω_i) adalah node ketetanggaan dari semut yang diberikan pada node ke (i). (τ) adalah *update* feromon dan (η) adalah informasi heuristik.

$$P_{i,j}^{(n)} = \frac{(\tau_{i,j}^{n-1})^\alpha (\eta_{i,j})^\beta}{\sum_{j \in \Omega_i} (\tau_{i,j}^{n-1})^\alpha (\eta_{i,j})^\beta} \text{ if } j \in \Omega_i \quad (1)$$

DoDaemonActions merupakan solusi konstruksi yang dilakukan untuk tambahan langkah sebelum pembaruan nilai feromon. Kegiatan ini sebagai tindakan tambahan sebelum memperbarui nilai-nilai feromon. Proses ini tidak bisa dilakukan jika hanya dengan *single ant*.

UpdateFeromons merupakan kegiatan pembaruan feromon setelah proses konstruksi dan *daemon actions* dilakukan. Terdapat dua kali *update*, yaitu *update* global feromon dan *update* lokal feromon. Pembaruan lokal feromon dilakukan setiap kali langkah konstruksi. Pada tahap ini feromon akan mengalami kerusakan (*feromon decay*). Hal ini bertujuan untuk menurunkan konsentrasi feromon di tepi yang dilalui. *Update* local feromon ditunjukkan pada Persamaan 2, dimana (ϕ) merupakan kerusakan feromon dan (τ) merupakan *update* feromon. *Update* global feromon dilakukan setelah step konstruksi maksimal dalam satu iterasi dilalui. Pada tahap ini terjadi penguapan feromon. *Update* global feromon ditunjukkan pada Persamaan 3.

$$\tau_{i,j} = (1 - \phi)\tau_{i,j} + \phi\tau_0 \quad (2)$$

$$\tau_{i,j} = (1 - \rho)\tau_{i,j} + \rho\Delta\tau_{i,j} \quad (3)$$

Pada ACO terdapat aturan perpindahan dengan faktor probabilitas pada delapan piksel ketetanggaan (Anna. dkk, 2010). Proses ini dihitung dengan menggunakan matrik perpindahan probabilitas. Piksel dengan faktor probabilitas maksimum dalam mendeteksi ketetanggaan memiliki piksel tepi. Untuk mengurangi perpindahan yang berulang pada semut dilakukan aturan *stopping criteria*, yaitu perpindahan semut akan berhenti jika melewati jalur yang sudah dilewati semut yang lain dan ketika semua piksel ketetanggaan (delapan piksel) sudah dilewati semua oleh semut maka perpindahan akan berhenti. (Verma. dkk, 2011).

2.2. Gradient

Implementasi penggunaan *magnitude* dari *gradient* merupakan langkah awal pada pengolahan citra digital. (Gonzalez. dkk, 1992). Nilai *gradient* pada suatu area menunjukkan adanya keberadaan tepi pada area tersebut. *Gradient* terbentuk atas transisi atau perubahan warna secara gradual. Untuk fungsi $f(x,y)$ pada koordinat *gradient* (x,y) didefinisikan sebagai *two-dimensional column vector* dengan Persamaan 4. Untuk *magnitude* dari vektor ditunjukkan oleh Persamaan 5.

$$\nabla f = \begin{bmatrix} G_x \\ G_y \end{bmatrix} = \begin{bmatrix} \frac{\partial f}{\partial x} \\ \frac{\partial f}{\partial y} \end{bmatrix} \quad (4)$$

$$\nabla f = \left[\left(\frac{\partial f}{\partial x} \right)^2 + \left(\frac{\partial f}{\partial y} \right)^2 \right]^{1/2} \quad (5)$$

Komponen dari *gradient* vektor itu sendiri adalah operator linear, tetapi *magnitude* dari vektor adalah bukan merupakan operator linear karena merupakan operasi kuadrat dan akar kuadrat (Gonzalez. dkk, 1992). Dalam pengolahan citra, operator berbasis turunan menjadi dasar dalam penghitungan. Beberapa contoh yaitu menggunakan operator sobel, prewit, robert. (Gonzalez. dkk, 1992). Gambar 2 merupakan *pseudo-convolution kernels* yang digunakan untuk menghitung *gradient magnitude* dengan cepat.

a ₁	a ₂	a ₃
a ₄	a ₅	a ₆
a ₇	a ₈	a ₉

Gambar 2. *Pseudo-convolution kernels* ukuran 3 x 3

Salah satu metode penentuan *gradient* citra yaitu operator sobel. Langkah pertama yang dilakukan operator sobel adalah memperkirakan nilai *gradient* horisontal (G_x) dan *gradient* vertikal (G_y). Cara ini dilakukan dengan operasi kernel terhadap matrik citra seperti yang ditunjukkan pada Gambar 2. Matrik kernel untuk G_x , didefinisikan sebagai operator K_x , sedangkan matrik kernel untuk G_y didefinisikan sebagai K_y , seperti pada Persamaan 7.

$$K_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad K_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} \quad (7)$$

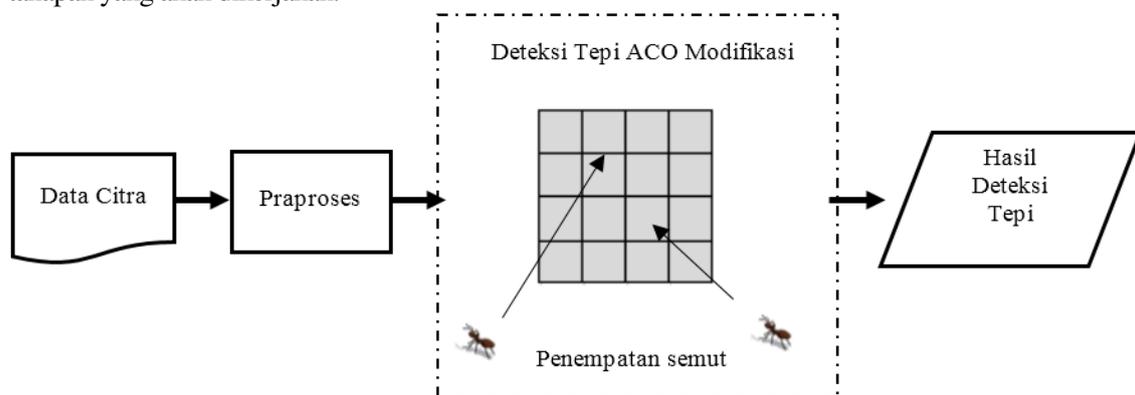
Setelah nilai *gradient* pada arah vertikal dan horisontal didapatkan, kemudian dicari nilai penghitungan matrik yang menghasilkan nilai *gradient* citra. Nilai *gradient* pada arah vertikal dan horisontal juga digunakan untuk menentukan arah sudut dari tepi (θ). Nilai dari θ akan mempengaruhi perlakuan suatu piksel terhadap piksel tetangganya. Penghitungan nilai arah tepi dihitung dengan Persamaan 8.

$$\theta = \tan^{-1} \left(\frac{G_y}{G_x} \right) \quad (8)$$

3. Metodologi Penelitian

3.1. Metode yang Diusulkan

Pada bagian ini akan dibahas tentang metode *Ant Colony Optimization* (ACO) modifikasi yang digunakan pada penelitian ini. Pembahasan lain mengenai langkah-langkah yang dilakukan untuk memecahkan permasalahan yang sedang diteliti. Dalam penelitian ini akan dilakukan beberapa tahapan meliputi pemilihan data citra, praproses dan deteksi tepi. Gambar 3 berikut menggambarkan ilustrasi tahapan yang akan dikerjakan.



Gambar 3. Diagram metode penelitian

3.1.1. Data Citra

Terdapat dua jenis data yang digunakan pada penelitian ini, yaitu data dengan objek yang memiliki nilai kontras yang berbeda dengan *background* dan data dengan objek kompleks. Gambar dengan kontras berbeda digunakan sebagai uji coba pengaruh penempatan semut berdasarkan *gradient*.

Sedangkan gambar kompleks digunakan untuk melihat hasil akurasi deteksi tepi.

Data yang digunakan berupa gambar daun dengan *background* berwarna putih, gambar daun hasil foto, gambar peta dunia dan berbagai macam gambar kompleks. Contoh gambar dengan kontras yang berbeda dengan *background* dan gambar dengan objek kompleks seperti pada Gambar 4.



Gambar 4. Contoh data citra

3.1.2. Praproses

Data masukan yang digunakan dalam sistem adalah citra RGB. Dari data citra RGB tersebut diubah menjadi citra keabuan. Proses perubahan citra warna menjadi citra keabuan menggunakan metode penghitungan *lightness* dan *average*. Selain metode itu ada juga metode *luminosity* yang merupakan pengembangan dari metode *lightness* dan *average*. Metode ini menghitung nilai setiap elemen warna dari citra, yaitu *red*, *green* dan *blue*. Penghitungan ini dengan cara menambahkan bobot sesuai persepsi penglihatan manusia.

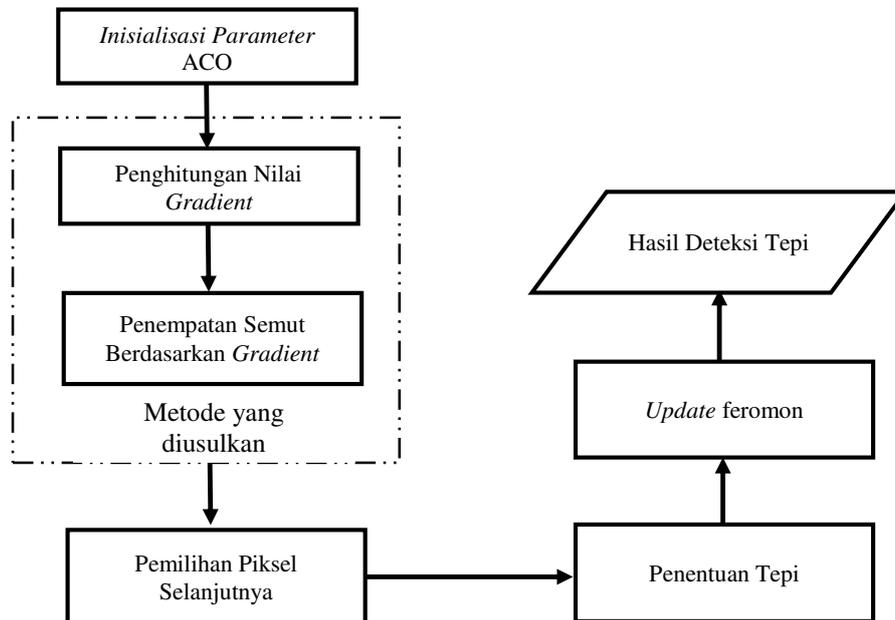
Metode *luminosity* dilakukan dengan menghitung nilai setiap elemen warna *R*, *G*, dan *B* dengan menambahkan bobot sesuai persepsi penglihatan manusia. Penglihatan manusia lebih sensitif terhadap warna hijau, oleh karena itu elemen *G* memiliki bobot yang paling tinggi dari elemen warna lainnya. Proses penghitungan untuk konvensi citra warna menjadi citra keabuan dapat dirumuskan secara matematis seperti Persamaan 9.

$$Luminosity = \frac{R*299+G*587+B*114}{1000} \quad (9)$$

Pada penelitian ini metode *luminosity* digunakan untuk melakukan konversi citra warna menjadi citra keabuan karena sifat metode *luminosity* yang lebih dekat dengan persepsi penglihatan manusia. Setelah diperoleh citra keabuan kemudian dilakukan deteksi tepi menggunakan metode ACO modifikasi.

3.1.3. Deteksi Tepi ACO Modifikasi

Ada beberapa tahapan yang dilakukan pada deteksi tepi ACO modifikasi. Pada penelitian ini diusulkan ACO modifikasi pada tahapan proses penempatan semut awal berdasarkan nilai *gradient*. Gambar 5 adalah blok diagram proses deteksi tepi yang akan dilakukan pada ACO modifikasi.



Gambar 5. Blok diagram metode ACO modifikasi

Inisialisasi parameter ACO, tahapan awal dari metode ACO modifikasi adalah inisialisasi parameter masukan. Parameter terdiri dari jumlah semut (K), jumlah step konstruksi (L), iterasi (N), *pheromone evaporation rate* (ρ), feromon peluruhan (ϕ), faktor bobot feromon (α), dan faktor pembobot informasi heuristik (β). Matrik informasi heuristik berupa matrik intensitas warna setiap piksel. Matrik feromon awal diinisialisasi satu atau nol. Jumlah semut yang digunakan sebagai parameter masukan pada penelitian ini berdasarkan Persamaan 10.

$$K = \sqrt{\text{width} * \text{height}} \tag{10}$$

Penghitungan nilai *gradient*, proses ini bertujuan untuk mengetahui potensi nilai tepi dari area gambar. Operator *gradient* dapat direpresentasikan oleh dua buah kernel konvolusi (G_x) dan (G_y), yang masing-masing mendefinisikan operasi penghitungan *gradient* dalam arah sumbu (x) dan sumbu (y) yang saling tegak lurus.

Penempatan semut, proses penempatan semut dilakukan dengan menyesuaikan potensi tepi berdasarkan nilai *gradient*. Semut tidak disebar secara acak, akan tetapi ditempatkan pada piksel yang memiliki *gradient* yang tinggi. Pada *gradient* tertinggi, perjalanan semut dilakukan dahulu untuk menentukan jalur. Cara ini diharapkan dapat digunakan untuk optimasi penemuan jalur dalam proses ACO.

Pemilihan piksel, proses perpindahan piksel ke piksel tetangganya berdasarkan derajat eksplorasi (q). Aturan *proportional pseudorandom* menggunakan ambang batas yang ditetapkan (q_0), yang nilainya antara nol hingga satu. Hubungan ketetanggaan menunjukkan nilai intensitas yang mewakili tiap piksel dari gambar. Nilai intensitas diperoleh dari informasi heuristik (η_{ij}) yang ditunjukkan pada Persamaan 11.

$$\eta_{ij} = \frac{\max_{ij} \begin{pmatrix} |I(i-1,j-1)-I(i+1,j+1)|, \\ |I(i-1,j+1)-I(i+1,j-1)|, \\ |I(i,j-1)-I(i,j+1)|, \\ |I(i-1,j)-I(i+1,j)| \end{pmatrix}}{\eta_{max}} \tag{11}$$

Pada gerakan setiap semut yang dilakukan secara acak menghasilkan (q) yang nilainya disebarkan antara nol dan satu. Jika (q) lebih besar dari (q_0), maka aturan *proportional pseudorandom* digunakan untuk menentukan gerakan semut. Namun, jika (q) lebih kecil dari (q_0), semut harus bergerak sesuai dengan transisi yang memaksimalkan nilai *update* feromon (τ) dan informasi heuristik (η). Aturan *proportional pseudorandom* diberikan dalam Persamaan 12.

$$P_{i,j}^{(n)} = \frac{(\tau_{i,j}^{n-1})^\alpha (\eta_{i,j})^\beta}{\sum_{j \in \Omega_i} (\tau_{i,j}^{n-1})^\alpha (\eta_{i,j})^\beta} \text{ if } j \in \Omega_i \quad (12)$$

Update feromon, langkah ini digunakan sebagai informasi heuristik akan menarik semut untuk mengikuti jalur perjalanan semut lainnya. Karena itu, feromon akan diperbarui dua kali, pertama setelah perpindahan yang dilakukan tiap semut dan kedua setelah perpindahan yang dilakukan semua semut. Terdapat dua mekanisme pembaruan feromon (τ) yaitu pembaruan feromon lokal dan pembaruan feromon global. Matrik feromon lokal akan diperbarui setiap langkah iterasi. Pada langkah ini feromon akan mengalami kerusakan (*pheromone decay*). Pembaruan feromon lokal bertujuan untuk memverifikasi pencarian yang dilakukan oleh semut berikutnya selama iterasi (Dorigo. dkk, 2006).

3.2. Pengujian

Penilaian secara numerik dapat digunakan untuk mengetahui akurasi hasil uji coba deteksi tepi yang dilakukan. Pengujian hasil deteksi tepi citra dapat dilakukan berdasarkan metode *Peak Signal to Noise Ratio* (PSNR) dan *Mean Square Error* (MSE). PSNR merupakan sebuah hubungan perbandingan antara nilai maksimum dan nilai kerusakan (*noise*) yang menunjukkan ketelitian dari hasil proses. Nilainya akan tinggi pada citra berkualitas baik. (Agrawal dkk, 2012). Penghitungan nilai MSE ditunjukkan pada Persamaan 13, dimana m adalah tinggi citra, n adalah lebar citra, $I(i,j)$ adalah nilai piksel *frame* masukan, $K(i,j)$ adalah nilai piksel *frame* rekonfigurasi. Sedangkan penghitungan nilai PSNR ditunjukkan pada Persamaan 14, dimana MAX_I adalah nilai maksimum piksel.

$$MSE = \frac{1}{m \cdot n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - K(i,j)]^2 \quad (13)$$

$$PSNR = 10 \log_{10} \left(\frac{MAX_I^2}{MSE} \right) \quad (14)$$

Pada uji coba akan membandingkan nilai PSNR antara hasil deteksi tepi menggunakan metode ACO tradisional dan metode ACO modifikasi yang diusulkan. Pengujian dilakukan untuk mengetahui pengaruh nilai *gradient* terhadap penempatan semut, lama waktu yang dibutuhkan, nilai *Mean Square Error* (MSE) dan *Peak Signal to Noise Ratio* (PSNR) yang dihasilkan.

Lingkungan pengujian yang digunakan untuk implementasi dan uji coba pada penelitian ini menggunakan pemrograman bahasa *java* dengan *library* pada aplikasi *java netbean*. Selain itu digunakan juga *toolbox* MATLAB R2013a. Pada penelitian ini digunakan beberapa parameter masukkan sebagai nilai penghitungan metode ACO. Tabel 1 berikut merupakan parameter uji coba yang digunakan pada metode yang diusulkan.

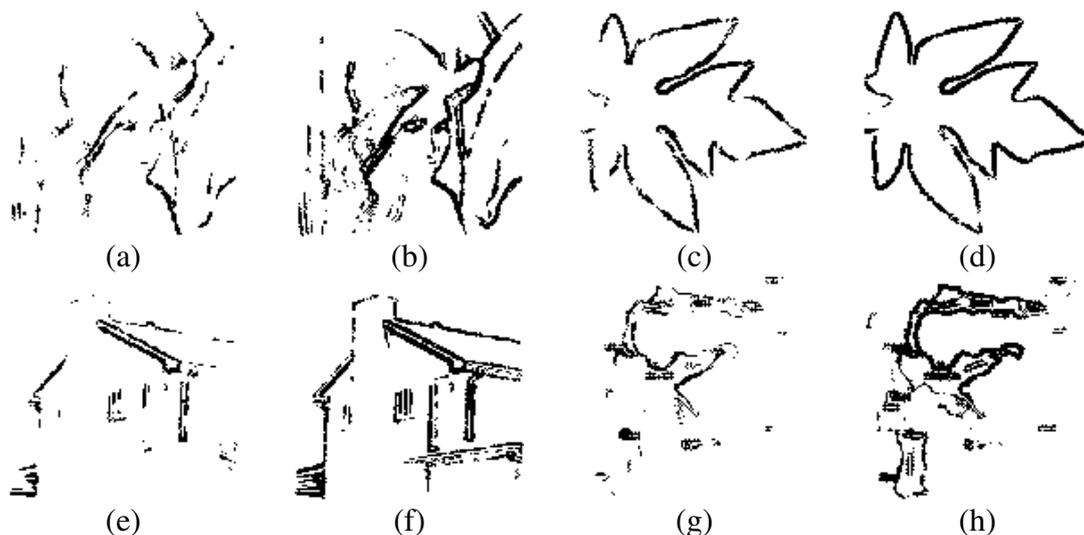
Tabel 1. Parameter uji coba

Parameter	Nilai
K (jumlah semut)	$\sqrt{\text{width} * \text{height}}$
L (jumlah konstruksi)	300
α (faktor pembobot feromon)	1
β (faktor pembobot informasi heuristik)	0,1
ρ (feromon penguapan)	0,1
φ (feromon peluruhan)	0,05
τ_{init} (feromon init)	0,0001
Ω (piksel ketetangaan)	8

4. Hasil dan Pembahasan

Pada bab ini dijelaskan mengenai skenario pengujian beserta hasil pengujian yang dilakukan dan analisis hasil uji yang diperoleh. Pengujian dilakukan untuk mengetahui pengaruh nilai *gradient* terhadap penempatan semut, lama waktu yang dibutuhkan, nilai *Mean Square Error* (MSE) dan *Peak Signal to Noise Ratio* (PSNR) yang dihasilkan. Pengujian juga digunakan untuk mengetahui performa sistem dalam mendeteksi tepi suatu gambar. Pengujian dilakukan dengan membandingkan nilai PSNR dan MSE yang dihasilkan ACO tradisional dengan metode yang diusulkan. Data uji coba yang digunakan pada penelitian ini sebanyak 40 gambar.

Pada uji coba yang dilakukan dengan membandingkan hasil deteksi tepi antara ACO modifikasi yang diusulkan dengan ACO tradisional. Sebagai pembanding, ACO tradisional akan dilakukan proses deteksi tepi sebanyak 10 kali tiap masing-masing gambar, kemudian hasil deteksi tepi akan dipilih nilai PSNR tertinggi. Nilai PSNR tertinggi ini akan digunakan sebagai pembanding dengan metode ACO modifikasi. Contoh hasil uji coba deteksi tepi antara ACO modifikasi yang diusulkan dengan ACO tradisional yang dilakukan seperti pada Gambar 6.



Gambar 6. Hasil deteksi tepi ACO tradisional (a) (c) (e)(g); ACO modifikasi (b) (d) (f) (h)

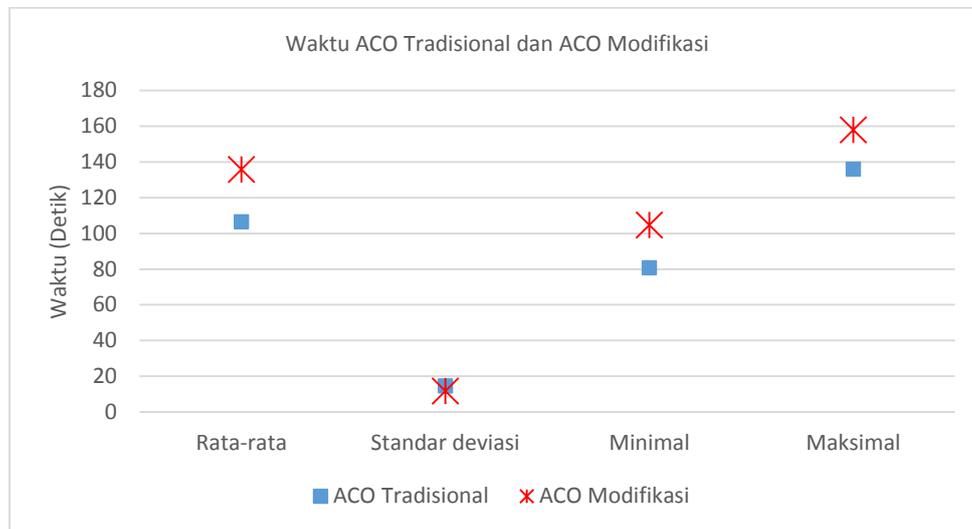
Pada Gambar 6 menunjukkan hasil deteksi tepi dari metode ACO modifikasi memiliki garis tepi yang lebih detail dibandingkan metode ACO tradisional. Gambar *lenna* (b) hasil deteksi tepi dengan ACO modifikasi memiliki garis tepi yang lebih detail dibanding *lenna* (a) dari hasil ACO tradisional. Gambar daun (d) hasil deteksi tepi dengan ACO modifikasi lebih halus dibanding daun (c) dari hasil ACO tradisional. Begitu juga pada gambar *house* (f) dan peta Sulawesi (h) dari hasil ACO modifikasi juga menghasilkan tepi yang lebih detail dibandingkan gambar *house* (e) dan peta Sulawesi (g) dari hasil ACO tradisional. Hasil deteksi tepi untuk gambar dengan kontras yang berbeda dengan *background* seperti gambar daun menunjukkan hasil yang bagus. Sedangkan untuk gambar kompleks seperti gambar *lenna* dan *house* terdapat banyak informasi tepi yang hilang.

Dari hasil uji coba yang telah dilakukan menunjukkan hasil deteksi tepi ACO modifikasi lebih baik dibandingkan ACO tradisional. Hal ini membuktikan proses penempatan semut berdasarkan nilai *gradient* berpengaruh dalam menentukan tepi dari sebuah gambar dengan metode ACO. Pada Tabel 2 merupakan nilai hasil uji coba dengan ACO modifikasi yang dibandingkan dengan ACO tradisional.

Tabel 2. Nilai uji coba ACO modifikasi dan ACO tradisional

Hasil	Waktu (Detik)		MSE		PSNR	
	Tradisional	Modifikasi	Tradisional	Modifikasi	Tradisional	Modifikasi
<i>Mean</i>	106,6	135,7	0,082	0,075	12,268	12,724
<i>Stdev</i>	14,7	11,8	0,062	0,057	3,721	3,796
<i>Min</i>	80,7	104,7	0,014	0,012	6,486	6,955
<i>Max</i>	136,0	158,0	0,225	0,202	18,565	19,357

Proses untuk menjalankan metode ACO modifikasi dan ACO tradisional membutuhkan waktu yang bervariasi. Grafik waktu yang dibutuhkan metode ACO modifikasi dan ACO tradisional ditunjukkan pada Gambar 7.



Gambar 7. Grafik waktu pada ACO tradisional dan ACO modifikasi

Dari uji coba yang telah dilakukan menunjukkan bahwa waktu yang dibutuhkan untuk menjalankan proses deteksi tepi menggunakan ACO tradisional lebih cepat dibandingkan menggunakan ACO modifikasi yang diusulkan. Kondisi ini disebabkan proses yang dilakukan untuk menempatkan semut memerlukan waktu yang relatif lama. Waktu tersebut dibutuhkan untuk melakukan perhitungan nilai *gradient*, menentukan tingkatan *gradient* dan penempatan posisi semut pada gambar.

Pada uji coba menggunakan metode ACO tradisional untuk proses deteksi tepi didapatkan hasil rata-rata PSNR 12,268, sedangkan saat menggunakan metode ACO modifikasi yang diusulkan didapatkan hasil rata-rata PSNR yang lebih baik yaitu naik menjadi 12,724. Nilai standar deviasi PSNR ACO modifikasi mencapai 3,796 lebih besar dari ACO tradisional 3,721. Kondisi ini disebabkan metode ACO tradisional masih melakukan penempatan semut secara acak, sedangkan ACO modifikasi yang diusulkan dengan melakukan penempatan posisi semut pada gambar berdasarkan nilai *gradient* sehingga menghasilkan deteksi tepi yang lebih optimal.

Hasil rata-rata PSNR dari uji coba yang dilakukan pada ACO modifikasi dan ACO tradisional hanya memiliki perbedaan yang sedikit. Hal ini dikarenakan ACO modifikasi masih menggunakan proses pergerakan semut secara acak seperti halnya pada ACO tradisional. Proses pergerakan semut untuk mencari dan menentukan tepi piksel ketetanggaan berpengaruh terhadap jalur tepi yang dihasilkan. Jika semut bergerak secara acak dan memilih piksel yang bukan tepi akan menyebabkan jalur tepi yang terputus-putus.

5. Kesimpulan

Proses penempatan posisi semut awal pada metode ACO berpengaruh pada hasil deteksi tepi yang didapatkan. Waktu yang diperlukan untuk menjalankan ACO modifikasi yang diusulkan membutuhkan rata-rata waktu 135,7 detik, sedangkan menggunakan ACO tradisional membutuhkan rata-rata waktu 106,6 detik. Hal ini menunjukkan proses ACO modifikasi lebih lama dibandingkan proses ACO tradisional. Kondisi ini dikarenakan ACO modifikasi membutuhkan waktu untuk proses penghitungan *gradient* dan penempatan semut awal. Hasil deteksi tepi menggunakan ACO modifikasi yang diusulkan menunjukkan nilai rata-rata PSNR yang lebih tinggi sebesar 12,724 dibandingkan menggunakan ACO tradisional yang menghasilkan nilai rata-rata PSNR sebesar 12,268.

Kinerja metode ACO modifikasi yang diusulkan baik untuk gambar dengan kontras yang berbeda dengan *background*. Pada gambar kompleks ACO modifikasi menghasilkan deteksi tepi yang kurang mendetail akan tetapi masih lebih baik dibandingkan ACO tradisional. Hal ini menunjukkan

bahwa ACO modifikasi lebih baik dibandingkan dengan ACO tradisional. Penempatan posisi semut awal berdasarkan nilai *gradient* mampu menghasilkan deteksi tepi yang lebih baik daripada melakukan penempatan semut secara acak seperti yang dilakukan ACO tradisional.

6. Saran

Pergerakan atau perjalanan semut untuk menentukan tepi piksel ketetangaan secara acak perlu dilakukan perbaikan misalnya dengan menggunakan *gradient* arah dalam menentukan piksel berikutnya yang dipilih sebagai tepi.

Referensi

- Agrawal, Prateek, Kaur, Simranjeet, Dhiman, Amita. 2012. Analysis and Synthesis of an Ant Colony Optimization Technique for ImageEdge Detection, *IEEE International Conference on Computing Sciences*, 127:131.
- Anna, B.V, Oppus, C. 2010. Image Edge Detection Using Ant Colony Optimization, *International Journal of Circuits, Systems and Signal Processing*, Vol 4. Issue 2, 24-33.
- Charu, G., Sunanda, G. 2013. Edge Detection of an Image based on Ant Colony Optimization Technique, *International Journal of Science and Research*, Vol.2, Issue 6, 114-120.
- Dorigo, M., M. Birattari, Stutzle. T. 2006. Ant Colony Optimization: Artificial Ants as a Computational Intelligence Technique, *IEEE Computational Intelligence Magazine*, 28-39.
- Etemad, Ali. S., White, Tony. 2011, An Ant Inspired Algorithm for detection of Image Edge Feature, *Elsevier*, Vol.11, 4883-4893.
- Gonzalez. R., Woods. R. 1992. *Digital Image Processing*, Addison Wesley.
- Rahebi, Javad, Elmi, Zahra, Nia, A.F., Shayan, Kamran. 2010. Digital Image Edge Detection Using An Ant Colony Optimization Based On Genetic Algorithm, *Cybernetics and Intelligent Systems (CIS), 2010 IEEE Conference on*. IEEE, 145-149.
- Tian, Jiang, Yu, Weiyu, Xie Shengli. 2008. An Ant Colony Optimization Algorithm for Image Edge Detection, *IEEE Congress on Evolutionary Computation*, 751:756.
- Verma, Om P., Hanmandlu, M., Sultania, Kumar, A., Dhruv. 2010. A Novel Fuzzy Ant System for Edge Detection, *Computer and Information Science (ICIS), 2010 IEEE/ACIS 9th International Conference on*. IEEE, 228-233.
- Verma, Om P., Singhal, P., Garg, S., Deepti, S.C. 2011. Edge Detection Using Adaptive Thresholding and Ant Colony Optimization, *Information and Communication Technologies (WICT), 2011 World Congress on*. IEEE, 113-118.