

BAB II: TINJAUAN PUSTAKA

Bab ini akan memberikan penjelasan awal mengenai konsep logika *fuzzy* beserta pengenalan sistem inferensi *fuzzy* secara umum.

2.1 LOGIKA FUZZY

Konsep mengenai logika *fuzzy* diawali pada tahun 1965, saat seorang professor bernama Lutfi A. Zadeh memodifikasi teori himpunan dimana setiap anggotanya memiliki derajat keanggotaan yang bernilai kontinu antara 0 sampai 1. Himpunan ini kemudian disebut sebagai Himpunan Kabur (*Fuzzy Set*). Himpunan ini diinspirasi dari kehidupan sehari-hari, dimana sulit untuk memutuskan masalah dengan jawaban “Ya” atau “Tidak”. Contohnya, adalah relatif untuk menyatakan seseorang berbadan “tinggi” atau “pendek”. Misalkan, jika orang yang berbadan pendek memiliki kisaran nilai tinggi badan dari 0 – 100 cm, maka bila orang tersebut bertinggi badan 100,1 cm akan dikatakan sebagai “tinggi”, sedangkan orang yang bertinggi badan 100 cm akan dianggap “pendek”. Hal ini menjadi bersifat membingungkan dan tidak adil. Sehingga, kemunculan dari logika *fuzzy* sangat diperlukan untuk merepresentasikan estimasi manusia yang bersifat ambigu.

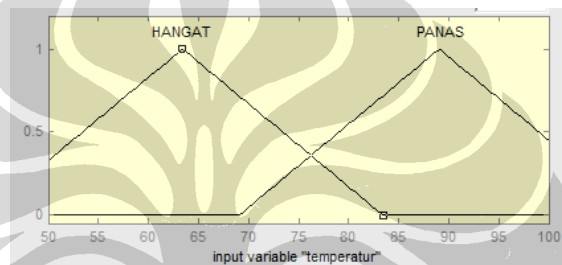
Definisi dari logika *fuzzy* sendiri adalah suatu cara yang tepat untuk memetakan suatu ruang input ke dalam suatu ruang output [1]. Sebagai contoh:

- Penumpang taksi berkata kepada sopir taksi secepat laju kendaraan yang diinginkan, sopir taksi akan mengatur pijakan gas taksinya.

- Anda mengatakan kepada saya seberapa sejuk suhu ruangan yang anda inginkan, saya akan mengatur suhunya.

Kedua contoh di atas menggambarkan bagaimana suatu sistem dapat dikembangkan untuk memahami keinginan manusia yang tidak dapat direpresentasikan secara matematis, untuk kemudian memberikan output yang sesuai.

Logika *fuzzy*, atau yang lebih dikenal sebagai himpunan *fuzzy*, haruslah bersifat saling *overlap* satu sama lain. Untuk lebih jelasnya perhatikan gambar berikut:



Gambar 1: Temperatur dengan kisaran nilai fungsi {0,1} dengan sumbu-x = variabel suhu (°C)

Gambar 1 tersebut merepresentasikan dua buah himpunan *fuzzy*, yang posisinya saling *overlap*. Maksud dari *overlap* tersebut adalah bahwa nilai temperatur HANGAT akan turun di suatu sisi, dimana pada saat itu juga nilai temperatur PANAS akan naik yang menandakan adanya pergantian kondisi dari temperatur udara tersebut. Jika kedua himpunan tersebut tidak saling *overlap*, maka tidak ada hubungan antara temperatur udara HANGAT dengan PANAS, padahal kita tahu bahwa jika temperatur udara semakin naik menjauhi kondisi HANGAT, maka pada saat yang bersamaan temperatur udara akan semakin mendekati kondisi PANAS.

2.2 ALASAN PENGGUNAAN LOGIKA FUZZY

Dalam pembuatan Prototipe Sistem Pengawasan Pergerakan Sederhana Manusia Menggunakan Sistem Inferensi Fuzzy ini, penulis akan menggunakan konsep logika *fuzzy* dan menerapkannya dalam bentuk pembuatan sistem inferensi *fuzzy*. Alasan penggunaan konsep tersebut adalah [1]:

- Konsep logika *fuzzy* merupakan konsep matematis yang mendasari penalaran secara sederhana dan mudah dimengerti.
- Logika *fuzzy* bersifat fleksibel dan memiliki toleransi terhadap data-data yang tidak tepat, terutama untuk keterkaitan antara data-data seperti jumlah piksel pada citra dengan jumlah orang yang direpresentasikan.
- Logika *fuzzy* dapat memodelkan fungsi-fungsi nonlinear yang sangat kompleks, seperti fungsi *Gauss*.

2.3 HIMPUNAN CRISP DAN HIMPUNAN FUZZY

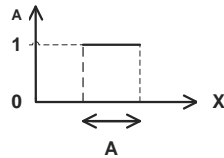
2.3.1 Himpunan Crisp

Himpunan *fuzzy* yang akan digunakan pada pembuatan Tugas Akhir ini didasari pada gagasan untuk memperluas jangkauan fungsi karakteristik dari himpunan *crisp*. Himpunan *crisp* yang dimaksud memiliki definisi matematis sebagai berikut:

Diketahui, $X = \{x\}$: himpunan universal, A : himpunan *crisp* (pada X), μ = fungsi *crisp*, $\mu_A: X \rightarrow \{1,0\}$

Jika $x \in X$ dan $\mu_A(x) \in \{1,0\}$, maka $\mu_A(x) = \begin{cases} 0, & x \notin A \\ 1, & x \in A \end{cases}$

Fungsi ini bersifat diskret. Sehingga grafik yang dihasilkan adalah sebagai berikut:



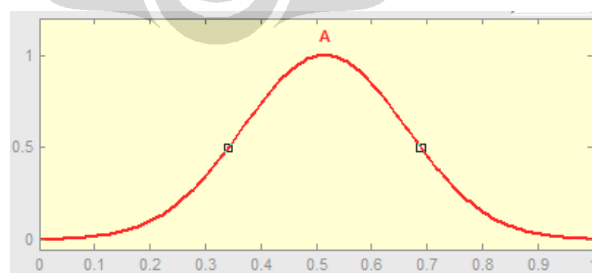
Gambar 2: Grafik himpunan *crisp*

2.3.2 Himpunan *Fuzzy*

Sedangkan fungsi pada himpunan *fuzzy* memiliki karakteristik nilai fungsi yang mencakup bilangan *real* pada interval $[0,1]$, berbeda dengan himpunan *crisp* dimana jika nilai fungsi tidak 0, maka pastilah 1. Berikut definisi matematisnya:

Diketahui $X = \{x\}$: himpunan universal, A : himpunan *fuzzy* (pada X), μ = fungsi *fuzzy*, $\mu_A: X \rightarrow [1,0]$

Fungsi *fuzzy* ini bernilai bilangan *real* antara 0 hingga 1 serta bersifat kontinu. Sehingga grafik yang dihasilkan adalah sebagai berikut:



Gambar 3: Grafik himpunan *fuzzy*

2.4 OPERATOR-OPERATOR HIMPUNAN FUZZY

Satu atau lebih himpunan *fuzzy* dapat melakukan operasi matematika dengan bantuan beberapa operator khusus. Berikut adalah beberapa operator dasar yang sering digunakan terutama dalam pembentukan sistem inferensi *fuzzy*:

2.4.1 Operator *t-norm* (\otimes)

Bila diberikan dua buah input, x_1 dan x_2 , maka sifat-sifat dari operator tersebut adalah seperti pada tabel berikut:

Tabel 1: Tabel operasi menggunakan *t-norm* pada input x_1 dan x_2

| x_1 | x_2 | $x_1 \otimes x_2$ |
|-------|-------|-------------------|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

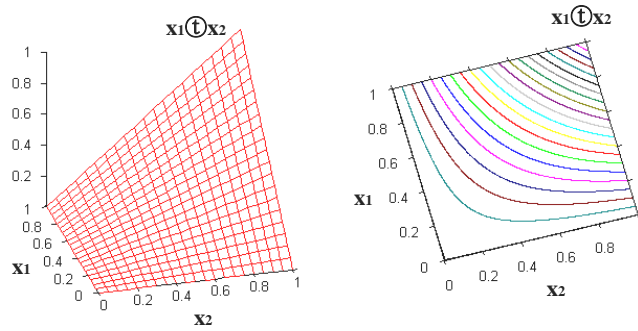
Sedangkan empat buah teorema yang berkaitan dengan operator *t-norm* adalah:

Jika diketahui: $\forall x_1, x_2, x_3, x \in [1,0]$

Maka:

- $x_1 \otimes 0 = 0, x_1 \otimes 1 = x$
- $x_1 \otimes x_2 = x_2 \otimes x_1$
- $x_1 \otimes (x_2 \otimes x_3) = (x_1 \otimes x_2) \otimes x_3$
- $x_1 \leq x_2 \rightarrow x_1 \otimes x_3 \leq x_2 \otimes x_3$

Sehingga grafik yang dihasilkan berdasarkan keempat buah terorema di atas adalah:



Gambar 4: Grafik Teorema *t-norm*

2.4.2 Operator *s-norm* (⊕)

Bila diberikan dua buah input, x_1 dan x_2 , maka sifat-sifat dari operator *s-norm* adalah seperti pada tabel berikut:

Tabel 2: Tabel operasi menggunakan *s-norm* pada input x_1 dan x_2

| x_1 | x_2 | $x_1 \oplus x_2$ |
|-------|-------|------------------|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

Sedangkan empat buah teorema yang berkaitan dengan operator *s-norm* adalah:

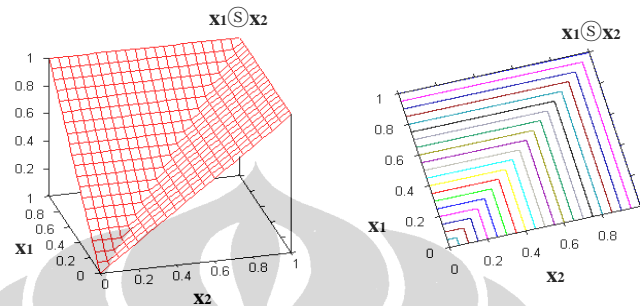
Jika diketahui: $\forall x_1, x_2, x_3, x_i \in [1,0]$

Maka:

- $x \oplus 0 = x, x \oplus 1 = 1$
- $x_1 \oplus x_2 = x_2 \oplus x_1$

- $x_1 \textcircled{S} (x_2 \textcircled{S} x_3) = (x_1 \textcircled{S} x_2) \textcircled{S} x_3$
- $x_1 \leq x_2 \rightarrow x_1 \textcircled{S} x_3 \leq x_2 \textcircled{S} x_3$

Sehingga grafik yang dihasilkan berdasarkan keempat buah terorema di atas adalah:



Gambar 5: Grafik Teorema *s-norm*

2.5 FUNGSI KEANGGOTAAN

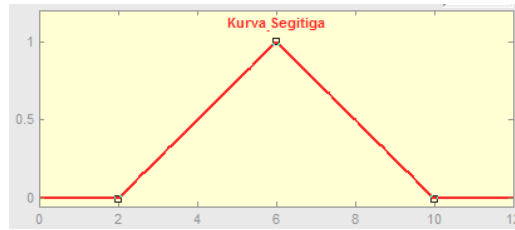
Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik input data ke dalam nilai keanggotaannya (dikenal dengan derajat keanggotaan) yang memiliki interval antara 0 sampai 1 [1]. Fungsi keanggotaan merupakan elemen terpenting untuk merepresentasikan suatu himpunan *fuzzy* secara visual.

Berikut adalah beberapa fungsi keanggotaan yang cukup sering digunakan:

2.5.1 Kurva Segitiga

Kurva ini terdiri dari empat parameter dengan rumus sebagai berikut:

$$f(x, a, b, c) = \begin{cases} 0 & ; x \leq a \\ (x-a)/(b-a) & ; a \leq x \leq b \\ (c-x)/(c-b) & ; b \leq x \leq c \\ 0 & ; c \leq x \end{cases}$$

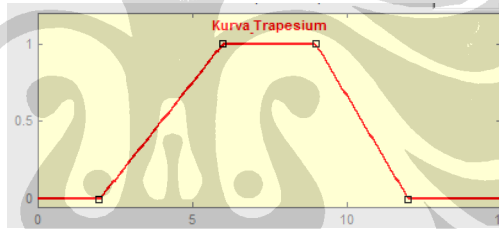


Gambar 6: Contoh kurva segitiga dengan $f(x,a,b,c) = f(x,2,6,10)$

2.5.2 Kurva Trapezium

Kurva ini terdiri dari lima parameter dengan rumus sebagai berikut:

$$f(x, a, b, c, d) = \begin{cases} 0 & ; x \leq a \\ (x-a)/(b-a) & ; a \leq x \leq b \\ 1 & ; b \leq x \leq c \\ (d-x)/(d-c) & ; c \leq x \leq d \\ 0 & ; d \leq x \end{cases}$$

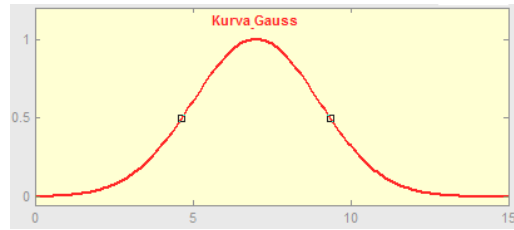


Gambar 7: Contoh kurva trapesium dengan $f(x,a,b,c,d) = f(x,2,6,9,12)$

2.5.3 Kurva Gauss

Kurva ini adalah kurva yang digunakan Uri Kartoun dalam membentuk sistem inferensi *fuzzy* pada prototipenya. Kurva ini terdiri dari tiga parameter dengan rumus sebagai berikut:

$$f(x; \sigma, c) = e^{-\frac{(x-c)^2}{2\sigma^2}}$$



Gambar 8: Contoh kurva *gauss* dengan $f(x,\sigma,c) = f(x,2,7)$

Penentuan fungsi keanggotaan yang tepat didasari dari pembentukan berbagai data yang akan diolah ketika membuat sistem inferensi *fuzzy*.

2.6 SISTEM INFERENSI FUZZY

Sistem inferensi *fuzzy* adalah proses pemetaan satu atau lebih input menjadi satu atau lebih output dengan menggunakan konsep logika *fuzzy* [1]. Pada Tugas Akhir ini, aplikasi berbasis *fuzzy*, atau sistem inferensi *fuzzy*, digunakan untuk mendeteksi jumlah manusia yang terdapat pada suatu kumpulan piksel (dikenal dengan istilah *blob*), serta untuk mendeteksi kecepatan pergerakan maupun arah gerakannya. Semua hal tersebut sulit dilakukan jika menggunakan model matematika biasa karena memerlukan deskripsi matematika secara rinci dan tepat. Seperti, untuk mengetahui berapa kisaran nilai jumlah piksel pada *blob* yang diperlukan agar dapat mengidentifikasi jumlah orang yang terdapat pada *blob* tersebut haruslah memerlukan data yang besar dan lama. Namun dengan menggunakan sistem berbasis *fuzzy*, hal tersebut akan lebih mudah ditelusuri dan dikerjakan.

Metode inilah yang digunakan Uri Kartoun untuk membuat prototipe awal dari Sistem Pengawasan Pergerakan Sederhana Manusia pada tahun 2001. Dan kini, penulis akan berusaha menyempurnakan metode tersebut dengan memodifikasi sistem inferensi *fuzzy* yang ia gunakan, dimana jenis dari sistem inferensi *fuzzy* tersebut adalah sistem inferensi *fuzzy*

yang dicetuskan oleh Ebrahim Mamdani (dikenal dengan nama *fuzzy* Mamdani). Selain Mamdani, adapula beberapa sistem inferensi *fuzzy* yang lainnya, yaitu:

- *Fuzzy* Sugeno → diperkenalkan oleh Michio Sugeno dimana output dari sistem inferensi *fuzzy* ini menerima input berupa nilai *fuzzy* dan memetakan output dalam bentuk konstanta atau persamaan linear.
- *Fuzzy* Tsukamoto → merupakan metode pembentukan sistem inferensi *fuzzy* dimana setiap konsekuensi pada aturan yang berbentuk *if-then* harus direpresentasikan dengan himpunan *fuzzy* yang memiliki fungsi keanggotaan yang monoton serta output hasil inferensi akan diberikan dalam bentuk nilai *crisp* berdasarkan α -predikat (*fire strength*) [3].

Sedangkan *fuzzy* Mamdani sendiri, baik input maupun outputnya merupakan nilai *fuzzy* serta lebih diterima oleh banyak pihak. Selanjutnya, beberapa pertanyaan yang akan timbul mencakup pemilihan sistem inferensi *fuzzy* Mamdani dalam pengembangan prototipe ini dibandingkan dengan metode *fuzzy* yang lain dapat dilihat sebagai berikut:

- Jika menggunakan *fuzzy* Sugeno akan sulit untuk menentukan arah pergerakan dari manusia yang terdeteksi karena sulit untuk memodelkannya dalam bentuk konstanta atau persamaan linear.
- Jika menggunakan *fuzzy* Tsukamoto, maka himpunan *fuzzy* yang dilibatkan hanya terbatas pada fungsi keanggotaan yang monoton saja.
- Dengan menggunakan *fuzzy* Mamdani akan lebih mudah untuk membandingkan hasil yang didapat pada akhir pembuatan prototipe ini mengingat metode serupa juga digunakan oleh Uri Kartoun pada pembuatan prototipe awal.

Setelah mengetahui metode *fuzzy* yang akan digunakan, berikutnya adalah merancang sistem inferensi *fuzzy* yang diinginkan. Tahapan ini mencakup perancangan input-output, fungsi-fungsi keanggotaan yang terdapat pada setiap input dan output, serta aturan-aturan yang merepresentasikan hubungan antar input agar menghasilkan output yang diinginkan (dikenal dengan istilah *if-then rules*). Rincian mengenai tahap perancangan sistem *fuzzy* yang dilibatkan akan dijelaskan lebih rinci pada Bab 5 dan 6.

Setelah perancangan sistem inferensi *fuzzy* tersebut, maka akan dilakukan pengujian nilai-nilai input pada sistem inferensi *fuzzy* tersebut. Nilai-nilai yang akan diuji akan melewati berbagai tahapan tersendiri sebagai berikut:

2.6.1 Fuzzifikasi Variabel Input

Tahap fuzzifikasi adalah tahap untuk mengolah input yang diberikan, misal x_1 dan y_1 , dan menentukan derajat keanggotaannya.



Gambar 9: Contoh penerapan fuzzifikasi pada dua buah input x_1 dan y_1

Pada Gambar 9 sebelah kiri, terlihat bahwa input X memiliki tiga fungsi keanggotaan (A_1 , A_2 , dan A_3), dimana ketika menerima input x_1 , maka akan dievaluasi nilai input x_1 ke dalam sebuah nilai pada setiap fungsi keanggotaan sebagai berikut, $\mu_{(x=A1)} = 0.5$, $\mu_{(x=A2)} = 0.2$, dan $\mu_{(x=A3)} = 0$.

Hal yang sama berlaku pada gambar sebelah kanan, dimana hasil evaluasi nilai input y_1 adalah $\mu_{(y=B1)} = 0.1$ dan $\mu_{(y=B2)} = 0.7$

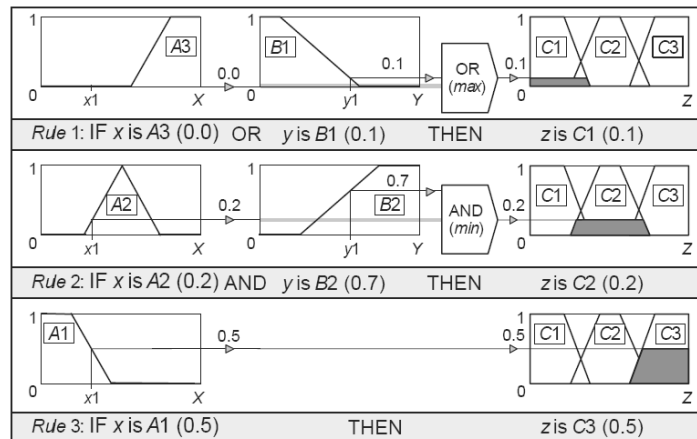
2.6.2 Evaluasi Aturan

Pada tahap ini akan digunakan hasil evaluasi nilai-nilai input yang didapat pada tahap sebelumnya, yaitu, $\mu_{(x=A1)} = 0.5$, $\mu_{(x=A2)} = 0.2$, $\mu_{(x=A3)} = 0$, $\mu_{(y=B1)} = 0.1$ dan $\mu_{(y=B2)} = 0.7$ untuk diaplikasikan sebagai anteseden dari aturan *fuzzy*. Kemudian, dengan menggunakan operator-operator *fuzzy* (seperti *AND*, *OR*, dan *NOT*), akan didapatkan nilai tunggal yang merepresentasikan hasil evaluasi anteseden. Nilai ini (dikatakan sebagai nilai kebenaran) diaplikasikan sebagai konsekuensi dari fungsi keanggotaan [1].

Beberapa contoh operator *fuzzy* yang dimaksud pada paragraf di atas, diterapkan dengan operasi sebagai berikut:

- Operator *fuzzy OR* $\rightarrow \mu_A(x) \cup_B(x) = \max [\mu_A(x), \mu_B(x)]$, digunakan untuk operasi *fuzzy* gabungan.
- Operator *fuzzy AND* $\rightarrow \mu_A(x) \cap_B(x) = \min [\mu_A(x), \mu_B(x)]$, digunakan untuk operasi *fuzzy* irisan.
- Operator *fuzzy NOT* $\rightarrow \neg\mu_A(x) = [1-\mu_A(x)]$, digunakan untuk operasi *fuzzy* komplemen.

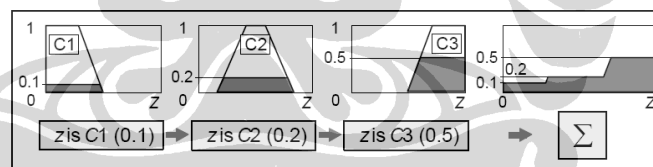
Lihat gambar berikut untuk kejelasan penggunaan operator-operator tersebut pada 3 aturan (*rule*) contoh:



Gambar 10: Contoh penggunaan operator OR dan AND saat mengevaluasi input pada aturan-aturan fuzzy

2.6.3 Aggregasi Aturan Output

Pada tahap ini terjadi proses unifikasi semua aturan keluaran dengan cara mengkombinasikan semua nilai fungsi keanggotaan dari aturan konsekuensi demi mendapatkan sebuah himpunan fuzzy tunggal [1]. Pada gambar berikut diberikan contoh penerapan tahap ini dengan mengkombinasikan tiga buah keluaran aturan pada Gambar 10 (C_1 , C_2 , dan C_3).



Gambar 11: Proses unifikasi himpunan fuzzy C1, C2, dan C3 yang merupakan hasil keluaran pada tahap sebelumnya

2.6.4 Defuzzifikasi

Input dari proses defuzzifikasi adalah suatu himpunan fuzzy yang diperoleh dari komposisi aturan-aturan fuzzy, sedangkan output yang dihasilkan berupa suatu bilangan (nilai crisp) pada domain himpunan fuzzy tersebut.

Ada beberapa metode defuzzifikasi yang digunakan, namun Uri Kartoun memilih untuk menggunakan metode *Centroid* untuk mendapatkan nilai crisp hasil unifikasi tahap sebelumnya. Metode ini bekerja dengan menggunakan rumus *center of gravity (COG)* sebagai berikut:

$$COG = \frac{\int_a^b \mu_A(x) x dx}{\int_a^b \mu_A(x) dx}$$

Meskipun terdapat metode penghasil nilai *crisp* lainnya seperti *Bisektor*, *Mean of Maximum (MOM)*, dan yang lainnya, namun metode *Centroid* dianggap paling mudah dari segi komputasi serta nilai defuzzifikasi yang dihasilkan akan bergerak secara halus sehingga perubahan dari satu topologi ke topologi lainnya akan berjalan secara halus atau lebih mendekati nilai akhir yang sebenarnya.

2.6.5 Konversi Nilai *Crisp* Menjadi Nilai *Fuzzy*

Sebenarnya, nilai *crisp* yang dihasilkan pada tahap di atas merupakan akhir dari evaluasi input-input pada sistem inferensi *fuzzy* yang dibangun. Namun prototipe yang dibangun Uri Kartoun membutuhkan akhir nilai berupa nilai *fuzzy* seperti *single* (representasi satu orang) maupun *couple* (representasi dua orang). Maka setelah mendapat nilai *COG* hasil perhitungan di atas, nilai tersebut akan dievaluasi kembali pada berbagai fungsi keanggotaan yang terdapat pada output *fuzzy* yang telah ditentukan sebelumnya dengan cara mencari nilai yang maksimum. Sebagai contoh, jika mengacu pada tiga fungsi keanggotaan (C1, C2, dan C3) pada Gambar 11, maka nilai *COG* yang didapat akan dievaluasi pada ketiga fungsi keanggotaan tersebut dan dicari nilai yang paling maksimum. Jika nilai tersebut tepat berada pada titik potong antara kedua fungsi keanggotaan, maka penulis memutuskan untuk memilih salah satu saja, dalam hal ini fungsi keanggotaan yang lebih dahulu (sebelah kiri).

2.7 SEKILAS MENGENAI 3 FASE PENGEMBANGAN

Karena pengerjaan Tugas Akhir ini berfokus kepada 3 fase pengembangan yang terdapat pada paper *A Prototype Fuzzy System for Surveillance Picture Understanding* yang ditulis oleh Uri Kartoun [2], maka pembelajaran mengenai 3 fase tersebut pun dilakukan. Sehingga nantinya pengembangan Prototipe Sistem Pengawasan Pergerakan Sederhana Manusia Menggunakan Sistem Inferensi Fuzzy ini juga mengikuti sebagian besar isi yang tertera pada *paper* tersebut. Isi yang dimaksud mencakup pembuatan sistem inferensi *fuzzy* yang diperlukan. Dimana nantinya akan terdapat beberapa perbedaan dari prototipe sejenis yang dikembangkan oleh Uri Kartoun tersebut.

3 fase pengembangannya sendiri terdiri dari fase *Image Preprocessing* dan dua fase lainnya yang berkaitan dengan pembentukan sistem inferensi *fuzzy*. Namun, pada *paper* yang ditulis oleh Uri Kartoun [2] tidak dibahas secara rinci mengenai metode yang digunakan untuk fase pertama tersebut, sehingga penulis memutuskan untuk memilih sendiri metode yang dipakai. Dan pada Tugas Akhir ini, penulis lebih menekankan pada sistem inferensi *fuzzy* yang dibangun pada kedua fase tersebut.

Selanjutnya, ketiga fase pengembangan tersebut akan dijelaskan lebih rinci pada bab-bab berikutnya. Baik dalam bentuk pembuatan modul dalam perancangan sistem (Bab 3) maupun secara terpisah berdasarkan proses pengerjaannya (Bab 4, 5, dan 6).