

BAB I

PENDAHULUAN

1.1. Latar Belakang

Citra atau gambar yang terlihat pada layar computer, sebenarnya adalah kumpulan sejumlah titik-titik warna. Pada umumnya citra berwarna direpresentasikan dengan 3 elemen warna yaitu R (Red=merah), G (green=hijau), dan B (blue=biru) yang masing-masing memiliki nilai tertentu. Dengan pencampuran 3 warna ini maka didapatkan warna tertentu. Pada citra RGB 24 bit, masing-masing elemen warna (RGB) memiliki nilai maksimal sebesar 2^8 , yaitu berkisar antara 0 - 255. Dari sebuah citra bisa didapatkan informasi sesuai dengan kepentingan, namun terkadang pada citra yang ada tidak bisa didapatkan secara langsung informasi yang diperlukan. Ada kalanya sulit untuk mengolah informasi dari sebuah citra secara langsung secara kasat mata dengan hanya mengandalkan indera penglihatan. Hal ini bisa dimaklumi, mengingat citra adalah kumpulan titik-titik warna yang jumlahnya banyak. Salah satu alternatif untuk membantu menampilkan dan mengolah informasi ini adalah dengan segmentasi citra.

Dengan cara atau metode segmentasi, sebuah citra yang semulanya sulit diolah dengan indera penglihatan secara langsung, kemudian setelah disegmentasi bisa mempermudah mengolah informasi yang terkandung didalamnya. Sebagai contoh, pada bidang biologi, sebuah citra atau gambar sel hasil pembesaran dengan mikroskop kemudian pada penginderaan jauh (remote sensing) kita bisa mendapatkan informasi dari sumber-sumber alam. Kemudian pada segmentasi terdapat berbagai macam metode yang digunakan diantaranya adalah metode iterasi, K-means, Fuzzy C-means, jaringan syaraf kohonen, dan berbagai teknik cluster lainnya. Pada skripsi ini menggunakan metode Fuzzy C-means karena metode ini sangat baik digunakan untuk Segmentasi Citra (Darma Putra, 2010).

1.2. Rumusan Masalah

Berdasarkan latar belakang yang dipaparkan di atas, maka dalam penelitian ini dapat diambil rumusan masalah sebagai berikut bagaimana menerapkan metode Fuzzy C-means dalam kasus pengelompokkan masing-masing jumlah kluster terhadap kesamaan sifat dari besarnya nilai intensitas cahaya yang dihasilkan pada citra.

1.3. Batasan Masalah

Agar ruang lingkup penelitian tidak terlalu luas, maka diperlukan suatu batasan-batasan pada permasalahan yang diangkat. Adapun batasan-batasan yang terdapat pada penelitian yang diusulkan ini adalah sebagai berikut :

1. Ukuran citra yang diinputkan ialah memiliki lebar dan panjang lebih kecil sama dengan 500 x 500 pixel.
2. Citra masukan system adalah citra Bitmap(.bmp)
3. Proses yang dilakukan aplikasi adalah proses segmentasi pada citra masukan.
4. menggunakan citra uji berupa citra penginderaan jauh, citra biomedis, dan citra pengenalan pola.
5. Perangkat lunak yang digunakan untuk merancang aplikasi adalah visual studio 2012.

1.4. Tujuan Penelitian

Tujuan dari penelitian ini adalah

1. Menerapkan fuzzy C-means ke dalam system segmentasi citra digital.
2. Merancang aplikasi untuk segmentasi citra dengan metode fuzzy C-means.
3. Dapat menampilkan proses segmentasi berdasarkan kesamaan sifat dari data-data tersebut

1.5. Manfaat Penelitian

Adapun manfaat yang diharapkan penulis dalam penelitian ini adalah sebagai berikut:

a. Bagi Jurusan Teknik Elektro Universitas Mataram

1. Menjadi perangkat pembelajaran dalam matakuliah pemrosesan (pengolahan) citra digital.
2. Menunjukkan kepada para mahasiswa bagaimana memanfaatkan fungsi-fungsi pustaka C#.NET untuk aplikasi segmentasi citra digital, selain MATLAB.

b. Untuk Bidang-Bidang Aplikasi Lain

1. Fotografi: Para astronom dapat melakukan pengukuran terhadap posisi dan jarak suatu bintang dari foto udara.
2. Biomedis: Membantu Para dokter untuk mengidentifikasi penyakit paru-paru, hati, tulang, segmentasi tulang dari otot yang lainnya, klasifikasi gigi, dan analisis citra mikroskopis.
3. Penginderaan-Jauh (remote sensing): mendapatkan informasi penting dari sumber-sumber alam seperti pertanian, perairan, kelautan, mineral, hutan dan geologi.

1.6.Sistematika Penulisan

Untuk memberikan gambaran singkat mengenai isi laporan tugas akhir secara keseluruhan, maka akan diuraikan beberapa tahapan dari penulisan laporan secara sistematis, yaitu :

BAB I

PENDAHULUAN

Dalam bab ini menguraikan latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA

Dalam bab ini menguraikan teori dari setiap pengklasteran yang diusulkan.

BAB III METODE PENELITIAN

Dalam bab ini menguraikan tahap – tahap pengembangan sistem perangkat lunak yang akan dirancang. Tahap ini juga mencakup perancangan antarmuka yang diusulkan.

BAB IV HASIL DAN PEMBAHASAN

Dalam bab ini membahas tentang hasil yang didapatkan pada pengklasteran fuzzy untuk segmentasi citra digital menggunakan pemrograman C#.NET.

BAB V SIMPULAN DAN SARAN

Dalam bab ini berisi kesimpulan dari apa yang telah diuraikan pada bab sebelumnya beserta saran – saran untuk pengembangan aplikasi selanjutnya.

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

Keh-Shih Chuang dkk, (2006) melakukan penelitian algoritma fuzzy C-means dengan menambahkan spatial function untuk menangani masalah segmentasi citra yang mengandung noise.

Soesanti, dkk (2010) melakukan penelitian yang bertujuan untuk menganalisis komputasi pada segmentasi citra medis adaptif berbasis logika fuzzy teroptimasi.

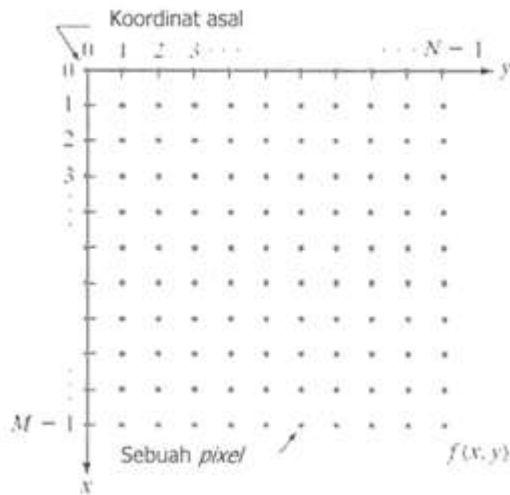
Sofyan Hizir, dkk (2010) pada penelitian ini melakukan pengelompokan piksel pada data citra satelit.

Faisal (2009) melakukan penelitian yang bertujuan untuk menerapkan metode clustering dengan algoritma fuzzy C-means dalam pengelompokan mahasiswa berdasarkan transkrip nilai mata kuliah.

2.2. Pengertian Citra Digital

Secara umum, pengolahan citra digital menunjuk pada pemrosesan gambar 2 dimensi menggunakan computer. Dalam konteks yang lebih luas, pengolahan citra digital mengacu pada pemrosesan setiap data 2 dimensi. Citra digital merupakan sebuah larik (array) yang berisi nilai-nilai real maupun kompleks yang direpresentasikan dengan deretan bit tertentu.

Suatu citra dapat didefinisikan sebagai fungsi $f(x,y)$ berukuran M baris dan N kolom, dengan x dan y adalah koordinat spasial, dan amplitudo f dititik koordinat (x,y) dinamakan intensitas atau tingkat keabuan dari citra pada titik tersebut. Apabila nilai x, y, dan nilai amplitudo f secara keseluruhan berhingga (finite) dan bernilai diskrit maka dapat dikatakan bahwa citra tersebut adalah citra digital. Gambar 2.1 menunjukkan posisi koordinat citra digital.



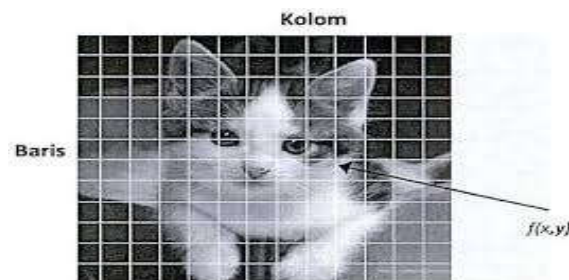
Gambar 2.1 Koordinat citra digital

Citra digital dapat ditulis dalam bentuk matrik sebagai berikut.

$$f(x,y) = \begin{bmatrix} f(0,0) & f(0,1) & \dots & f(0,N-1) \\ f(1,0) & f(1,1) & \dots & f(1,N-1) \\ \vdots & \vdots & \ddots & \vdots \\ f(M-1,0) & f(M-1,1) & \dots & f(M-1,N-1) \end{bmatrix}$$

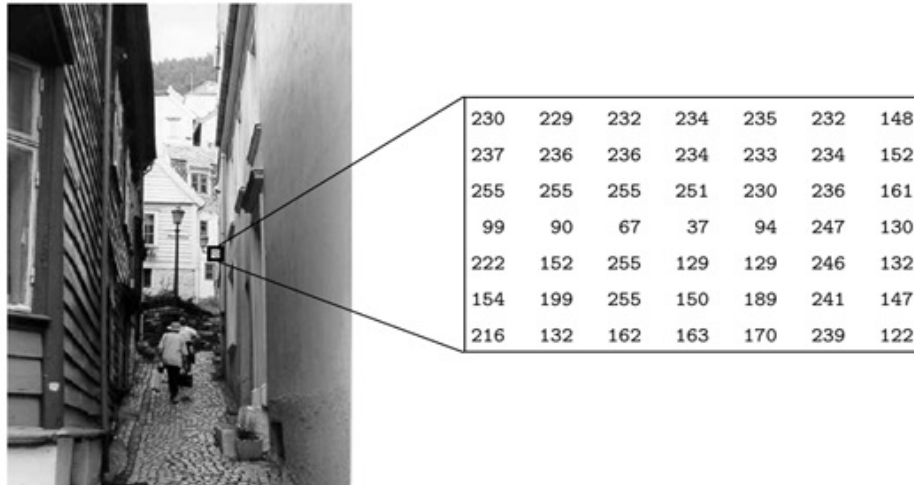
Gambar 2.2 Representasi citra digital

Nilai pada suatu irisan antara baris dan kolom (pada posisi x,y) disebut dengan picture elements, image element, pels, atau pixels. Istilah terakhir (pixel) paling sering digunakan pada citra digital. Gambar 2.3 menunjukkan ilustrasi digitalisasi citra dengan M= 16 baris dan N= 16 kolom.



Gambar 2.3 ilustrasi digitalisasi citra (pixel pada koordinat x=10, y=3 memiliki nilai 110)

Gambar 2.4 menyajikan contoh lain dari suatu citra digital (citra grayscale), dengan nilai intensitas dari citra pada area tertentu.



Gambar 2.4 contoh citra grayscale pada area tertentu beserta nilai intensitasnya.

2.3 Jenis Citra

Nilai suatu pixel memiliki nilai dalam rentang tertentu, dari nilai minimum sampai nilai maksimum. Jangkauan yang digunakan berbeda-beda tergantung dari jenis warnanya. Namun secara umum jangkauannya adalah 0-255. Citra dengan penggambaran seperti ini digolongkan kedalam citra integer. Berikut adalah jenis-jenis citra berdasarkan nilai pixelnya.

2.3.1 Citra Biner

Citra biner adalah citra digital yang hanya memiliki dua kemungkinan nilai pixel yaitu hitam dan putih. Citra biner juga disebut sebagai citra black and white atau citra monokrom. Hanya dibutuhkan 1 bit untuk mewakili nilai setiap pixel dari citra biner.

Citra biner sering kali muncul sebagai hasil dari proses pengolahan seperti segmentasi, pengambangan, morfologi, ataupun dithering.



Gambar 2.5 citra biner

2.3.2 Citra Grayscale

Citra grayscale merupakan citra digital yang hanya memiliki satu nilai kanal pada setiap pixelnya, dengan kata lain nilai bagian RED= GREEN = BLUE. Nilai tersebut digunakan untuk menunjukkan tingkat intensitas. Warna yang dimiliki adalah warna dari hitam, keabuan, dan putih. Tingkatan keabuan disini merupakan warna abu dengan berbagai tingkatan dari hitam hingga mendekati putih. Citra grayscale berikut memiliki kedalaman warna 8 bit (256 kombinasi warna keabuan).



Gambar 2.6 Citra Grayscale

2.4 Elemen Dasar suatu citra

Elemen – elemen Citra yang dimaksud adalah sebagai berikut :

1.Pixel (picture element)

Gambar yang bertipe bitmap tersusun dari pixel –pixel, pixel disebut juga dengan dot. Berbentuk bujur sangkar dengan ukuran relatif kecil yang merupakan penyusun atau pembentuk gambar bitmap.

Banyaknya pixel tiap satuan luas tergantung pada resolusi yang digunakan keanekaragaman warna pixel tergantung pada *bit depth* yang dipakai. Semakin banyak jumlah pixel tiap satuan luas, semakin baik kualitas gambar yang dihasilkan dan tentu semakin besar ukuran filenya.

2.Bit depth

Bit depth (kedalaman warna) /*pixel depth/color depth*. *Bit depth* menentukan berapa banyak informasi warna yang tersedia untuk ditampilkan / dicetak dalam setiap pixel, semakin besar nilainya semakin bagus kualitas gambar yang dihasilkan , tentu ukuranya juga semakin besar

3.Resolusi

Resolusi adalah jumlah pixel persatuan luas yang ada disuatu gambar. Satuan pixel sering dipakai adalah dpi (*dot per inch*)/ppi (*pixel per inch*). Satuan dpi menentukan jumlah pixel yang ada setiap satu satuan luas. Yang dalam hal ini adalah satu inch kuadrat. Resolusi sangat berpengaruh pada detil dan perhitungan gambar.

4.Kecerahan dan Kontras

Yang dimaksud dengan kecerahan (*brightness*) adalah intensitas yang terjadi pada satu titik citra. Dan lazimnya pada sebuah citra , kecerahan ini merupakan kecerahan rata – rata dari suatu daerah lokal. Sistem visual manusia mampu menyesuaikan dirinya dengan “tingkat kecerahan“ (*brightness level*) dengan jangkauan (dari yang terendah sampai tertinggi)

Batas penyesuaian gelap (terendah) disebut dengan *scotopic threshold*, sedangkan batas penyesuaian terang (tertinggi) disebut dengan *glare threshold* sebagai contoh *scotopic threshold* terasa pada mata kita setelah lampu kita padamkan (terang ke gelap). Sementara *glare threshold* kita rasakan setelah kita keluar dari gedung bioskop (gelap ke terang)

Untuk menentukan kepekaan kontras (*contrast sensitivity*) pada mata manusia, dilakukan cara pengukuran sebagai berikut. Pada suatu bidang gambar dengan intensitas B , kita perbesar intensitas objek lingkaran sehingga intensitasnya menjadi $B + \Delta B$. Pertambahan intensitas (ΔB) ini dilakukan sampai mata kita dapat mendeteksi perbedaan ini. Dengan demikian kepekaan kontras dinyatakan dalam rasio weber sebagai $\Delta B / B$. dari hasil percobaan, diperoleh bahwa mata manusia memiliki rasio weber 2 % untuk bermacam-macam nilai B .

a. Acuity

Yang dimaksud *acuity* disini adalah kemampuan mata manusia untuk merinci secara detail bagian – bagian pada suatu citra (pada sumbu visual)

b. kontur

yang dimaksud dengan kontur (*countour*) adalah suatu keadaan pada citra dimana terjadi perubahan intensitas dari suatu titik ke titik tetangganya. Dengan perubahan intensitas inilah mata manusia sanggup mendeteksi pinggiran atau kontur suatu benda.

c. Warna

Warna (*color*) adalah reaksi yang dirasakan oleh sistem visual mata manusia terhadap perubahan panjang gelombang cahaya. Setiap warna memiliki panjang gelombang sendiri – sendiri. Warna merah memiliki panjang gelombang (λ) yang paling tinggi, sedangkan warna violet mempunyai panjang gelombang paling rendah.

d. Bentuk

Pada umumnya citra yang dibentuk oleh mata merupakan citra 2 dimensi, sedangkan objek yang diamati biasanya 3 dimensi. Kesulitannya banyak benda 3 dimensi setelah diproyeksikan ke bidang 2 dimensi kelihatan sama.

e. Tekstur

Pada hakekatnya sistem visual manusia tidak menerima informasi citra terpisah pada setiap titik, tetapi suatu citra dianggapnya sebagai satu kesatuan. Jadi definisi kesamaan objek perlu dinyatakan dalam bentuk kesamaan dari satu himpunan parameter citra (*brightness, color, size, dll*) atau dengan kata lain dua buah citra tidak dapat disamakan hanya dengan satu parameter saja.

f. Waktu dan pergerakan

Respon suatu sistem visual tidak hanya berlaku pada faktor ruang, tetapi juga pada faktor waktu. Sebagai contoh bila gambar diam ditampilkan bergantian secara cepat, maka kita akan mendapatkan kesan melihat gambar yang bergerak. Contoh gambar untuk film kartun.

g. Deteksi dan pengenalan

Dalam deteksi serta mengenali suatu citra, sering tidak hanya sistem visual kita yang bekerja, tetapi juga seluruh ingatan yang kita miliki. Contoh, apabila kita melihat seseorang wanita dari samping, kita mungkin menafsirkannya sebagai wanita muda tapi setelah melihat muka sebenarnya kita terkecoh dan sebaliknya.

2.5 Logika Fuzzy

2.5.1 Pengertian Logika Fuzzy

Fuzzy secara bahasa diartikan sebagai kabur atau samar-samar. Suatu nilai dapat bernilai besar atau salah secara bersamaan. Dalam fuzzy dikenal derajat keanggotaan yang memiliki rentang nilai 0 (nol) hingga 1 (satu). Berbeda dengan himpunan tegas yang memiliki nilai 1 atau 0 (ya atau tidak).

Logika Fuzzy merupakan suatu *logika* yang memiliki nilai kekaburan atau kesamaran (*fuzzyness*) antara benar atau salah. Dalam teori *logika fuzzy* suatu nilai bias bernilai benar atau salah secara bersama. Namun berapa besar keberadaan dan kesalahan suatu tergantung pada bobot keanggotaan yang dimilikinya. *Logika fuzzy* memiliki derajat keanggotaan dalam rentang 0 hingga 1. Berbeda dengan logika digital yang hanya memiliki dua nilai 1 atau 0. *Logika fuzzy* digunakan untuk menterjemahkan suatu besaran yang diekspresikan menggunakan bahasa (*linguistic*), misalkan besaran kecepatan laju kendaraan yang diekspresikan dengan pelan, agak cepat, cepat, dan sangat cepat. Dan *logika fuzzy* menunjukkan sejauh mana suatu nilai itu benar dan sejauh mana suatu nilai itu salah. Tidak seperti *logika klasik* (crisp)/ tegas, suatu nilai hanya mempunyai 2 kemungkinan yaitu merupakan suatu anggota himpunan atau tidak. Derajat keanggotaan 0 (nol) artinya nilai bukan merupakan anggota himpunan dan 1 (satu) berarti nilai tersebut adalah anggota himpunan.

Logika fuzzy adalah suatu cara yang tepat untuk memetakan suatu ruang *input* kedalam suatu ruang *output*, mempunyai nilai kontinyu. *Fuzzy* dinyatakan dalam derajat dari suatu keanggotaan dan derajat dari kebenaran. Oleh sebab itu sesuatu dapat dikatakan sebagian benar dan sebagian salah pada waktu yang sama (Kusumadewi. 2004)

Logika Fuzzy memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk *linguistik*, konsep tidak pasti seperti “sedikit”, “lumayan”, dan “sangat” (zadeh 1965)

Kelebihan dari teori logika *fuzzy* adalah kemampuan dalam proses penalaran secara bahasa (*linguistic reasoning*). Sehingga dalam perancangannya tidak memerlukan persamaan matematik dari objek yang akan dikendalikan.

2.5.2 Fuzzy Clustering

Fuzzy clustering adalah salah satu teknik untuk menentukan *cluster* optimal dalam suatu ruang *vektor* yang didasarkan pada bentuk normal *euclidian* untuk jarak antar *vektor*. *Fuzzy clustering* sangat berguna bagi pemodelan *fuzzy* terutama dalam mengidentifikasi aturan-aturan *fuzzy*. Metode *clustering* merupakan pengelompokan data beserta parameternya dalam kelompok – kelompok sesuai kecenderungan sifat dari masing-masing data tersebut (kesamaan sifat).

Ada beberapa algoritma *clustering* data, salah satu diantaranya adalah *Fuzzy C-Means*. *Fuzzy C-Means* adalah suatu teknik peng-*cluster*-an yang mana keberadaan tiap-tiap titik data dalam suatu *cluster* ditentukan oleh derajat keanggotaan. Teknik ini pertama kali diperkenalkan oleh Jim Bezdek pada tahun 1981.

Konsep dari *Fuzzy C-Means* pertama kali adalah menentukan pusat *cluster*, yang akan menandai lokasi rata-rata untuk tiap-tiap *cluster*. Pada kondisi awal, pusat *cluster* ini masih belum akurat. Tiap-tiap titik data memiliki derajat keanggotaan untuk tiap-tiap *cluster*. Dengan cara memperbaiki pusat *cluster* dan derajat keanggotaan tiap-tiap titik data secara berulang, maka akan dapat dilihat bahwa pusat *cluster* akan bergerak menuju lokasi yang tepat. Perulangan ini didasarkan padaminimasi fungsi obyektif yang menggambarkan jarak dari titik data yang diberikan kepusat *cluster* yang terbobot oleh derajat keanggotaan titik data tersebut.

Output dari *Fuzzy C-Means* merupakan deretan pusat *cluster* dan beberapaderajat keanggotaan untuk tiap-tiap titik data. Informasi ini dapat digunakan untuk membangun suatu *fuzzy inference system*.

2.5.3 Algoritma *Fuzzy C-Means*

Algoritma *Fuzzy C-Means* adalah sebagai berikut:

1. *Input* data yang akan dicluster X , berupa matriks berukuran $n \times m$ (n =jumlah sample data, m =atribut setiap data). X_{ij} =data sample ke- i ($i=1,2,\dots,n$), atribut ke- j ($j=1,2,\dots,m$).

2. Tentukan:

- Jumlah *cluster* $= c$;
- Pangkat $= w$;
- Maksimum iterasi $= \text{MaxIter}$;
- Error terkecil yang diharapkan $= \zeta$;
- Fungsi obyektif awal $= P_0 = 0$;
- Iterasi awal $= t = 1$;

3. Bangkitkan nilai acak ∞_{ik} , $i=1,2,\dots,n$; $k=1,2,\dots,c$; sebagai elemen-elemen matriks partisi awal u .

∞_{ik} adalah derajat keanggotaan yang merujuk pada seberapa besar kemungkinan suatu data bisa menjadi anggota ke dalam suatu *cluster*. Posisi dan nilai matriks dibangun secara random. Dimana nilai keanggotaan terletak pada interval 0 sampai dengan 1. Pada posisi awal matriks partisi U masih belum akurat begitu juga pusat *cluster*nya. Sehingga kecenderungan data untuk masuk suatu *cluster* juga belum akurat.

Hitung jumlah setiap kolom (atribut) Menggunakan persamaan berikut (Yan,1994) :

$$Q_j = \sum_{k=1}^c \mu_{ik}$$

Q_j adalah jumlah nilai derajat keanggotaan perkolom $= 1$ dengan $j=1,2,\dots,m$, Menggunakan persamaan berikut Yan, 1994) :

$$\mu_{ik} = \frac{\mu_{ik}}{Q_j}$$

4. Hitung pusat *Cluster* ke-k: V_{kj} , dengan $k=1,2,\dots,c$; dan $j=1,2,\dots,m$. (Yan, 1994) :

$$V_{kj} = \frac{\sum_{i=1}^n ((\mu_{ik})^w * X_{ij})}{\sum_{i=1}^n (\mu_{ik})^w}$$

5. Hitung fungsi obyektif pada iterasi ke-t, P_t .

Fungsi obyektif digunakan sebagai syarat perulangan untuk mendapatkan pusat cluster yang tepat. Sehingga diperoleh kecenderungan data untuk masuk ke *cluster* mana pada *step* akhir.

Menggunakan persamaan berikut (Yan, 1994) :

$$P_t = \sum_{i=1}^n \sum_{k=1}^c ([\sum_{j=1}^m (X_{ij} - V_{kj})^2] (\mu_{ik})^w)$$

6. Hitung perubahan matriks partisi menggunakan persamaan berikut (Yan, 1994) :

$$\mu_{ik} = \frac{[\sum_{j=1}^m (X_{ij} - V_{kj})^2]^{\frac{-1}{w-1}}}{\sum_{k=1}^c [\sum_{j=1}^m (X_{ij} - V_{kj})^2]^{\frac{-1}{w-1}}}$$

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

7. cek kondisi berhenti:

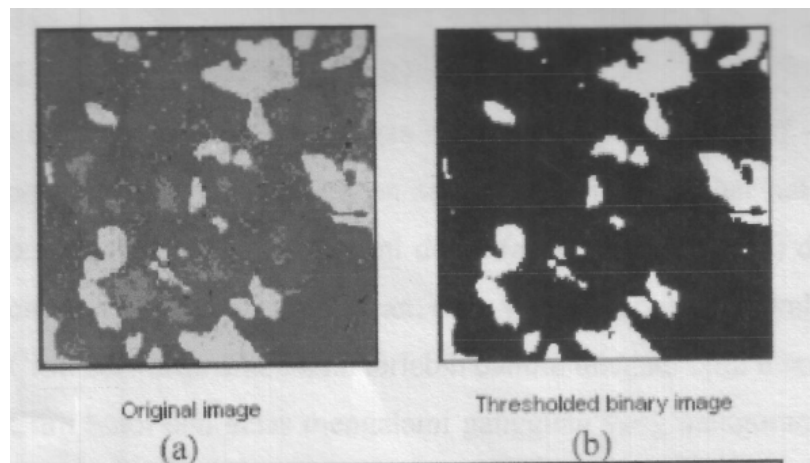
- jika: $(|P_t - P_{t-1}| < \zeta)$ atau $(t > \maxIter)$ maka berhenti;
- jika tidak: $t = t+1$, ulangi langkah ke-4.

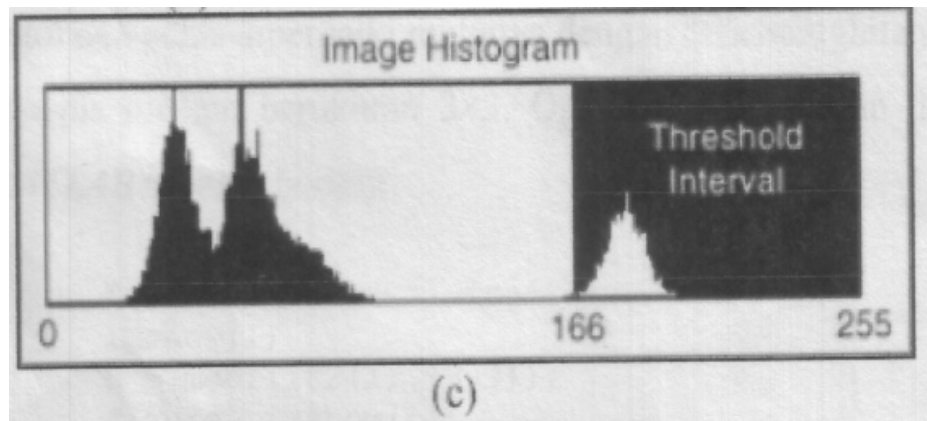
2.6 Segmentasi Citra

Segmentasi citra bertujuan untuk memisahkan wilayah (Region) objek dengan wilayah latar belakang agar objek didalam citra mudah dianalisis dalam rangka mengenali objek.

Pengembangan citra (image thresholding) merupakan metode yang paling sederhana untuk melakukan segmentasi. Operasi pengembangan membagi citra menjadi dua wilayah, yaitu wilayah objek dan wilayah latar belakang. Wilayah objek diset berwarna putih sedangkan sisanya diset berwarna hitam (atau sebaliknya). Hasil dari operasi pengembangan adalah citra biner yang hanya mempunyai dua derajat keabuan : hitam dan putih.

Nilai ambang T dipilih sedemikian sehingga galat yang dipilih sekecil mungkin. Cara yang umum menentukan nilai T adalah dengan membuat histogram citra. Nilai T dapat dipilih secara manual atau dengan teknik yang otomatis. Teknik yang manual dilakukan dengan cara coba-coba (trial and error) dan menggunakan histograk sebagai panduan.





Gambar 2.7 Segmentasi Citra.

Gambar 2.4 memperlihatkan contoh segmentasi citra dengan menggunakan operasi pengambangan. Gambar 2.4(a) adalah citra awal (sebelum disegmentasi) yang memperlihatkan partikel-partikel. Gambar 2.4(c) adalah histogram citra yang sudah dinormalisasi. Kita dapat melihat bahwa histogram ini memperlihatkan ada 3 wilayah didalam gambar. Wilayah pertama adalah latar belakang gambar yang gelap yang ditunjukkan dengan bagian histogram paling kiri dengan sebuah puncak. Wilayah kedua menunjukkan partikel-partikel yang berwarna lebih gelap yang ditunjukkan dengan bagian histogram tengah dengan sebuah. Wilayah ketiga menunjukkan partikel-partikel berwarna terang yang ditunjukkan dengan bagian histogram paling kanan. Wilayah kedua dan wilayah ketiga dapat dibedakan dengan sangat jelas karena kedua puncaknya dipisahkan dengan jarak yang jauh. Jika kita ingin menganalisis partikel yang lebih terang, kita dapat memisahkannya dari latar belakang dan partikel yang lebih gelap dengan melakukan operasi pengambangan. Nilai ambang yang dipilih diperlihatkan pada gambar 2.4(c). hasilnya adalah citra biner dengan partikel-partikel lebih terang diset berwarna putih sedangkan gambar bagian lainnya diset hitam.

Untuk mendapatkan hasil segmentasi yang bagus, beberapa operasi perbaikan kualitas citra dilakukan terlebih dahulu untuk mempertajam batas antara objek dengan latar belakangnya.

2.7 Distance (jarak)

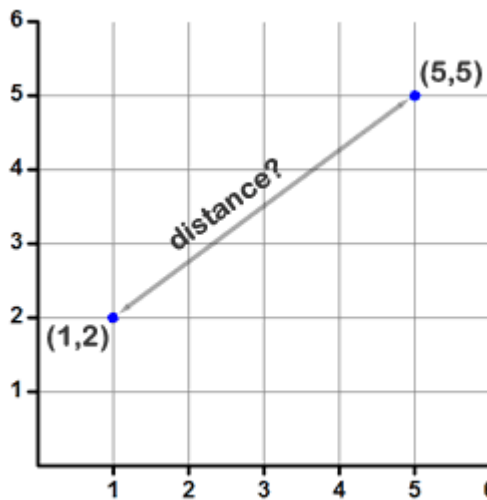
Jarak digunakan untuk menentukan tingkat kesamaan (similarity degree) atau ketidaksamaan (disimilarity degree) dua vektor fitur. Tingkat kesamaan berupa suatu nilai (score) dan berdasarkan score tersebut dua vektor fitur akan dikatakan mirip atau tidak. Berikut ini berbagai jenis metode jarak yang dapat digunakan untuk mengukur tingkat kemiripan dua buah vektor fitur.

2.7.1 Euclidean Distance

Euclidean distance adalah perhitungan jarak dari 2 buah titik dalam Euclidean space. Euclidean space diperkenalkan oleh Euclid, seorang matematikawan dari Yunani sekitar tahun 300 B.C.E. untuk mempelajari hubungan antara sudut dan jarak. Euclidean ini berkaitan dengan Teorema Pythagoras dan biasanya diterapkan pada 1, 2 dan 3 dimensi. Tapi juga sederhana jika diterapkan pada dimensi yang lebih tinggi.

Semisal ingin menghitung jarak Euclidean 1 dimensi. Titik pertama adalah 4, titik kedua adalah -10. Caranya adalah kurangkan -10 dengan 4, sehingga menghasilkan -14. Cari nilai absolut dari nilai -14 dengan cara memangkatkannya sehingga mendapat nilai 196. Kemudian diakarkan sehingga mendapatkan nilai 14. Sehingga jarak euclidean dari 2 titik tersebut adalah 14.

Kemudian pada 2 dimensi



Gambar 2.8 koordinat jarak

Caranya hampir sama. Misalkan titik pertama mempunyai kordinat (1,2). Titik kedua ada di kordinat (5,5). Caranya adalah kurangkan setiap kordinat titik kedua dengan titik yang pertama. Yaitu, (5-1,5-2) sehingga menjadi (4,3). Kemudian pangkatkan masing-masing sehingga memperoleh (16,9). Kemudian tambahkan semuanya sehingga memperoleh nilai $16+9 = 25$. Hasil ini kemudian diakarkan menjadi 5. Sehingga jarak euclideannya adalah 5.

Rumus dari Euclidean distance :

$$d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

2.8 C#.Net

C# merupakan sebuah bahasa pemrograman yang berorientasi objek yang dikembangkan oleh Microsoft sebagai bagian dari inisiatif kerangka .NET Framework. Bahasa pemrograman ini dibuat berbasiskan bahasa C++ yang telah dipengaruhi oleh aspek-aspek ataupun fitur bahasa yang terdapat pada bahasa-bahasa pemrograman lainnya seperti Java, Delphi, Visual Basic, dan lain-lain) dengan beberapa penyederhanaan. Menurut standar ECMA-334 C# *Language Specification*, nama C# terdiri atas sebuah huruf Latin C (U+0043) yang diikuti oleh tanda pagar yang menandakan angka # (U+0023). Tanda

pagar # yang digunakan memang bukan tanda kres dalam seni musik (U+266F), dan tanda pagar # (U+0023) tersebut digunakan karena karakter kres dalam seni musik tidak terdapat di dalam keyboard standar.

Standar *European Computer Manufacturer Association* (ECMA) mendaftarkan beberapa tujuan desain dari bahasa pemrograman C#, sebagai berikut:

- a. Bahasa pemrograman C# dibuat sebagai bahasa pemrograman yang bersifat bahasa pemrograman general-purpose (untuk tujuan jamak), berorientasi objek, modern, dan sederhana.
- b. Bahasa pemrograman C# ditujukan untuk digunakan dalam mengembangkan komponen perangkat lunak yang mampu mengambil keuntungan dari lingkungan terdistribusi.
- c. Portabilitas programmer sangatlah penting, khususnya bagi programmer yang telah lama menggunakan bahasa pemrograman C dan C++.
- d. Dukungan untuk internasionalisasi (*multi-language*) juga sangat penting.
- e. C# ditujukan agar cocok digunakan untuk menulis program aplikasi baik dalam sistem klien-server (*hosted system*) maupun sistem *embedded* (*embedded system*), mulai dari perangkat lunak yang sangat besar yang menggunakan sistem operasi yang canggih hingga kepada perangkat lunak yang sangat kecil yang memiliki fungsi-fungsi terdedikasi.
- f. Meskipun aplikasi C# ditujukan agar bersifat 'ekonomis' dalam hal kebutuhan pemrosesan dan memori komputer, bahasa C# tidak ditujukan untuk bersaing secara langsung dengan kinerja dan ukuran perangkat lunak yang dibuat dengan menggunakan bahasa pemrograman C dan bahasa rakitan.
- g. Bahasa C# harus mencakup pengecekan jenis (*type checking*) yang kuat, pengecekan larik (*array*), pendeteksian terhadap percobaan

terhadap penggunaan Variabel-variabel yang belum diinisialisasikan, portabilitas kode sumber, dan pengumpulan sampah (*garbage collection*) secara otomatis. (Adi, 2013)

2.9 Editor C#

Berikut beberapa *editor* yang bisa digunakan, dimana *editor-editor* ini mempunyai kelebihan dan kekurangan masing-masing. *Editor-editor* tersebut antara lain:

1. Notepad

Microsoft Notepad telah banyak digunakan sebagai *editor* berbasis teks untuk menulis berbagai macam bahasa pemrograman, termasuk C#. Namun terdapat beberapa alasan *notepad* tidak direkomendasikan untuk digunakan sebagai *editor* untuk membuat aplikasi dengan C#, yaitu:

- *File-file* C# disimpan dengan *extension* “.cs”. Jika tidak hati-hati dalam menyimpan *file* C# di *notepad*, maka tidak menutup kemungkinan *file* tersebut akan menjadi “.cs.txt”.
- *Notepad* tidak dapat menampilkan nomor baris, hal ini akan menyulitkan pada saat *mendebug* dari kemungkinan *error* yang ada yang memberikan informasi di baris mana *error* tersebut terjadi.
- *Notepad* tidak dapat melakukan *automatic indenting* (tab secara otomatis) sehingga kita harus melakukannya secara manual.

Meskipun demikian, *notepad* masih bisa digunakan sebagai *editor* untuk membuat aplikasi-aplikasi C# sederhana dan kecil dengan tetap memperhatikan beberapa permasalahan diatas.

2. Visual Studio 6

Salah satu keuntungan menggunakan *editor* khusus pemrograman (seperti *Microsoft Visual C++*) adalah adanya *syntax highlighting*, yang memudahkan pada saat membaca dan menganalisis kode-kode program. Namun, karena *visual studio 6* ini dibuat sebelum adanya bahasa C#,

maka untuk memodifikasi *setting editor* tersebut agar dapat menampilkan *syntax highlighting C#*.

3. *Visual Studio.NET*

Visual Studio.NET merupakan *editor* yang paling ideal untuk membuat aplikasi berbasis *Framework.NET*, termasuk aplikasi dengan bahasa C#. Di dalam modul ini, semua kode program akan ditulis dengan menggunakan *Visual Studio.NET 2005*. Dengan *editor* ini, maka dapat memanfaatkan kemampuan C# secara maksimal. *Editor* ini tidak hanya menyediakan berbagai macam *tools* dan *wizard* untuk membuat aplikasi C#, tetapi juga termasuk fitur-fitur produktif seperti *IntelliSense* dan bantuan yang dinamis. Dengan *IntelliSense*, dalam mengetik nama sebuah *namespace* atau nama *class*, maka anggota dari *namespace* atau *class* akan muncul secara otomatis sehingga tidak perlu lagi mengingat anggota dari semua *namespace* atau semua *class* yang digunakan.

4. *Editor-editor* lainnya

Selain *editor-editor* diatas, masih banyak *editor* lain yang dapat digunakan untuk membuat aplikasi dengan C# seperti *Visual SlickEdit* dari *MicroEdge*, *WebMatrix* untuk aplikasi C# berbasis web, *editor text* seperti *UltraEdit*, *Macromedia Homesite*, dan sebagainya. (Adi, 2013)

BAB III

METODE PENELITIAN

Penelitian ini dilakukan dengan tahapan-tahapan sebagai berikut:

3.1. Tahap studi literatur

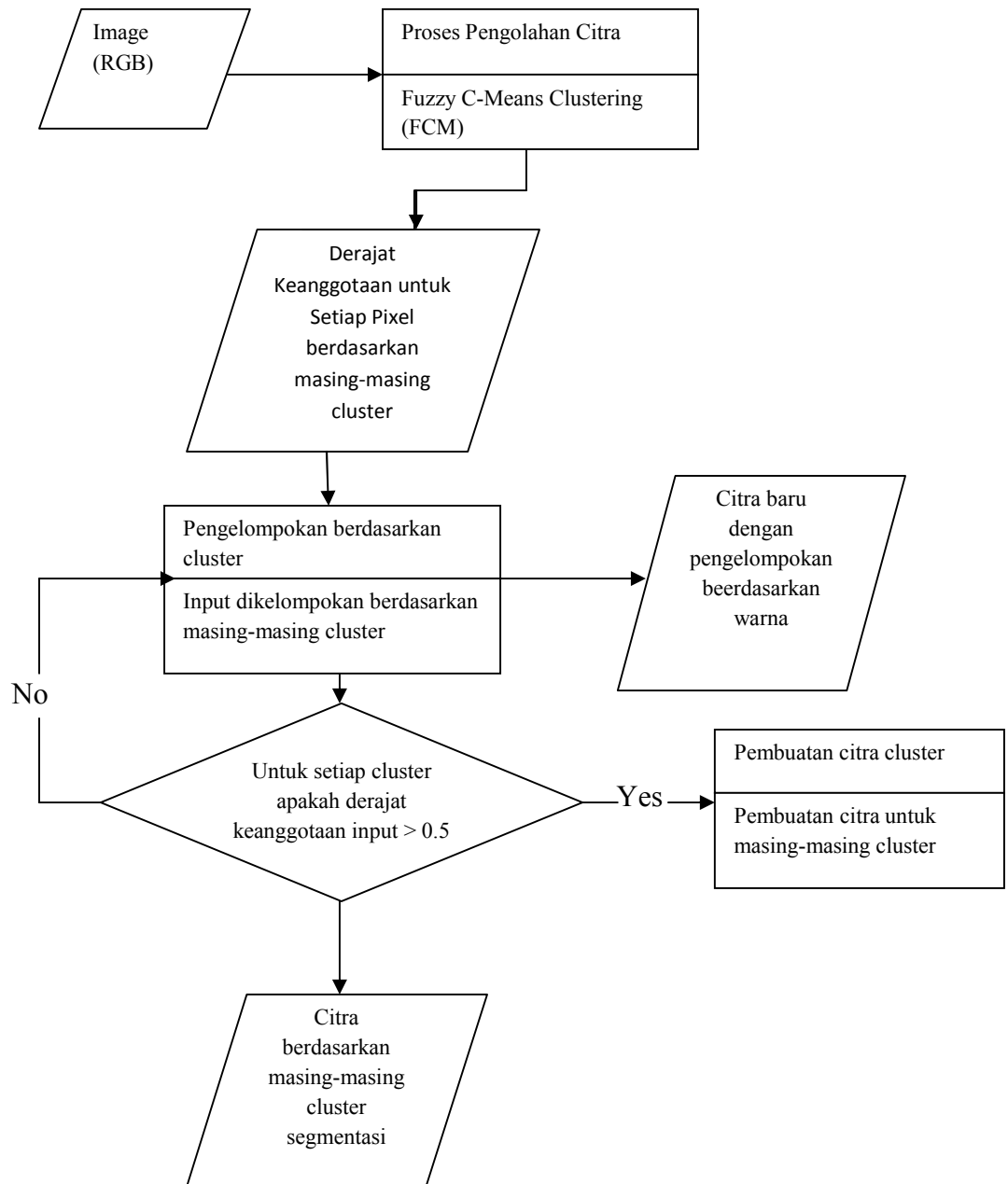
Mengumpulkan data melalui studi literatur dari buku-buku referensi dan jurnal yang terkait. Studi literatur merupakan tahap pendalaman materi, identifikasi permasalahan dan teori yang berkaitan dengan permasalahan dalam penelitian.

3.2. Tahap persiapan alat dan bahan

Alat dan bahan yang digunakan di dalam penelitian ini:

- 1) perangkat Lunak (software)
 - a. Sistem Operasi Windows 7 (64-bit)
 - b. Bahasa Pemrograman C# menggunakan Visual Studio 2012
- 2) Perangkat keras (Hardware)
 - a. Processor Intel Core i5 - 2,50 GHz
 - b. 4 GB RAM
- 3) Sejumlah citra digital sebagai citra uji.

3.3. Gambaran algoritma segmentasi



Gambar 3.1 Algoritma Proses Segmentasi citra berdasarkan fuzzy C-Means

Algoritma proses segmentasi berdasarkan warna yaitu input dari citra yang berupa citra RGB akan diproses oleh fuzzy C-Means Clustering yang disebut juga sebagai proses pengolahan citra. Output yang dihasilkan dari proses pengolahan citra ini adalah derajat keanggotaan dari setiap piksel berdasarkan masing-masing cluster. Output ini kemudian menjadi input untuk proses pengelompokan atau segmentasi, proses yang dilakukan adalah mengelompokkan masing-masing piksel kedalam kelompok warna (cluster) yang telah ditentukan berdasarkan derajat keanggotaan diatas 0,5. Proses ini akan menghasilkan output citra baru dengan pengelompokan berdasarkan warna(cluster) yang telah ditentukan dan output data baru yang berisi data citra per cluster yang ada. Output data baru tersebut yang akan digunakan dalam proses pembuatan citra baru hasil segmentasi, dimana citra baru tersebut hanya akan menampilkan citra dengan warna sesuai dengan cluster yang telah ditentukan.

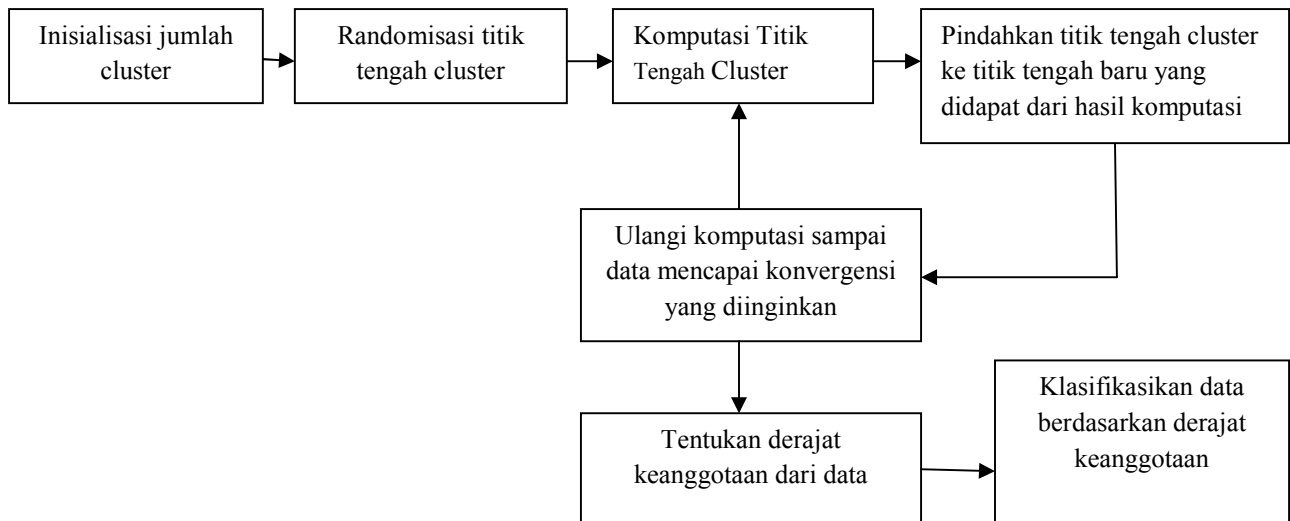
3.4 Gambaran Metode Fuzzy C-means

Fuzzifikasi adalah proses mengubah nilai-nilai yang tadinya bernilai pasti menjadi nilai yang bersifat samar. Contohnya : Andi berusia 50 tahun menjadi andi tua. Defuzzifikasi adalah proses kebalikannya yaitu mengubah nilai yang bersifat samar menjadi nilai yang pasti. Sedangkan proses clustering didalam Fuzzy C-Means mempunyai beberapa langkah penting. Langkah-langkah yang ditempuh dalam menggunakan fuzzy c-means adalah :

1. Pilih jumlah cluster atau jumlah segmentasi yang diinginkan
2. Secara random inialisasi sebanyak jumlah cluster yang diinginkan didalam gambar sebagai titik tengah dari cluster tersebut.
3. Komputasi titik tengah cluster-cluster tersebut.
4. Pindahkan titik tengah cluster ke titik tengah baru yang didapat dari hasil komputasi.
5. Ulangi kedua langkah terakhir sampai kriteria konvergensi telah terpenuhi, atau sampai titik-titik tengah tersebut tidak berubah keadaannya.

6. Setelah kriteria tersebut terpenuhi, maka akan didapat derajat keanggotaan dari masing-masing data.
7. Klasifikasi data tersebut kedalam cluster-cluster yang telah ditentukan.

Untuk lebih jelas tentang langkah-langkah Fuzzy C-Means dapat dilihat pada gambar 3.4 dibawah ini



Gambar 3.2 Langkah-langkah Fuzzy C-Means

3.5 Proses Segmentasi citra dengan metode Fuzzy C-means

- a) Menentukan Peta matriks yang akan diuji pada cluster X



(a) Citra Satelit 500x500 pixel (b) piksel dengan sampel 4x4

Nilai matrik X diambil dari hasil input citra pada tabel 1 dibawah ini :

Tabel 1 matrik kluster yang akan di uji

X(i,j)	1	2	3	4(a)
1	122	144	148	95
2	155	133	101	84
3	160	112	68	73
4(b)	160	99	65	72

- b) Menentukan Variabel proses kluster

- a) Jumlah kelompok yang akan dibentuk = C, dimana $C > 1$.

Pada perhitungan ini akan diberikan nilai $C = 4$.

- b) Pangkat = w dimana, $w > 1$. Pada perhitungan ini akan diberikan nilai $w = 2$.

- c) Maksimum iterasi = maxIter. Pada perhitungan ini akan diberikan nilai maxIter = 5.

- d) Error terkecil yang diharapkan, $\varepsilon = 1 * 10^{-6}$.

- e) Fungsi objektif awal, $P_0 = 0$

- f) Iterasi awal, $t = 1$

- c) Membangkitkan Nilai Random

Untuk menentukan nilai random μ_{ik} sebagai elemen matriks partisi awal U maka terlebih dahulu dicari nilai atribut Q_j (total nilai setiap elemen matriks $X_{ij} = 1791$). Untuk menentukan nilai random

μ pada posisi (i,k) dilakukan dengan cara membagi nilai pada elemen X_{ij} dengan 1791.

Contoh :

$$\begin{aligned}\mu_{11} &= x_{11} / Q_j \\ &= 122 / 1791 \\ &= 0.068118\end{aligned}$$

Berikut merupakan nilai random dari hasil perhitungan yang telah dilakukan pada tabel 2 dibawah ini :

Tabel 2 perhitungan nilai random derajat keanggotaan

μ_{ik}	1	2	3	4
1	0.068118	0.080402	0.082635	0.053042
2	0.086543	0.074260	0.056393	0.046901
3	0.089335	0.062534	0.037967	0.040759
4	0.089335	0.055276	0.036292	0.040201

d) Menghitung pusat Cluster

Untuk menghitung pusat cluster ke-k maka digunakan persamaan berikut :

$$V_{kj} = \frac{\sum_{i=1}^n ((\mu_{ik})^w, X_{ij})}{\sum_{i=1}^n (\mu_{ik})^w}$$

Dengan :

$K = 1, 2, \dots, c$

$J = 1, 2, \dots, m$

V_{kj} = pusat cluster ke-k untuk atribut ke-j

μ_{ik} = derajat keanggotaan untuk data sampel ke-i pada cluster ke-k

X_{ij} = data ke-i, atribut ke-j

Dengan menggunakan persamaan diatas maka akan diperoleh hasil seperti pada tabel 3 dibawah ini :

Tabel 3.hasil perhitungan pusat cluster

Vkj	1	2	3	4
1	0.566087	0.930885	1.010624	0.267278
2	1.160902	0.733434	0.321197	0.184775
3	1.276918	0.437976	0.098021	0.121274
4	1.276918	0.302488	0.085612	0.116360

e) Hitung fungsi objektif pada Iterasi ke-t

Nilai fungsi objektif pada iterasi ke-t dapat dihitung menggunakan persamaan berikut :

$$P_t = \sum_{i=1}^n \sum_{k=1}^c ([\sum_{j=1}^m (X_{ij} - V_{kj})^2](\mu_{ik})^w)$$

Dengan :

Vkj = pusat cluster ke-k untuk atribut ke-j

μik = derajat keanggotaan untuk data sampel ke-i pada cluster ke-k

Xij = data ke-i, atribut ke-j

Pt = fungsi objektif pada iterasi ke-t

Nilai fungsi objektif akan menentukan jumlah iterasi yang akan dilakukan untuk mendapat hasil akhir dari proses clustering. Dari persamaan fungsi objektif diatas akan diperoleh hasil :

Tabel 4 hasil perhitungan (Xik-Vkj)^2

14746.1952	20468.7716	21605.8766	8974.28861
23666.4680	17494.4444	10181.2093	7024.99194
25193.0167	12446.0852	4610.6787	5311.3087
25193.0167	9741.1988	4213.8777	5167.2577

Tabel 5 hasil perhitungan $(X_{ik}-V_{kj})^2 * (\mu_{ik})^2$

68.4232	132.219	147.536	25.2431
177.254	96.473	32.3779	15.4529
201.058	48.670	6.6462	8.58371
201.058	29.763	5.7438	8.35091

Dari tabel diatas diperoleh $Pt = 1204.852$

f) Menghitung perubahan Matriks Partisi

Pada tahap ini akan dilakukan kalkulasi terhadap perubahan matriks proses Fuzzy c-means yang akan digunakan sebagai nilai random untuk iterasi t+1. Untuk menghitung perubahan matriks partisi menggunakan persamaan berikut :

$$\mu_{ik} = \frac{[\sum_{j=1}^m (X_{ij} - V_{kj})^2]^{\frac{-1}{w-1}}}{\sum_{k=1}^c [\sum_{j=1}^m (X_{ij} - V_{kj})^2]^{\frac{-1}{w-1}}}$$

Dengan $i= 1,2,...,a$ dan $k= 1,2,...c$

Dimana :

V_{kj} =pusat cluster ke-k untuk atribut ke-j

μ_{ik} = derajat keanggotaan untuk data sampel ke-i pada cluster ke-k

X_{ij} = data ke-i, atribut ke-j

Dari hasil perhitungan menggunakan persamaan perubahan matriks maka diperoleh hasil berikut :

Tabel 6 hasil perhitungan perubahan matriks

μ_{ik}	1	2	3	4
1	6.7814E-5	4.8854E-5	4.6283E-5	1.1142E-4
2	4.2253E-5	5.7161E-5	9.8220E-5	1.4234E-4
3	3.9693E-5	8.0346E-5	2.1688E-4	1.8827E-4
4	3.9693E-5	1.0265E-4	2.3731E-4	1.935E-4

Setelah menghitung perubahan matriks maka akan dilakukan pengujian kondisi stop pada proses fuzzy c-means.

Jika

$$P_t - P_{t-1} < \varepsilon;$$

Atau $t > \text{MaxIterasi}$;

Maka

End Process()

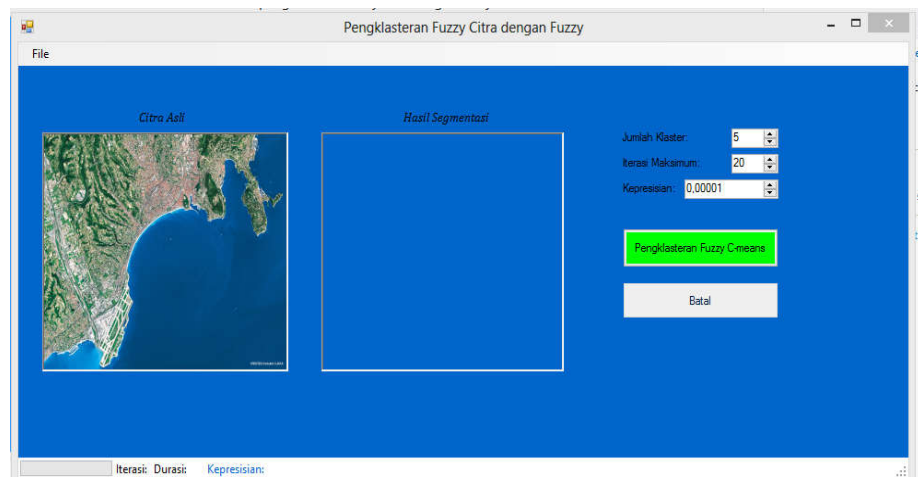
Jika tidak,

$$t = t + 1$$

then

ulangi step d

3.6 Perancangan antar muka



Gambar 3.3 Tampilan antar muka sistem

Antar muka system ini berfungsi untuk mempermudah pengoperasian Segmentasi Citra. Aplikasi ini dibangun menggunakan bahasa programan C# pada editor Visual studio 2012. Dimana jenis file yang dapat dimuat disini adalah file gambar bertipe bitmap(.bmp)