

**PENGGUNAAN ALGORITMA GENETIKA DALAM
PENINGKATAN KINERJA FUZZY CLUSTERING UNTUK
PENGENALAN POLA**
(Application of Genetic Algorithm to Enhance the Performance of Clustering
Fuzzy in Pattern Recognition)

Naniek Widyastuti¹, Amir Hamzah¹

¹Fakultas Teknologi Industri, Institut Sains & Teknologi AKPRIND Yogyakarta

ABSTRAK

Algoritma genetika (AG) adalah metode untuk menyelesaikan persoalan optimasi berbasis teori evolusi dalam biologi. Algoritma ini bekerja pada populasi calon penyelesaian yang disebut kromosom yang awalnya dibangkitkan secara random dari ruang penyelesaian fungsi tujuan. Dengan menggunakan mekanisme operator genetik yaitu persilangan dan mutasi populasi dievolusikan melalui fungsi fitness yang diarahkan pada kondisi konvergensi. Algoritma ini dapat diterapkan dalam banyak area fungsi-fungsi optimasi, salah satu penerapannya adalah fungsi objektif berbasis fuzzy clustering.

Fuzzy clustering bertujuan mengelompokkan n objek yang disajikan dengan vektor kedalam C -cluster berdasarkan kesamaannya dengan pusat cluster yang diukur melalui fungsi jarak. Fungsi tujuan J_m didefinisikan sebagai fungsi V (matriks sifat prototipe) dan U (fungsi keanggotaan Fuzzy dari objek $i: 1, 2, \dots, n$ menjadi anggota cluster $j: 1, 2, \dots, c$) yang diminimumkan melalui iterasi. Karena sensitivitas inisiasi random terjebak dalam minimum lokal, pendekatan genetik biasanya dapat diterapkan untuk menghilangkan masalah diatas.

Penelitian ini ditujukan untuk mempelajari unjuk kerja AG pada masalah fuzzy clustering, khususnya fuzzy C Mean (FCM). Guided GA (GGA) sebagai alternatif FCM diperkenalkan melalui kode MATLAB, kedua algoritma ini dibandingkan melalui tiga kasus numerik. Unjuk kerja FCM dan GGA-FCM diuji melalui beberapa parameter seperti kesahihan cluster, kompleksitas, laju klasifikasi komputasi. Hasilnya menunjukkan bahwa GGA-FCM mempunyai unjuk kerja yang lebih baik dibandingkan FCM dalam bentuk J_m , kesahihan cluster dan laju klasifikasi, meskipun dalam kompleksitas komputasi lebih rumit dan memerlukan waktu lebih banyak.

Kata kunci : fuzzy clustering, Algoritma Genetika, GGA-FCM, FCM.

ABSTRACT

Genetic Algorithm(GA) is a method for solving optimization problem that based on evolutionary theory in biology. This algorithm work with a population of candidate solutions named as chromosom that initially generated randomly from the area of the solution space of objective function. By using a mechanism of genetic operator i.e. cross-over and mutation the population is evolved controlled by fitness function that directed to konvergence condition. This algorithms can be applied in many area of function optimizations, one of those is applications on objective function based fuzzy clustering.

Fuzzy clustering aims to cluster n objects represented by feature vectors into c cluster based on similarity to the cluster center measured by a distance function. Objective function J_m was defined as function of V (matrix of feature prototype) and U (fuzzy

membership function of object $i:1,2,\dots,n$ become member of cluster $j:1,2,\dots,c$) to be minimized through iteration. Because of sensitivity to random initialization to be trapped in local minima, genetic approach generally can be applied to reduce this problem.

This research was aimed to study the performance of GA approach to the problem of fuzzy clustering, mainly fuzzy-c-mean (FCM). The Guided Genetic Algorithm as an alternative to FCM, called GGA-FCM was introduced to be coded in MATLAB script. The two algorithms was compared through the tests of those algorithms with three numerical cases. The performance of both FCM and GGA-FCM was examined by some parameters such as cluster validity, complexity of computation classification rate. The results showed that GGA-FCM has better performance than FCM in terms of J_m , cluster validity and classification rate, although in term of computational complexity is more complex and time consuming.

Keywords: fuzzy clustering, genetic algorithm, GGA-FCM, FCM

Makalah diterima tanggal 3 April 2007

1. PENDAHULUAN

Pengenalan pola (*pattern recognition*) merupakan konsep yang sangat luas aplikasinya dalam banyak bidang antara lain : biomedical (EEG,ECG,Röntgen, Nuclear, Tomography,Tissue,Cells,Chromosomes, meteorolgy (remote sensing), industrial inspection (robotic vision) dan digital microscopy. Beberapa aplikasi dalam bidang komputer dan informatika diantaranya : Speech recognition, speaker identification, character recognition, signature verification, image segmentation dan artificial intelligence.

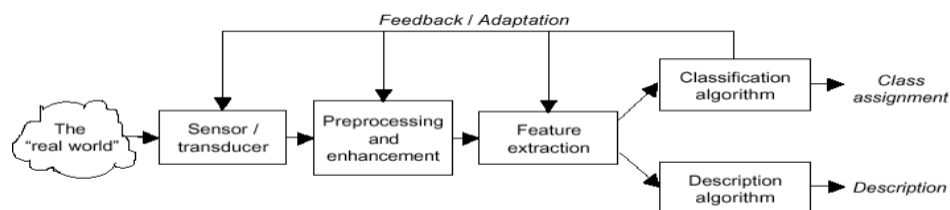
Belanche and Nebot (2002) menggambarkan pengenalan pola secara garis besar sebagai serangkaian kegiatan yang mencakup: kegiatan pengukuran dunia nyata dengan alat ukur yang menggambarkan fenomena yang akan diukur diikuti serangkaian kegiatan: preprosesor, ekstrak feature, klasifikasi atau diskripsi pola (lihat Gambar 1). Kegiatan yang vital dalam pengenalan pola adalah kegiatan klasifikasi dari ruang feature yang

diperoleh dari kegiatan seleksi dan ekstrak feature.

Metode yang lebih baik dari metode clustering secara tegas adalah aplikasi teori fuzzy dalam proses kluster (fuzzy clustering). Hamzah (2001) menunjukkan bahwa proses kluster secara fuzzy (fuzzy clustering) menunjukkan hasil yang lebih baik dan lebih alami dibandingkan dengan proses kluster dengan pendekatan tegas.

Sejak dikembangkan pertama kali oleh Holland tahun 1975, algoritma genetika (Genetic Algorithms=GA) terus mengalami perkembangan dalam banyak aplikasi (Gen and Cheng 2000). Pesatnya perkembangan aplikasi GA pada berbagai problem optimasi dipacu oleh perkembangan teknologi komputer dan mikro prosesor.

Salah satu minat dalam aplikasi genetic-fuzzy system adalah genetic-fuzzy clustering. Pada fuzzy clustering berbasis fungsi tujuan persoalan mencari kluster terbaik akan identik dengan persoalan optimasi fungsi tujuan. Penggunaan algoritma genetika untuk fuzzy clustering dimungkinkan dapat meningkatkan unjuk kerja fuzzy clustering. Penerapan GFS pada



Gambar 1. Skema Kegiatan dalam Pengenalan Pola(Belanche and Nebot, 2002)

fuzzy clustering terutama Fuzzy-C-Means Clustering (FCM) adalah untuk mengoptimasi parameter-parameter dalam FCM.

Pertanyaan yang muncul kemudian dalam penerapan GA pada FCM adalah sejauh mana peningkatan kinerja FCM dengan pendekatan GA tersebut. Parameter-parameter FCM apa yang kemudian harus dipertimbangkan dalam penerapan GA, latar belakang inilah yang mendasari dilakukannya penelitian ini.

Adapun tujuan penelitian ini adalah mengkaji penerapan algoritma genetika dalam fuzzy clustering, utamanya FCM, serta mengkaji sejauh mana algoritma genetika dapat meningkatkan unjuk kerja fuzzy clustering.

Fuzzy-C-Means clustering

Pada pendekatan tegas (crisp), untuk setiap objek ke- k ($k=1\dots n$) secara tegas hanya dapat menjadi anggota cluster ke- i ($i=1\dots c$), dengan keputusan menjadi anggota cluster ke- i berdasarkan jarak minimal objek ke- k dengan pusat-pusat cluster ke- i . Algoritma pendekatan tegas dengan jumlah cluster k Clustering dan pusat cluster ditentukan dengan cara rata-rata ini sering disebut sebagai K-Means.

Pada pendekatan fuzzy metode klustering berdasarkan kenyataan bahwa objek-objek tertentu mungkin secara tegas tidak dapat dikelompokkan pada cluster tertentu. Dengan pendekatan fuzzy setiap objek ke- k ($k=1,2,\dots,n$) dianggap menjadi anggota dari semua cluster ke- i ($i=1,2,\dots,c$) dengan fungsi keanggotaan antara 0 sampai 1. Keputusan objek ke- i menjadi anggota cluster ke- j berdasarkan fungsi keanggotaan yang terbesar. Model clustering seperti ini terkenal dengan sebutan Fuzzy-C-Means Clustering (FCM).

Banyak macam fungsi objektif telah dirumuskan, baik yang pendekatan tegas (crisp) maupun pendekatan fuzzy (Sato and Sato, 1995). Untuk menghasilkan formulasi yang presisi dalam menentukan kriteria klustering dapat ditempuh dengan metode fungsi objektif (objective-function methods), yaitu dengan mengukur kemampuan untuk

dilibatkan dalam cluster sebagai fungsi dari c (yaitu cacah cluster) dengan suatu fungsi objektif tertentu (Zimmerman, 1991). Fuzzy clustering FCM dengan fungsi tujuan menggunakan jarak euclidean mengasumsikan bentuk fungsi tujuan spherical. Untuk data tertentu kondisi spherical mungkin tidak terpenuhi.

Pemilihan fungsi tujuan dan kriteria jarak dengan demikian sangat tergantung pada sebaran data objek. Untuk fungsi tujuan spherical dimana FCM dengan jarak euclidean didefinisikan, kriteria cluster yang baik selanjutnya adalah cluster yang meminimalkan fungsi tujuan J_m yang dirumuskan sebagai fungsi dari U dan V sebagai berikut :

$$J_m(U, V) = \sum_{i=1}^c \sum_{k=1}^n m_{ik}^m D_{ik}^2 (v_i - x_k) \quad (1)$$

untuk: $i=1,2,\dots,c$ dan $k=1,2,\dots,n$.

Matrik $U \in M_{cn}$ adalah matrik partisi dengan ordo cn dengan elemen matrik μ_{ik} . Untuk klustering secara tegas $\mu_{ik} \in \{0,1\}$, yaitu bernilai 1 jika objek k anggota cluster i dan 0 jika tidak. Untuk klustering secara fuzzy $\mu_{ik} \in [0,1]$ yang bernilai real. Nilai parameter m merupakan derajat ke-fuzzi-an dari proses klustering dengan nilai $m \in [1, \infty)$. $V = [v_1, v_2, \dots, v_c]$ adalah matrik parameter prototipe (pusat cluster) $v_i \in \mathbb{R}^p$. $D_{ik}^2(v_i, x_k)$ adalah jarak dari vektor x_k ke pusat cluster v_i .

Struktur cluster terbaik yang dicari adalah penyelesaian minimum (minimizer) untuk persamaan (1) yang dapat diselesaikan dengan proses iterasi untuk penyelesaian optimasi fungsi. Pada penyesuaian iterasi mula-mula dibangkitkan vektor pusat cluster secara acak dan kemudian iterasi dilakukan untuk mengupdate matrik U . Iterasi dilakukan sampai matrik U relatif tidak berubah. Berikut adalah nilai U dan V berturut-turut yang digunakan dalam iterasi:

$$m_{ik} = \frac{\left[\frac{1}{|x_k - v_i|^2} \right]^{1/(m-1)}}{\sum_{j=1}^c \left[\frac{1}{|x_k - v_j|^2} \right]^m} \text{ untuk } i = 1,2,\dots, c; k = 1,2,\dots, n \quad (2)$$

Dengan nilai U yang telah diupdate, maka vektor pusat kluster diperbaharui dengan :

$$v_i = \frac{1}{\sum_{k=1}^n m_{ik}^m} \sum_{k=1}^n m_{ik}^m x_{ik} \text{ untuk } i = 1,2,\dots, c \quad (3)$$

Proses penyelesaian optimasi secara klasik sering menghantar pada penyelesaian yang merupakan optimum lokal dari himpunan penyelesaian yang mungkin. Pendekatan algoritma genetika dapat menjadi alternatif yang dapat diterapkan pada proses penyelesaian optimasi yang menghindari optimum lokal.

Algoritma Genetika

Algoritma genetika dikembangkan pertama kali oleh Holland tahun 1975 (Gen and Cheng,2000). Sejak dikembangkan sampai saat ini GA ini terus menjadi objek riset dalam berbagai aplikasi. Alasan mengapa GA banyak menjanjikan, antara lain banyak problem dibidang sains dan teknik tidak dapat dipecahkan dengan algoritma deterministik biasa meskipun dengan waktu yang meningkat secara polynomial.

Secara umum algoritma ini memiliki prosedur yang dapat dirumuskan sebagai berikut Procedure Genetic Algorithms

```

Begin
t ← 0
initialize P(t)
evaluate P(t)
while (not termination condition )
do
  begin
    recombine P(t) to yield C(t)
    evaluate C(t)
    select P(t+1) from P(t) and C(t)
    t ← t+1
  end
end

```

Dengan t = generasi, P(t)= Populasi pada generasi t , dan C(t)=Populasi tambahan atau individu baru (offspring) dari hasil proses operasi genetik Cross Over dan Mutasi.

Encoding.

Dalam algoritma genetika pengkodean (encoding) solusi problem kedalam suatu kromosom merupakan isu penting. Populasi awal yang berisi N kromosom dibangkitkan secara random yang menjangkau keseluruhan ruang solusi. Proses evolusi dilakukan dengan melakukan operasi genetik (cross-over dan mutasi) dan melakukan seleksi kromosom untuk generasi berikutnya sampai sejumlah generasi yang dikehendaki dengan panduan fungsi fitness.

Algoritma genetika pada dua ruang, yaitu ruang solusi (disebut phenotip) dan ruang coding (disebut genotip). Operasi genetika (cross over dan mutasi) dilakukan pada ruang genotip sementara operasi seleksi dilakukan pada ruang phenotip. Dalam binary encoding variabel keputusan diwakili oleh deretan bit 0,1 yang panjangnya disesuaikan dengan ruang pencarian. Tiap bit 0,1 dapat dianggap sebagai sebuah gen. Untuk optimasi 2 variabel misalnya, maka solusi adalah x_1 dan x_2 .

Integer dan literal permutation encoding adalah kode terbaik untuk problem combinatorial optimization karena inti dari problem ini adalah mencari kombinasi atau permutasi terbaik dari item solusi terhadap kendala yang ada. Untuk problem yang lebih kompleks encoding menggunakan data struktur yang lebih sesuai.

Fungsi fitness.

Fungsi fitness adalah fungsi yang digunakan untuk menjustifikasi apakah suatu kromosom layak bertahan. Pada setiap generasi dipilih kromosom yang mendekati solusi dengan mengevaluasi fungsi kecocokan dari kromosom tersebut. Fungsi ini didefinisikan sedemikian sehingga

semakin besar nilai fitness semakin besar probabilitas untuk terseleksi pada generasi berikutnya. Untuk maksimasi maka fungsi tujuan dapat dijadikan sebagai fungsi fitness, sehingga kromosom yang mewakili nilai fungsi besar akan memiliki probabilitas terseleksi yang besar juga. Untuk minimasi dapat dirumuskan sedemikian sehingga fungsi tujuan yang semakin kecil maka memiliki fungsi fitness yang besar.

Pada penyelesaian yang berada diluar daerah visible dapat digunakan suatu modifikasi fungsi fitness dengan menambahkan suatu fungsi yang sering disebut sebagai fungsi penalti, sehingga fungsi fitness menjadi :

$$\text{fitness}(x)=f(x)+p(x) \quad (4)$$

Seleksi.

Setiap anggota populasi diwakili deretan string (disebut kromosom) dengan panjang tertentu. Elemen string tersebut dapat berupa digit 0,1 (untuk binary encoding), bilangan real (untuk real encoding), atau elemen lain. Untuk ukuran populasi N yang biasanya dipertahankan tetap prosedur seleksi diperlukan untuk memilih anggota populasi yang mana yang akan tetap eksis pada generasi berikutnya. Fungsi fitness digunakan untuk menjustifikasi apakah kromosom layak dipertahankan atau tidak dalam generasi berikutnya.

Sebelum dilakukan seleksi jumlah anggota populasi ditambah dengan hasil offspring dari proses operasi genetik yang dapat berupa cross over dan mutasi. Hasil operasi genetik dan populasi semula selanjutnya diseleksi dengan metode tertentu untuk diambil n anggota populasi yang terbaik. Untuk kasus minimasi maka yang terpilih adalah n anggota populasi dengan nilai fitness yang terkecil.

Crossover

Proses operasi crossover dirancang untuk mencari kemungkinan yang lebih baik dari anggota populasi yang telah ada. Dari pasangan induk yang terpilih berdasarkan seleksi fungsi fitness diambil sejumlah pasangan dengan probabilitas P_c untuk dikenakan operasi crossover.

Mutasi.

Mutasi dalam konteks binary encoding adalah perubahan pada bit tunggal (bit 0 jadi 1 dan sebaliknya) anggota populasi yang terpilih. Banyaknya bit yang mengalami mutasi pada setiap generasi diatur oleh probabilitas mutasi (P_m) yang nilainya merupakan cacah bit mutasi dibagi cacah bit total dalam populasi.

Algoritma Genetika Untuk Fuzzy Clustering.

Algoritma genetika (GA) sebagai teknik optimasi dapat diterapkan pada clustering yang berbasis optimasi fungsi tujuan. Pada pendekatan GA untuk fuzzy-clustering fungsi fitness diambil dari fungsi objektif yang diminimumkan, yaitu $J_m(U,V)$.

Validitas Clustering dan Classification Rate.

Hasil akhir FCM atau GGA-FCM adalah V,U dan R_m tertentu untuk suatu c yang diinputkan. Pada beberapa kasus c yang tepat mungkin tidak diketahui. Untuk itu beberapa pendekatan telah diusulkan untuk menentukan berapa sebaiknya c sehingga hasil klustering dapat dianggap terbaik. Ukuran ini dikenal sebagai validitas klustering. Chi et.al. (1996) merangkum setidaknya ada empat alat ukur, yaitu :

1. Koefisien Partisi (partition coefficient)

Koefisien Partisi didefinisikan sebagai fungsi U dan c:

$$F(U, c) = \frac{1}{n} \sum_{i=1}^c \sum_{k=1}^n (u_{ik})^2 \quad (5)$$

Koefisien partisi ini mengukur kedekatan dari semua sampel masukan terhadap prototipe yang terpilih.

2. Entropy partisi (Partition Entropy)

$$H(U, c) = -\frac{1}{n} \sum_{i=1}^c \sum_{k=1}^n u_{ik} \log(u_{ik}) \quad (6)$$

Entropy partisi ($0 < H(u,c) < \log c$) diterjemahkan sebagai derajat ketidakpastian (degree of fuzzyness) dari objek. $H(U,c)$ besar yang menunjukkan hasil klustering yang jelek.

3. **Fungsi validitas Kekompakan dan separasi (compactness and separation validity function)**

$$S(U, c) = \frac{\frac{1}{n} \sum_{i=1}^c \sum_{k=1}^n u_{ik}^2 |x_k - v_i|^2}{\min_{i, k} |v_k - v_i|^2} \quad (7)$$

S(U,c) adalah rasio antara rata-rata jarak sampel dengan prototipe yang terpilih dengan minimum jarak antar prototipe.

Classification rate.

Keberhasilan clustering terkadang diukur juga melalui kemampuan clustering tersebut dalam klasifikasi objek. Hal ini dapat ditentukan apabila objek-objek yang diklasifikasi telah diketahui berasal dari suatu klas tertentu. Banyaknya objek yang dapat diklasifikasikan dengan tepat sesuai dengan klas dari mana objek tersebut berasal disebut sebagai classification rate. Index ini dapat dituliskan sebagai :

$$CR = \frac{n}{N} \times 100\% \quad (8)$$

Dimana : n=banyaknya objek yang dapat diklasifikasi dengan benar N=banyaknya objek yang diklasifikasi

Algoritma Fuzzy clustering konvensional (FCM).

Berikut diuraikan algoritma fuzzy clustering konvensional, yaitu penyelesaian fuzzy clustering dengan cara iteratif dengan melakukan update pada matrik keanggotaan U dan matrik prototipe kluster V. Dalam algoritma diperlukan : Sampel objek sebanyak n, tiap objek p parameter, dituliskan : $\mathbf{X} = \{x_1, x_2, \dots, x_n\}$ $x_i \in R^p$,

$i=1,2,\dots,n$. Ditentukan dalam proses penyelesaian melalui iterasi :

$\mathbf{U} = [\mu_{ik}]$ matrik ukuran $c \times n$; $i=1,2,\dots,c$; $k=1,2,\dots,n$; $\mathbf{V} = \{v_1, v_2, \dots, v_c\}$ $i=1,2,\dots,c$

Berikut adalah algoritma yang diajukan :

1. Initialization step :

Tentukan: n=cacah objek yang akan dikluster ; p=cacah parameter dalam tiap objek; c=cacah kluster; t=0 (iterasi ke); m=derajat fuzziness = dipilih 2;

ϵ =nilai yang cukup kecil mendekati

0

Tentukan secara acak : $\mathbf{U}^{(0)}$ dan $\mathbf{V}^{(0)}$

2. Iteration step :

a). Dengan menggunakan $\mathbf{U}^{(t)}$, hitung pusat kluster $\mathbf{V}^{(t)}$ menurut rumus :

$$v_i = \frac{1}{\sum_{k=1}^n m_{ik}^m} \sum_{k=1}^n m_{ik}^m x_{ik} \quad (9)$$

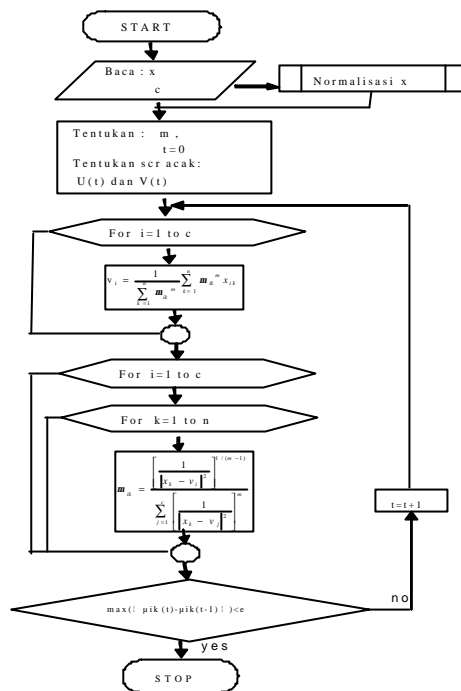
untuk $i = 1, 2, \dots, c$,

b). Dengan menggunakan $\mathbf{V}^{(t)}$, update membership $\mathbf{U}^{(t)}$, dengan rumus :

$$m_{ik} = \frac{\left[\frac{1}{|x_k - v_i|^2} \right]^{1/(m-1)}}{\sum_{j=1}^c \left[\frac{1}{|x_k - v_j|^2} \right]^m} \quad (10)$$

untuk $i = 1, 2, \dots, c$; $k = 1, 2, \dots, n$

c). Jika $\max(|\mu_{ik}^{(t)} - \mu_{ik}^{(t-1)}|) < \epsilon$ berhenti, Jika tidak ulangi langkah 2.a). Diagram alir untuk Fuzzy Clustering seperti tersaji dalam Gambar 3.



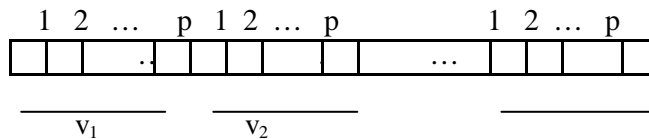
Gambar 3. Diagram Alir Fuzzy Clustering (FCM)

Algoritma Fuzzy Clustering dengan Pendekatan Algoritma genetika (GeneticGuidedAlgorithm for Fuzzy c-Means = GGA-FCM)

Pada pendekatan algoritma genetika untuk penyelesaian fuzzy clustering ditempuh pilihan untuk menggunakan pendekatan Prototype-based algorithms, yaitu meng-evolusikan matrik pusat cluster V. Beberapa hal yang ditentukan adalah :

Encoding dan Struktur kromosom :

Encoding yang digunakan adalah real encoding . Struktur kromosom untuk V dalam populasi yang dievolusikan adalah vektor real beranggotakan cxp elemen (c=cacah kluster dan p cacah elemen dalam objek), seperti dalam Gambar 4.



Gambar 4. Struktur Kromosom untuk Encoding V

Fungsi Fitness :

Fungsi fitness yang digunakan adalah objective function Jm, yaitu:

$$J_m(\mathbf{U}, \mathbf{V}) = \sum_{i=1}^c \sum_{k=1}^n m_{ik}^m D_{ik}^2 (v_i - x_k) \quad (11)$$

Dimana μ_{ik} dikalkulasi dengan :

$$m_{ik} = \frac{\left[\frac{1}{|x_k - v_i|^2} \right]^{1/(m-1)}}{\sum_{j=1}^c \left[\frac{1}{|x_k - v_j|^2} \right]^m} \quad (12)$$

untuk $i = 1, 2, \dots, c; k = 1, 2, \dots, n$

dan jarak D_{ik}^2 menggunakan jarak eucledian.

Operasi Genetik.

Cross-Over:

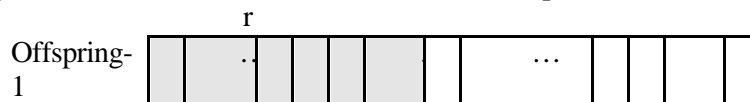
Operasi cross over dipilih metode one-point cross-over, yaitu untuk tiap kromosom induk ditetapkan 1 titik secara random antara 1 sampai cpx, misalkan disebut titik r.

Selanjutnya offspring didapatkan dengan mensilangkan (cross-over) nilai mulai posisi r sampai akhir kromosom. Untuk lebih jelasnya urutan cross over disajikan dalam Gambar 5.

Langkah 1. Pilih titik r : angka random antara 1 sampai cpx



Langkah 2. Tukarkan nilai vektor antara titik r sampai akhir



Gambar 5. One-point cross over untuk kromosom

Mutasi :

Metode mutasi yang dipilih adalah **Nonuniform mutation** dengan algoritma untuk mutasi dapat dituliskan sebagai berikut :

1. Pilih satu induk yang akan dimutasi berdasarkan fungsi fitness dari kromosom anggota populasi
2. Tentukan digit (posisi) gen yang akan dimutasi, dengan menentukan $k = \text{round}(\text{rand} * (n-1)) + 1$
3. Pilih gen yang dimutasi, yaitu : x_k
4. Tentukan batas bawah (x_k^L) dan batas atas (x_k^U)
5. Nilai offspring diperoleh (x_k') dipilih secara acak dari salah satu :
 $x_k' = x_k' + \Delta(t, x_k^U - x_k)$ atau

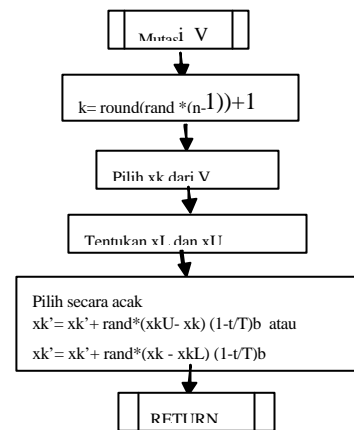
$$x_k' = x_k' + \Delta(t, x_k - x_k^L)$$

Dengan kisaran (range) nilai real untuk x_k adalah $[x_k^L, x_k^U]$.

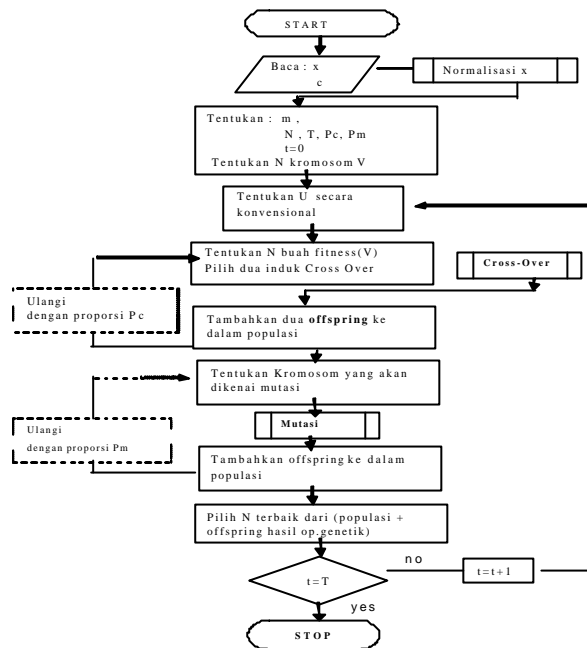
Nilai $\Delta(t, x_k - x_k^L)$ atau $\Delta(t, x_k^U - x_k)$ dari kriteria itu diberikan oleh :

$$\Delta(t, y) = yr(1-t/T)^b$$

dengan r = bilangan random kisaran [0,1] dan b adalah bilangan bulat.



Gambar 6. Diagram Alir Mutasi



Gambar 7. Diagram Alir Fuzzy Genetik Clustering (GGA-FCM)

2. PEMBAHASAN

Kasus yang dijadikan bahan pengujian algoritma-algoritma yang diteliti adalah

Kasus 1. Data dua dimensi dengan jumlah sampel 14 objek

Tabel 1. Data vektor atribut untuk 14 objek

Objek Ke-k	$X^{(k)}$	
	$X_1^{(k)}$	$X_2^{(k)}$
1	1	1
2	8	4
3	2	6
4	2	1
5	8	0
6	2	2
7	8	2
8	4	5
9	7	3
10	1	2
11	9	3
12	1	5
13	9	1
14	5	6

Kasus 2. Data matrik pixel dari citra peta berukuran 134 x 153 x 3 dengan derajat keabuan 0 sampai 255 sebagai Gambar 8 (Chi et.al., 1996). Dari kasus-kasus yang dicobakan dibandingkan hasil Fuzzy-Clustering konvensional dan Genetic Algorithm Fuzzy Clustering ditinjau dari beberapa aspek antara lain : waktu komputasi, Classification rate, Cluster Validity.

Pengujian kasus 1

Secara visual data kasus 1 (14 objek dua dimensi) dapat digambarkan seperti Gambar 9.

Dari Tabel 2 terlihat bahwa semakin besar nilai C maka nilai J_m (untuk $m=2$ ditulis sebagai J_2) semakin kecil. Apabila nilai C sama dengan n maka nilai J_2 akan sama dengan 0. Untuk nilai c tertentu, kluster yang baik adalah yang memberikan J_2 terkecil dari trial yang dilakukan berulang. Apabila nilai c yang dicobakan bermacam-macam dan akan dicari nilai c

terbaik maka validitas kluster FU, HU dan SU dapat dijadikan pedoman. Nilai c yang baik ditinjau dari derajat ke-fuzzy-an apabila FU cukup besar dan HU serta SU cukup kecil. Dari tabel 2 terlihat bahwa untuk ukuran kluster c=3 dan c=5 merupakan kluster yang baik. Pengujian data kasus 1

dengan algoritma genetika ditetapkan beberapa parameter algoritma genetika, yaitu :

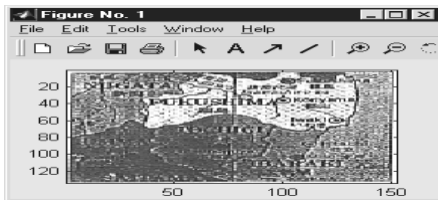
Jumlah populasi (nP) = 30; Jumlah generasi (nG) = 500; Probabilitas CrossOver (Pc) = 0.2; Probabilitas Mutasi (Pm) = 0.1; Nilai m = 2; Trial = 100 (ulangan runs)

Tabel 2. Hasil Kluster dengan FCM untuk m=2

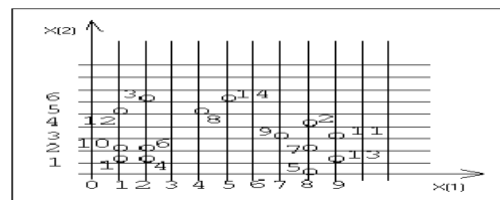
C	Iter	J2	FU	HU	SU
2	26	1.2359	0.8325	0.2899	0.1006
3	37	0.4759	0.8196	0.3624	0.0877
4	63	0.2729	0.7524	0.4967	0.1475
5	58	0.2248	0.7031	0.6142	0.0829
6	39	0.1270	0.6698	0.6474	0.1874

Pengujian algoritma untuk cacah cluster c=3 dan c=4 hasil pelacakan terbaik GGA-FCM dapat disajikan seperti dalam Gambar 10. Garis-garis dalam grafik ada peta pelacakan dari titik random awal inialisasi menuju titik akhir (konvergensi) kluster optimal.

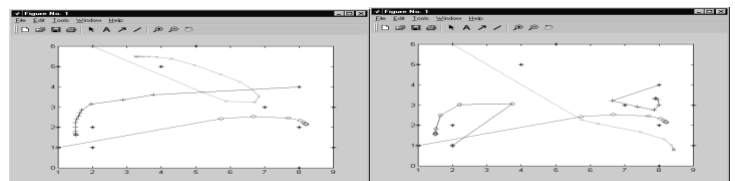
Perbandingan secara visual untuk metode FCM dan GGA-FCM menampakkan hasil yang hampir sama. Untuk lebih detail perbandingan dapat diteliti melalui nilai optimal Jm yang diperoleh dan parameter-parameter clustering melalui validitas cluster. Untuk itu pengujian untuk nilai-nilai c yang lebih lengkap, yaitu c=2,3,4,5 dan 6 hasilnya adalah seperti tabel 3 berikut.



Gambar 8. Citra Peta (Chi *et.al.*, 1996)



Gambar 9. Plot Data 14 objek Pengujian dengan metode iteratif FCM untuk cacah kluster c=2, c=3, c=4, c=5 dan c=6.



Gambar 10. Peta pelacakan cluster untuk c=3 dan c=4

Tabel 3 Hasil Kluster dengan GGA-FCM untuk m=2

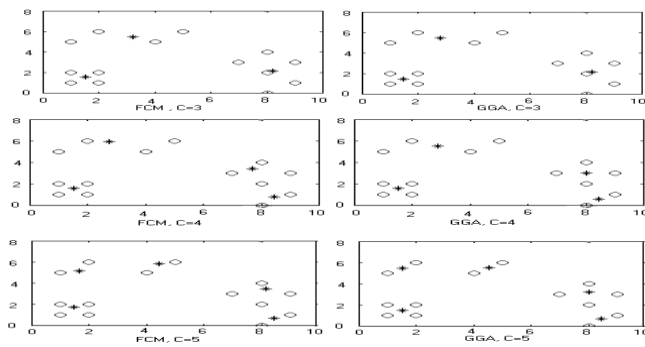
C	Rerata J2	FU	HU	SU
2	0.9543	0.8712	0.2091	0.0996
3	0.4143	0.8262	0.3122	0.0711
4	0.2511	0.7629	0.4227	0.1286
5	0.1126	0.7138	0.5932	0.0729
6	0.0932	0.6988	0.6025	0.1123

Jika hasil kluster GGA-FCM dalam tabel 4 dibandingkan dengan hasil FCM dalam Tabel 3 terlihat bahwa nilai-nilai J2 sebagai fungsi tujuan yang diminimalkan pada GGA-FCM memiliki nilai yang lebih kecil, untuk semua C. Ini menunjukkan bahwa secara rerata pusat-pusat kluster yang ditemukan oleh GGA-FCM lebih baik dari FCM.

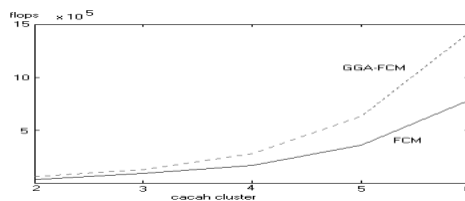
Beban Komputasi Untuk Data Kasus 1

Beban komputasi untuk data kasus 1 ditentukan berdasarkan cacah operasi

floating point (flops) dari algoritma tersebut. Secara eksak beban komputasi tidak dapat ditentukan disebabkan faktor random yang muncul dalam pembangkitan data awal, sehingga untuk jumlah kluster yang sama dapat terjadi cacah flops yang berbeda cukup signifikan. Pendekatan yang dilakukan untuk menentukan beban komputasi dilakukan dengan mengulang running program untuk cacah kluster yang sama dan menentukan rerata beban komputasi dan standard deviasi beban komputasi .



Gambar 11. Posisi pusat kluster (*) terhadap data (o) untuk data kasus 1, dengan C=3,4 dan 5 pada metode FCM dan GGA-FCM



Gambar 12. Hubungan Cacah cluster dengan flops untuk FCM dan GGA-FCM

Tabel 4. Rerata dan standard deviasi beban komputasi clustering FCM

Cacah cluster (C)	Run	FCM		GGA-FCM	
		Rerata flops	StdDev flops	Rerata flops	StdDev flops
2	10	30.075	2.802	60.125	5.324
3	10	88.972	5.430	124.567	8.247
4	20	168.260	58.040	276.545	76.235
5	20	356.230	240.870	632.421	405.754
6	20	788.000	323.860	1.432.402	511.176

Dari Tabel 4. dan Gambar 12. terlihat bahwa cacah flops meningkat signifikan dan cenderung eksponensial dengan bertambahnya cacah cluster. Dari grafik juga terlihat bahwa GGA-FCM memiliki cacah flops yang lebih besar dan bahkan hampir dua kali cacah flops untuk algoritma FCM.

Classification Rate untuk Data Kasus 1

Untuk membandingkan classification rate data kasus 1 dengan pendekatan dua algoritma FCM dan GGA-FCM dilakukan proses klasifikasi dari cluster yang diperoleh untuk beberapa pilihan c. Hasilnya dapat disajikan dalam Tabel 5.

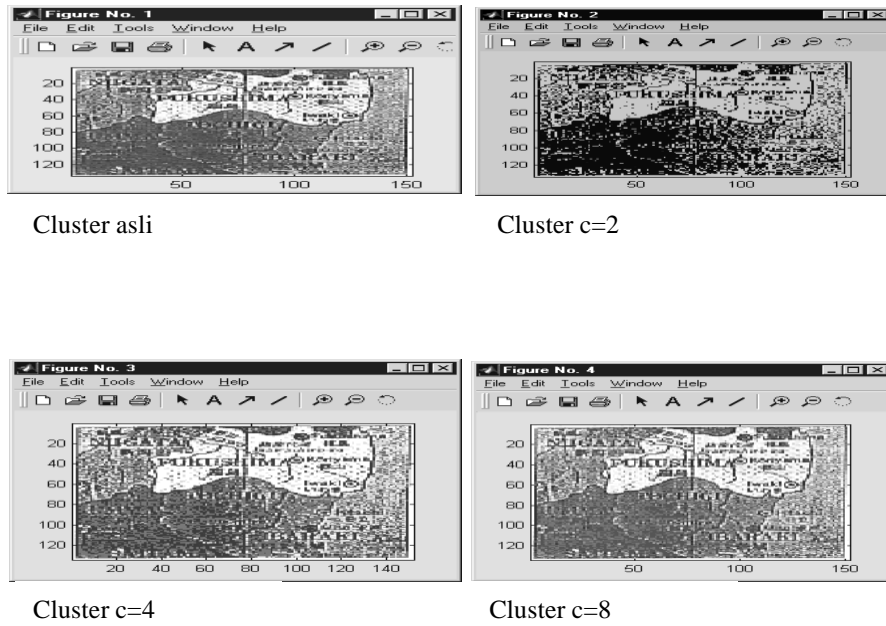
Tabel 5. Classification rate clustering FCM dan GGA-FCM

Cacah cluster (C)	FCM	GGA-FCM
2	95.2 %	95.4 %
3	96.1 %	96.7 %
4	98.3 %	99.4 %
5	99.2 %	99.5 %
6	98.6 %	98.4 %

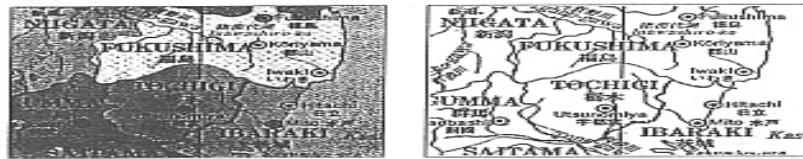
Dari tabel 5. terlihat bahwa kedua metode tidak jauh berbeda dalam menghasilkan cluster untuk proses klasifikasi objek yang ditunjukkan oleh nilai classification rate. Nilai terbaik klasifikasi diperoleh untuk cacah cluster c=5 untuk algoritma GGA-FCM.

Pengujian dengan Kasus 2

Aplikasi clustering dalam pengolahan citra dapat ditunjukkan dalam upaya mengubah citra aras keabuan (grayscale image) menjadi citra hitam-putih (black white image). Dalam citra asli ada derajat keabuan yang diwakili oleh nilai keabuan yang menyusun setiap pixel citra. Untuk sebuah peta setiap pixel penyusun peta dapat dikelompokkan sebagai pixel background (warna putih) atau foreground (warna hitam sebagai bentuk peta yang dapat berupa jalan, atau keterangan peta). Data citra berikut diambil dari Chi.et.all (1996) yang merupakan citra ukuran 134 x 153 x 3 pixel dengan setiap pixelnya 8bit skala keabuan. Hasil cluster untuk beberapa nilai c=2, c=4 dan c=8 dan digambarkan hasilnya seperti dalam Gambar 13.



Gambar 13. Clustering GGA-FCM untuk c=2, c=4 dan c=8



Gambar 14. Citra Peta dan Hasil Penajaman Citra dengan Rule-Base dengan 34-Fuzzy Rule

Dari hasil cluster untuk berbagai ukuran cluster terlihat bahwa dari sisi perbaikan citra, maka hasil yang didapatkan masih kurang memuaskan. Salah satu manfaat yang dapat diperoleh dari hasil clustering adalah variasi skala keabuan yang jauh lebih sederhana dengan pengklusteran (atau lebih dikenal dengan istilah segmentasi citra) yang dapat dimanfaatkan misalnya untuk kompresi citra. Untuk Citra peta tersebut penurunan ukuran file dapat didekati dengan hasil cluster.

Untuk pengolahan citra yang bertujuan penajaman citra, metode clustering dengan FCM dan GGA-FCM belumlah dapat memberikan hasil yang baik. Pendekatan rule-base fuzzy clustering dalam hal ini dapat merupakan alternatif yang lebih baik dalam penajaman citra. Gambar 14 adalah

contoh hasil penajaman citra dengan rule-base fuzzy clustering dari Chi et.all. (1996).

Tabel 6. Classification rate clustering FCM dan GGA-FCM

Tipe	Cacah pixel	Cacah Bit per pixel	Ukuran file
Citra asli	134 x 153	8 bit	100 %
Cluster c=2	134 x 153	1 bit	12.5 %
Cluster c=4	134 x 153	2 bit	25.0 %
Cluster c=8	134 x 153	3 bit	37.5 %

3. KESIMPULAN

Beberapa kesimpulan yang dapat diperoleh dari penelitian ini antara lain dapat diuraikan sebagai berikut :

1. Metode klustering dengan algoritma genetika (GGA-FCM) dapat memperbaiki kinerja FCM dengan metode iterasi konvensional terutama ditinjau dari fungsi tujuan J_m dan koefisien Classification Rate.
2. Jika ditinjau dari beban komputasi dan waktu komputasi metode GGA-FCM masih relatif lebih kompleks dan memerlukan cacah flops yang jauh lebih besar dari metode FCM konvensional. Namun dengan perkembangan prosesor yang semakin baik kemungkinan kompleksitas komputasi dan waktu komputasi dapat diatasi.
3. Algoritma genetika (GGA-FCM), meskipun membutuhkan waktu komputasi lebih banyak tetapi dapat memberikan jawaban yang lebih baik dari metode metode iteratif konvensional (FCM).
4. Untuk aplikasi pengolahan citra yang berupa penajaman citra algoritma GGA-FCM belum dapat digunakan, tetapi untuk segmentasi citra FCM maupun GGA-FCM dapat diterapkan dan dapat menghasilkan citra dengan komposisi bit yang lebih sederhana dan ukuran file yang lebih kecil.

DAFTAR PUSTAKA

- Belanche L. and Nebot A., 2002, *Intelligent Data Analysis and Data Mining*, Wright State University.
- Chi, Z., Yan, H. and Pham, T., 1996, *Fuzzy Algorithms : With Application to Image Processing and Pattern Recognition, Advance in Fuzzy System-Application and Theory*, Vol.10., Word Scientific, Singapore.
- Gen, M. and Cheng, R., 2000, *Genetic Algorithms and Engineering Optimization*, John Wiley & Sons, New York.
- Hamzah, A, 2001, *Pengenalan Pola dengan Fuzzy Clustering*, ACADEMIA ISTA, Vol.4.No.1., Lembaga Penelitian, Institut Sains dan Teknologi AKPRIND, Yogyakarta.
- Sato, M. and Sato Y., 1995, *Fuzzy Clustering with Multiple Objectives Functions*, Kluwer Academic Publisher, Boston.
- Zimmerman, H.J., 1991, *Fuzzy Set Theory and Its Applications*, 2nd Ed., Norwel, Massachusetts.