

METODE HYBRID MAXIMUM TSALLIS ENTROPY DAN HONEY BEE MATING OPTIMIZATION UNTUK PENCARIAN MULTILEVEL THRESHOLD PADA CITRA GRAYSCALE

Yufis Azhar, Maskur, Ali S. Kholimi
Jurusan Teknik Informatika, Fakultas Teknik
Universitas Muhammadiyah Malang
Email : yufis.azhar@gmail.com

ABSTRAK

Thresholding merupakan metode yang cukup populer untuk segmentasi suatu gambar. Untuk mensegmentasi suatu gambar grayscale menjadi gambar biner, bi-level thresholding bisa digunakan. Sedangkan untuk mensegmentasi citra grayscale ke dalam beberapa varian digunakanlah multi-level thresholding. Metode Maximum Tsallis Entropy (MTT) adalah salah satu metode yang bisa digunakan untuk pencarian multi threshold pada suatu citra grayscale. Akan tetapi metode ini memiliki waktu komputasi yang sangat besar jika jumlah threshold yang dicari semakin banyak. Oleh karena itu, suatu metode baru diusulkan dalam penelitian ini yang merupakan penggabungan antara metode Maximum Tsallis Entropy dan algoritma Honey Bee Mating Optimization (HBMO) untuk mendapatkan multilevel threshold dari suatu citra grayscale dalam waktu yang relatif singkat. Metode penggabungan yang diusulkan ialah dengan memfungsikan algoritma MTT sebagai alat untuk mencari nilai fitness dari suatu individu dalam algoritma HBMO. Semakin baik nilai fitness yang dimiliki oleh individu, semakin baik pula threshold yang ditemukan. Hasil yang didapat dari uji coba menunjukkan bahwa algoritma hybrid ini mampu mencari multi threshold dengan tingkat akurasi mencapai 98% dan waktu komputasi hingga 10 kali lebih cepat dibandingkan dengan waktu komputasi dari metode MTT untuk mencari 3 level threshold.

Kata Kunci: *Thresholding, Maximum Tsallis Entropy, Honey Bee Mating Optimization*

1 PENDAHULUAN

Thresholding merupakan metode yang cukup populer untuk segmentasi suatu citra. Untuk mensegmentasi suatu citra grayscale menjadi citra biner, bi-level thresholding bisa digunakan. Sedangkan untuk mensegmentasi citra grayscale ke dalam beberapa varian digunakanlah *multi-level thresholding*. *Thresholding* banyak digunakan pada aplikasi pemrosesan citra seperti pengenalan tulisan tangan, diagnosa suatu penyakit, pengenalan wajah dan yang lainnya.

Ada banyak literatur yang membahas tentang *image thresholding*. Pada dasarnya metode *thresholding* bisa diklasifikasikan menjadi dua, yaitu metode *parametric* dan *nonparametric*. [1] Untuk pendekatan *parametric*, distribusi *gray-level* dari suatu citra diasumsikan mematuhi distribusi gaussian, kemudian pendekatan ini akan mencari nilai suatu parameter dari distribusi *gaussian* yang paling sesuai untuk histogram tersebut.

Pendekatan *nonparametric* akan berusaha menemukan *threshold* yang paling optimal yang memisahkan fraksi-fraksi dalam suatu histogram berdasar pada beberapa kriteria diskriminan. Kriteria Otsu [2] adalah salah satu metode pencarian *thresholding* dengan pendekatan *nonparametric* yang

paling populer. Otsu mencoba untuk mencari *thresholding* dari suatu histogram dengan cara memaksimalkan jarak antar varian kelas. Akan tetapi, formula yang digunakan untuk memaksimalkan jarak ini masih terlalu kompleks sehingga membutuhkan waktu komputasi yang cukup lama Huang dkk [3], memodifikasi metode Otsu untuk mengoptimasi pencarian *multi-level threshold* pada suatu citra. Di samping memaksimalkan jarak antar *class variance*, pengoptimasian pencarian *thresholding* juga menggunakan kriteria-kriteria yang lain

Para peneliti pada umumnya sependapat bahwa teknik pencarian menggunakan entropi maksimum adalah teknik pencarian yang tradisional, oleh karena itu banyak teknik lain dicoba untuk dikembangkan. Metode entropi maksimum tradisional pada umumnya merujuk pada konsep entropi yang diperkenalkan oleh Shannon [4]. Entropi Shannon secara umum banyak digunakan untuk mengevaluasi entropi dari distribusi densitas sebuah probabilitas. Namun, Entropi Shannon mempunyai kelemahan terhadap penanganan data yang mempunyai ketidakseimbangan dalam frekuensi data. Dalam kasus pencarian *thresholding*, Entropi Shannon akan menemui kesulitan saat terdapat suatu warna *foreground* yang memiliki frekuensi jauh lebih sedikit dari pada warna *background* yang ada di

sekitarnya. Tsallis mengajukan entropi yang lebih bersifat umum untuk memperbaiki kelemahan Entropi Shannon dengan menambahkan suatu parameter (q) yang mampu menjadikan distribusi probabilitas menjadi lebih sensitif. Metode *Maximum Tsallis Entropy* (MTT) ini cukup efisien untuk memecahkan permasalahan bi-level thresholding [5]. Akan tetapi, permasalahan muncul ketika metode ini dihadapkan pada kasus *multi-level thresholding*. Waktu komputasinya akan semakin meningkat secara signifikan dengan semakin banyaknya *threshold* yang harus ditemukan.

Yudhong, dkk [1] mengaplikasikan teknik *Maximum Tsallis Entropy* (MTT) dengan menggunakan algoritma *Artificial Bee Colony* yang terinspirasi oleh perilaku lebah dalam mencari makanan. Horng, dkk [6], mengaplikasikan teknik *Maximum Entropy* (MET) untuk pemilihan *thresholding* yang paling optimal dan menggunakan algoritma *honey bee mating optimization* (HBMO) untuk mempercepat proses pencariannya. Metode HBMO sendiri terinspirasi oleh perilaku koloni lebah ketika bereproduksi. Kelebihan dari algoritma HBMO dibandingkan dengan algoritma *Artificial Bee Colony* adalah adanya dua kali evaluasi pencarian individu yang terbaik, sehingga bisa mencapai nilai *threshold* yang lebih optimal.[7]

Oleh karena itu, dalam penelitian ini diusulkan suatu algoritma baru yang menggabungkan metode *Maximum Tsallis Entropy* dan *Honey Bee Mating Optimization* (MTHBMOT) untuk pencarian *multilevel threshold* pada suatu citra *grayscale* untuk mendapatkan nilai *threshold* yang lebih optimal dan lebih cepat.

2 METODE THRESHOLDING

2.1 Maximum Tsallis Entropy

Dalam pemrosesan citra, metode *Maximum Tsallis entropy* dapat dimanfaatkan untuk pencarian *threshold*. Pencarian *threshold* ini pada umumnya dipergunakan untuk pemisahan bagian *background* dan *foreground* dari suatu citra. Cara yang umum dipakai adalah mengubah citra tersebut terlebih dahulu menjadi citra *grayscale* kemudian menyusun histogram dengan menghitung berapa banyak jumlah pixel dari citra tersebut yang memiliki *graylevel* x . Tiap *graylevel* x kemudian akan diuji untuk menentukan *threshold* yang paling optimal.

Diasumsikan suatu citra memiliki *graylevel* i , dengan $i = \{1, 2, 3, \dots, L\}$. Probabilitas dari *graylevel* i direpresentasikan sebagai p_i . Jika citra tersebut dibagi menjadi 2 kelas, yaitu CA dan CB oleh *threshold* di *graylevel* t , maka bisa dikatakan bahwa CA memiliki *graylevel* 1 sampai t , dan CB memiliki *graylevel* $t+1$ sampai L . Akumulasi dari probabilitasnya, dapat dirumuskan sebagai berikut:

$$\omega_A = \sum_{i=1}^t p_i, [1] \quad (1)$$

$$\omega_B = \sum_{i=t+1}^L p_i. [1] \quad (2)$$

Akumulasi dari probabilitas sebagaimana pada persamaan (1) dan (2), dapat dinormalisasi untuk mendapatkan probabilitas dari tiap *graylevel* dalam CA dan CB sebagai berikut:

$$p_A = \frac{\{p_1, p_2, \dots, p_t\}}{\omega_A}, [1] \quad (3)$$

$$p_B = \frac{\{p_{t+1}, p_{t+2}, \dots, p_L\}}{\omega_B}. [1] \quad (4)$$

Maka untuk masing-masing kelas, persamaan *tsallis entropy* dapat dirumuskan sebagai berikut:

$$S_q^A(t) = \frac{1 - \sum_{i=1}^t (p_i^q)}{q-1}, [1] \quad (5)$$

$$S_q^B(t) = \frac{1 - \sum_{i=t+1}^L (p_i^q)}{q-1}. [1] \quad (6)$$

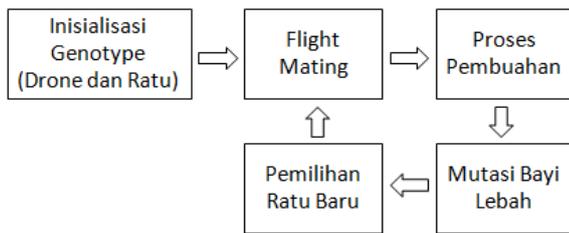
Sehingga untuk mendapatkan *tsallis entropy* dari keseluruhan citra dengan menggunakan metode *Maximum Tsallis entropy*, dapat ditulis sebagai berikut:

$$S_q(t) = S_q^A(t) + S_q^B(t) + (1 - q) \dots \\ \times S_q^A(t) \times S_q^B(t). [1] \quad (7)$$

Dengan menggunakan persamaan (7), nilai S_q yang paling maksimal akan dicari untuk mendapatkan nilai *threshold* yang paling optimal. Pencarian ini dilakukan secara berulang pada tiap *graylevel* yang terdapat pada citra.

Untuk pencarian 1 buah *threshold* (*bi-level thresholding*), metode *Maximum Tsallis entropy* ini dirasa cukup efisien dengan waktu komputasi yang tidak terlalu lama. Akan tetapi, masalah muncul ketika metode ini diterapkan pada *multi-level thresholding*. Fungsi *tsallis entropy* untuk keseluruhan citra akan menjadi semakin kompleks. Sebagai contoh untuk mencari 2 buah *threshold* saja, fungsinya akan terdefinisi sebagai berikut:

$$S_q(t) = S_q^A(t) + S_q^B(t) + S_q^C(t) \dots \\ + (1 - q). S_q^A(t). S_q^B(t) \dots \\ + (1 - q). S_q^A(t). S_q^C(t) \dots \\ + (1 - q). S_q^B(t). S_q^C(t) \dots \\ + (1 - q). S_q^A(t). S_q^B(t). S_q^C(t). [1] \quad (8)$$



Gambar 1. Metode MTHBMOT

Dari contoh pada persamaan (8), dapat diambil kesimpulan bahwa semakin banyak *threshold* yang akan dicari, maka semakin kompleks pula fungsi yang akan didefinisikan. Hal ini akan membuat waktu komputasinya juga akan semakin lama. Oleh karena itu, dalam penelitian ini algoritma HBMO diusulkan untuk mengoptimasi pencarian *thresholding* yang menggunakan metode *maximum tsallis entropy*.

2.2 Algoritma HBMO

Suatu koloni lebah madu biasanya terdiri atas seekor ratu yang dikelilingi oleh ribuan lebah jantan dan lebah pekerja. Biasanya dalam 1 koloni terdapat 10.000 hingga 60.000 ekor lebah pekerja. Satu koloni lebah bisa memiliki seekor atau lebih lebah ratu dalam daur hidupnya. Seekor lebah ratu bisa hidup hingga 6 tahun, dan hanya bertugas untuk bertelur. Sedangkan lebah jantan dan lebah pekerja tidak akan mempunyai usia lebih dari 6 bulan.

Lebah ratu mendapatkan telurnya setelah proses pembuahan oleh lebah jantan. Akan tetapi, lebah jantan ini akan mati setelah proses perkawinan tersebut. Sedangkan lebah pekerja, selain bertugas untuk memberi makan anggota koloni, juga bertugas untuk menjaga dan memelihara bayi lebah. Dari beberapa bayi lebah ini, jika terdapat bayi yang memiliki unsur genetik sama atau lebih baik dari ratu lebah, akan dipersiapkan untuk menjadi ratu berikutnya.

Dalam proses perkawinan, ratu lebah akan terbang menjauh dari sarang dengan diikuti oleh beberapa ekor lebah jantan. Dalam penerbangan ini, ratu lebah akan melakukan tarian yang menandakan bahwa ia siap untuk dibuahi. Lebah jantan yang mampu mengikuti kecepatan terbang ratu lebah kemudian akan melakukan proses perkawinan di udara.

Dalam setiap perkawinan, sperma dari lebah jantan akan disimpan dalam *spermatheca*. Proses perkawinan ini sendiri akan berakhir ketika energi ratu untuk terbang telah habis atau jumlah sperma yang tersimpan dalam *spermatheca* telah cukup. Nantinya, setiap kali ratu akan bertelur, akan diambil salah satu sperma yang ada di *spermatheca* untuk membuahi sel telur milik ratu. Setelah telur-telur

tersebut dilahirkan, lebah pekerja akan merawatnya sekaligus melihat apakah ada telur yang pantas untuk menjadi ratu berikutnya.

Dari penjelasan di atas, dapat dibentuk suatu algoritma dengan urutan sebagai berikut:

1. Algoritma dimulai dengan menginisialisasi *genotype* yang dimiliki oleh lebah ratu dan lebah jantan;
2. Dilanjutkan dengan proses perkawinan yang dilakukan oleh lebah ratu dan lebah jantan di udara, dimana lebah ratu (*best solution*) akan memilih lebah jantan secara probabilistik untuk membentuk *spermatheca*. Sebuah sel sperma kemudian akan dipilih dari *spermatheca* secara acak untuk dilakukan proses pembuahan;
3. Proses pembuahan dilakukan dengan cara perkawinan silang (*crossover*) antara sel telur ratu dan sel sperma lebah jantan;
4. Setelah individu baru dilahirkan, gunakan lebah pekerja untuk melakukan pencarian terhadap setiap individu baru, jika memungkinkan lakukan proses mutasi terhadap individu baru dengan memanfaatkan *royal jelly* yang dimiliki oleh lebah pekerja sebagai katalis;
5. Jika terdapat individu baru yang lebih baik dari ratu, ganti ratu yang lama (*best solution*) dengan individu baru tersebut dan lakukan proses perkawinan mulai dari awal kembali.

2.3 Algoritma MTHBMOT

Dalam algoritma yang ditawarkan, metode *Maximum Tsallis Entropy* (MTT) akan difungsikan untuk pencarian nilai *fitness* pada tiap individu dengan langkah-langkah yang sama seperti telah dijelaskan pada bagian sebelumnya. Sedangkan *graylevel* dari suatu citra yang disangka sebagai *threshold* berperan sebagai gen dari suatu individu lebah. Artinya jumlah *threshold* yang akan dicari menentukan jumlah gen yang dimiliki oleh setiap lebah. Secara umum algoritma yang ditawarkan dapat dilihat dalam Gambar 1.

Algoritma HBMO akan menjalankan peran sebagai algoritma optimasi dengan tahapan sebagai berikut:

1. Inisialisasi *genotype*
 Pada tahap ini, inisialisasi suatu matriks $D=[D1,D2,..., Dm]$, dimana D adalah individu lebah jantan yang akan melakukan proses perkawinan dengan ratu. Masing-masing lebah jantan memiliki *genotype* dengan panjang c . Sehingga didapatkan $D_i = (d_{i1},d_{i2},...,d_{ic})$, dimana d adalah satu buah gen dari individu D . Sedangkan seekor lebah ratu akan didefinisikan dengan suatu matriks $Q = (q1,q2,...,qc)$, dimana q adalah satu buah gen milik lebah ratu.

2. *Flight Mating*

Di tahap 2 ini, lebah jantan terbaik akan dipilih dengan menggunakan metode *simulated annealing* dan memasukkan *genotype* dari lebah jantan tersebut ke dalam *list* sperma yang ada di *spermatheca* melalui proses *flight mating*. Proses *flight mating* ini akan berlangsung terus menerus sampai jumlah sperma yang tertampung dalam *spermatheca* mencapai batas *n*. Variable *n* ini sendiri adalah kapasitas sperma yang mampu ditampung di *spermatheca* yang akan didefinisikan oleh *user* dan besarnya tidak boleh lebih dari jumlah individu lebah jantan yang ikut terbang selama proses perkawinan. Matriks yang mendefinisikan *spermatheca* didefinisikan oleh persamaan (9).

$$Sperm = [Sp_1, Sp_2, \dots, Sp_n]. \quad [6] \quad (9)$$

dimana *Sp-i* adalah sel sperma ke *i* dari suatu individu lebah jantan yang terdapat dalam *spermatheca* lebah ratu.

3. Proses Pembuahan

Pada tahap pembuahan ini, satu sel sperma yang terdapat dalam *spermatheca* akan diambil secara acak untuk kemudian dikawinkan dengan sel telur ratu yang disebut dengan proses *crossover*. Proses pembuahan ini sendiri akan menggabungkan gen yang dimiliki oleh lebah jantan dan lebah betina berdasarkan pada persamaan (10).

$$Brood_j = Q \pm \beta \times (Sp_j - Q), \quad [6] \quad (10)$$

dimana variable β adalah bilangan *real* antara 0 dan 1 yang dibangkitkan secara *random* selama proses pembuahan terjadi. Individu yang baru ini (*Brood*) akan memiliki jumlah gen yang sama seperti yang dimiliki oleh induknya.

$$Brood_j = (Brood_j^1, Brood_j^2, \dots, Brood_j^f). \quad [6] \quad (11)$$

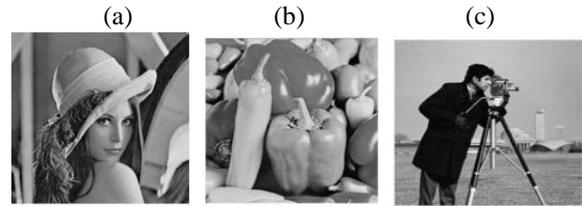
4. Mutasi Bayi Lebah

Untuk setiap individu baru yang dihasilkan. Bilangan acak *R* akan di-generate. Jika nilai dari *R* kurang dari nilai *Pm* yang didefinisikan oleh pengguna, maka proses mutasi akan dilakukan pada individu tersebut. Proses mutasi dilakukan dengan mengubah salah satu gen yang dimiliki oleh individu tersebut berdasarkan persamaan (12).

$$Brood_j^k = Brood_j^k \pm (\delta + \varepsilon) \times Brood_j^k.$$

$$\delta \in [0,1]; \quad 0 < \varepsilon < 1. \quad [6] \quad (12)$$

dimana δ adalah sebuah bilangan acak, sedangkan ε merupakan royal jelly milik lebah pekerja sebagai katalis yang digunakan untuk meningkatkan kualitas *genotype* dari suatu individu baru hasil dari mutasi yang nilainya didefinisikan oleh *user*.



Gambar 2. Test Citra : (a) LENA, (b) PEPPER, (c) CAMERA

5. Pemilihan ratu baru

Individu baru yang dihasilkan melalui proses pembuahan dan mutasi, akan dibandingkan nilai *fitness*-nya dengan lebah ratu yang sekarang. Jika individu baru ini memiliki nilai *fitness* yang lebih baik, maka individu baru tersebut akan menggantikan ratu lebah. Langkah selanjutnya kembali ke tahap pertama hingga sejumlah iterasi yang ditentukan oleh *user*.

3 SKENARIO UJI COBA

Algoritma MTHBMOT akan diujicoba untuk memecahkan masalah pencarian *multilevel thresholding* pada beberapa buah citra yang telah disiapkan. Metode *Maximum Tsallis Entropy* (tanpa optimasi) akan diujicoba pula untuk masalah yang sama sebagai perbandingan. Sebagai data uji akan digunakan 3 buah citra seperti yang dapat dilihat pada Gambar 2.

Dalam uji coba ini, 3 buah skenario telah disiapkan. Pada masing-masing skenario, algoritma MTHBMOT akan mencari *threshold* dari ketiga buah citra tersebut dengan jumlah *threshold* yang berbeda-beda, yaitu 2, 3 dan 4 buah *threshold*, dengan parameter untuk proses ujicoba seperti diperlihatkan dalam Tabel 1 yang didasarkan pada parameter uji coba yang dilakukan oleh Horng[7].

Hasil *threshold* yang didapatkan oleh MTHBMOT akan dibandingkan dengan hasil *threshold* yang didapatkan oleh algoritma *Maximum Tsallis Entropy* (MTT).

Tabel 1. Parameter pengujian

Parameter	Penjelasan	Nilai
<i>M</i>	Jumlah lebah jantan	150
<i>C</i>	Jumlah threshold	2, 3 atau 4
<i>L</i>	Rentang grayscale suatu citra	255
<i>A</i>	Speed reduction	0.98
<i>nsperm</i>	Kapasitas spermatheca	50
<i>SO</i>	Kecepatan awal ratu pada saat flight mating	1.0
<i>Pc</i>	Breeding ratio	0.8
<i>Pm</i>	Mutation ratio	0.01
<i>E</i>	Variasi mutasi	0.5

Tabel 2. Perolehan Treshold Dari Pengujian

Citra	c	Threshold yang didapatkan	
		MTHBMOT	MTT
LENA	2	80,150	80,150
	3	59,109,159	59,109,159
	4	55,99,143,180	55,100,144,182
PEPPER	2	74,146	74,146
	3	61,112,164	61,112,164
	4	57,104,148,194	57,104,148,194
CAMERA	2	103,191	103,191
	3	63,130,193	63,130,193
	4	60,107,158,202	60,107,158,202

4 HASIL UJI COBA

Dari Tabel 2, dapat dilihat bahwa algoritma MTHBMOT mampu melakukan pencarian *multilevel threshold* dari sebuah citra *grayscale* dengan hasil yang akurat. Hal ini dapat dilihat dari *threshold* yang didapatkan, dimana MTHBMOT mampu menemukan *threshold* yang sebagian besar sama dengan yang ditemukan oleh metode *Maximum Tsallis Entropy* (MTT). MTHBMOT memiliki tingkat akurasi sebesar 98% terhadap algoritma MTT.

Waktu komputasi yang diperlukan oleh algoritma MTHBMOT 10 kali lipat lebih cepat jika dibandingkan dengan algoritma MTT sebagaimana ditunjukkan oleh Tabel 3. Hal ini disebabkan waktu yang diperlukan oleh algoritma MTT untuk mencari *threshold* meningkat secara eksponensial sesuai dengan jumlah *threshold* yang dicari. Berbeda dengan algoritma MTT, algoritma MTHBMOT memiliki waktu komputasi yang cenderung stabil berapapun jumlah *threshold* yang dicari.

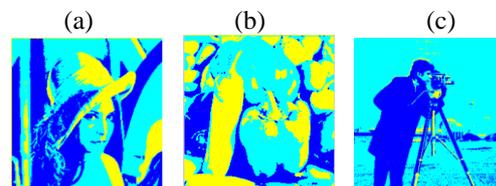
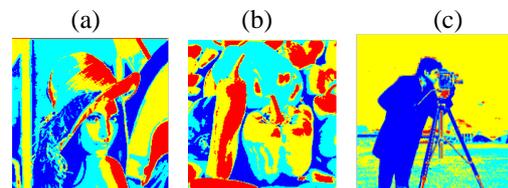
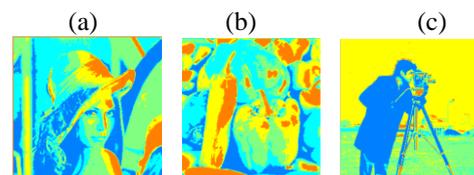
Dari hasil ujicoba implementasi metode MTHBMOT untuk pencarian *multilevel threshold* pada suatu citra *grayscale*, dapat dilihat bahwa metode ini memerlukan banyak inisialisasi parameter yang harus didefinisikan sendiri oleh *user*. Hal ini dapat berakibat pada menurunnya performa metode.

Sebagai contoh untuk pendefinisian kapasitas *spermatheca* yang dimiliki oleh ratu. Jika *user* memberikan nilai yang terlalu besar, ada kemungkinan tidak memerlukan iterasi yang terlalu banyak untuk menemukan *threshold* yang paling optimal. Akan tetapi waktu yang dibutuhkan pada setiap iterasi akan cenderung meningkat karena jumlah pembuahan akan semakin banyak. Demikian pula jika *user* mendefinisikan nilai kapasitas *spermatheca* ini terlalu kecil. Waktu yang dibutuhkan untuk setiap iterasi memang cukup singkat, tapi kemungkinan akan dibutuhkan banyak iterasi untuk menemukan *threshold* yang paling baik. Oleh karena itu, harus ada suatu metode yang dapat mengukur tingkat efisiensi suatu nilai yang didefinisikan oleh

user agar performa dari metode MTHBMOT ini tetap terjaga.

Tabel 3. Hasil Pengujian Waktu Komputasi

Citra	c	Waktu Komputasi (s)	
		MTHBMOT	MTT
LENA	2	46	4
	3	48	478
	4	55	>1hr
PEPPER	2	47	5
	3	49	482
	4	56	>1hr
CAMERA	2	47	5
	3	49	480
	4	55	>1hr


Gambar 3. Citra Hasil Pengujian 2 Threshold: (a) LENA, (b) PEPPER, (c) CAMERA

Gambar 4. Citra Hasil Pengujian 3 Threshold: (a) LENA, (b) PEPPER, (c) CAMERA

Gambar 5. Citra Hasil Pengujian 4 Threshold: (a) LENA, (b) PEPPER, (c) CAMERA

5 KESIMPULAN

Penelitian ini mengusulkan suatu metode baru untuk pencarian *multilevel thresholding* dari suatu citra *grayscale*, yaitu dengan menggabungkan metode *Maximum Tsallis Entropy* (MTT) dan Metode

Honey Bee Mating Optimization (HBMO). Ujicoba dilakukan dengan cara membandingkan algoritma tersebut dengan algoritma MTT tanpa optimasi.

Hasil yang didapat adalah algoritma MTHBMOT mampu mendapatkan *threshold* yang kurang lebih sama dengan algoritma MTT, akan tetapi dengan waktu komputasi yang jauh lebih singkat. Algoritma MTHBMOT hanya kalah dari algoritma MTT dalam hal waktu komputasi jika jumlah *threshold* yang dicari kurang dari 3. Ini membuktikan bahwa penggunaan metode HBMO secara efektif mampu mengoptimalkan penggunaan metode MTT dalam pencarian *multilevel thresholding* pada suatu citra *grayscale*.

6 DAFTAR PUSTAKA

- [1] Yudhong C., and Lenan Wu. 2011. "Optimal Multi-Level Thresholding Based on Maximum Tsallis Entropy via an Artificial Bee Colony Approach". *Entropy*. Vol. 13, Hal. 841–859.
- [2] Otsu N. 1979. "A threshold selection method from gray-level histograms". *IEEE Transactions on Systems Man Cybernetics*. Vol. 9. Hal. 62–66.
- [3] Huang D.Y. and Wang C.H. 2009. "Optimal multi-level thresholding using a two-stage Otsu optimization approach". *Pattern Recognition Letter*. Vol. 30, No. 3. Hal. 275–284.
- [4] Shannon C.E. 1948. "A Mathematical Theory of Communication". *Bell System Technical Journal*. Vol. 27. Hal. 379–423.
- [5] Tomasz Maszczyk and Włodzisław Duch. 2008. "Comparison of Shannon, Renyi and Tsallis Entropy used in Decision Trees." *Lecture Notes in Computer Sciences*. Vol. 5097. Hal. 643-651
- [6] Horng M.H. 2010. "A multilevel image thresholding using the honey bee mating optimization". *Applied Mathematics and Computation*. Vol. 215, No. 9. Hal. 3302–3310.
- [7] Hussein A. Abbas. 2001. "MBO: Marriage in Honey Bees Optimization A Haplometrosis Polygynous Swarming Approach." *Proceedings of the 2001 Congress on Evolutionary Computation*.