

Pengendalian Alat Penghapus Whiteboard Menggunakan Sistem *Fuzzy Logic*

Rafiuddin Syam¹ dan Dedy Harianto^{2*}

¹ Jurusan Mesin Fakultas Teknik Universitas Hasanuddin Makassar-Indonesia

² Akedemii Teknik Industri Makassar Jl. Sunu no 220 Makassar Sulawesi Selatan, Indonesia

¹rafiuddinsyam@gmail.com,²dedy_mits@yahoo.com

Abstrak

Menghapus *whiteboard* dengan cara manual memerlukan waktu yang dapat mengurangi keefisienan waktu kegiatan belajar mengajar serta beresiko membuat tangan pengajar menjadi kotor. Untuk itu dibuat suatu alat mekanik yang dapat bekerja secara otomatis dan terprogram sehingga menggantikan tugas pengajar dalam menghapus tulisan di *whiteboard*. Pada penelitian ini prototipe alat penghapus *whiteboard* menggunakan dua buah motor DC dengan kapasitas 12 volt yang dihubungkan dengan motor driver dan arduino uno yang telah diprogram untuk mengontrol pencekam penghapus dengan gerakan ke kanandan ke kiri. Pengontrolan arduino uno untuk mengendalikan gerakan motor pada alat penghapus *whiteboard* ini digambarkan dengan menggunakan kendali logika *fuzzy* (FLC) metode mamdani dan *defuzzyfikasi* menggunakan *Centroid of Area (COA)*. Teknik kendali *fuzzy* alat penghapus *whiteboard* disimulasikan dengan menggunakan perangkat lunak Matlab untuk menguji unjuk kerjanya.

Kata kunci : penghapus *whiteboard*, puli dan tali, Motor driver, Pengendalian *fuzzy*, Matlab.

Pendahuluan

Media tulis yang selalu ada di kantor atau juga sekolah/institusi pendidikan adalah *whiteboard* atau papan putih. Media ini memudahkan dalam penyampaian materi atau informasi, baik dalam rapat di kantor atau kegiatan belajar mengajar di sekolah/institusi pendidikan. Papan tulis jenis ini telah dikenal dikalangan masyarakat, baik dari segi pemasaran ataupun penggunaannya. Sesuai dengan namanya, papan tersebut berwarna putih dan menggunakan spidol sebagai alat tulisnya [1].

Untuk menghapus papan tulis ini menggunakan penghapus yang dilakukan secara manual, yaitu dengan tenaga manusia. [2]. Hal tersebut memerlukan waktu yang dapat mengurangi keefisienan waktu kegiatan belajar mengajar serta beresiko membuat tangan pengajar menjadi kotor.

Untuk mengatasi permasalahan tersebut, maka diperlukan suatu alat mekanik yang dapat bekerja secara otomatis dan terprogram, sehingga menggantikan tugas pengajar dalam menghapus papan tulis.



Gambar 1. Menghapus *whiteboard* secara manual

Sistem yang dibuat merupakan suatu prototipe yang memiliki bentuk seperti lengan pembawa penghapus untuk membersihkan papan tulis/*whiteboard* dan akan digabungkan dengan suatu alat mekanik pendukung lain yang berfungsi sebagai pengontrol gerakan penghapus ke kiri dan ke kanan, serta pengontrol gerakan penghapus ke atas dan kebawah sehingga untuk menggerakkannya, hanya dilakukan dengan menekan tombol arah yang terpasang [2].

Alat penghapus papan tulis/*whiteboard* otomatis telah dibuat oleh [3]. Pada saat pengujian penghapus tidak dapat menghilangkan sepenuhnya tulisan di papan tulis, karena tekanan penghapus terhadap papan tulis tidak maksimal. Penggunaan battery 12 V untuk mensuplay 2 motor DC (masing masing 12 V) dan Atmega8 menyebabkan torsi motor dalam menggerakkan penghapus tidak maksimal. Hal ini menyebabkan solusi permasalahan yang dapat dilakukan hanyalah pada penggantian kain bahan penghapus dan tidak pada mekanis yang dapat meningkatkan tekanan penghapus pada permukaan [3].

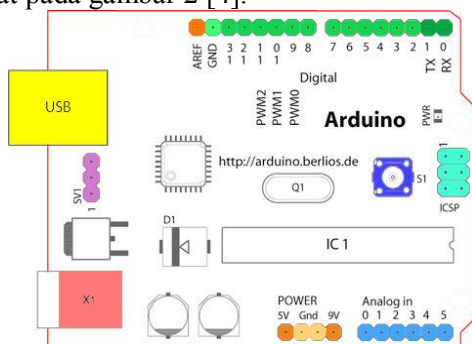
Penelitian ini akan disusun sebagai berikut: pada bagian pertama akan membahas pendahuluan, pada bagian kedua akan digambarkan tinjauan pustaka alat penghapus *whiteboard*. Selanjutnya kendali logika *fuzzy* akan dibahas pada bagian 3. Sedangkan pada bagian 4 akan dibahas kesimpulan.

Tinjauan Pustaka

Alat penghapus *whiteboard*. Secara sederhana alat penghapus *whiteboard* terdiri dari beberapa komponen penting yaitu penghapus, rel pengarah, puli dan tali, dua buah motor DC dan unit pengendali yang berupa arduino uno dan motor driver.

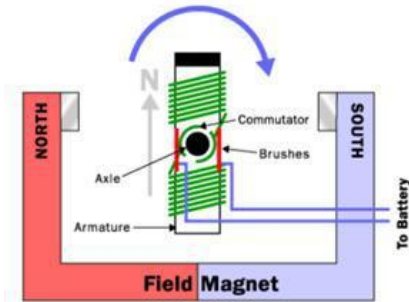
Cara kerja dari alat penghapus *whiteboard* adalah ketika *power* dinyalakan maka unit pengendali akan aktif. Mikrokontroler yang terdapat pada arduino uno akan memerintahkan motor DC 1 dan motor DC 2 untuk berputar menggerakkan puli yang dihubungkan dengan pencekam penghapus yang bergerak secara horisontal ke kanan. Apabila pencekam penghapus menyentuh *limit switch*, mikrokontroler yang terdapat pada arduino uno akan memerintahkan motor DC 1 dan motor DC 2 untuk berputar berlawanan arah dari putaran yang pertama sehingga pencekam penghapus akan bergerak secara horisontal ke kiri menghapus tulisan yang ada di *whiteboard*. Kecepatan menghapus *whiteboard* ini dipengaruhi oleh besarnya kecepatan putaran motor DC.

Arduino Uno. Arduino dikatakan sebagai sebuah *platform* dari *physical computing* yang bersifat *open source*. Pertama-tama perlu dipahami bahwa kata “*platform*” di sini adalah sebuah pilihan kata yang tepat. Arduino tidak hanya sekedar sebuah alat pengembangan, tetapi ia adalah kombinasi dari *hardware*, bahasa pemrograman dan *Integrated Development Environment (IDE)* yang canggih. IDE adalah sebuah *software* yang sangat berperan untuk menulis program, meng-*compile* menjadi kode biner dan meng-upload ke dalam *memory microcontroller*. Ada banyak proyek dan alat-alat dikembangkan oleh akademisi dan profesional dengan menggunakan Arduino, selain itu juga ada banyak modul-modul pendukung (sensor, tampilan, penggerak dan sebagainya) yang dibuat oleh pihak lain untuk bisa disambungkan dengan Arduino. Arduino berevolusi menjadi sebuah platform karena ia menjadi pilihan dan acuan bagi banyak praktisi. Papan arduino tipe usb dapat dilihat pada gambar 2 [4].



Gambar 2. Papan arduino tipe USB

Motor DC. Motor DC adalah piranti elektronik yang mengubah energi listrik menjadi energi mekanik berupa gerak rotasi. Pada motor DC terdapat jangkar dengan satu atau lebih kumparan terpisah. Tiap kumparan berujung pada cincin belah (komutator). Dengan adanya insulator antara komutator, cincin belah dapat berperan sebagai saklar kutub ganda (*double pole, double throw switch*). Mekanisme ini diperlihatkan pada gambar berikut ini [5].



Gambar 3. Bagan mekanisme kerja motor DC magnet permanen

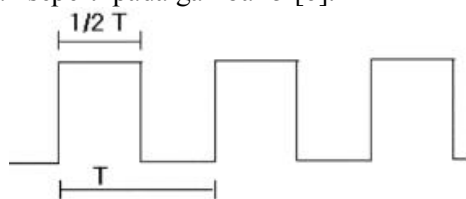
Motor DC yang digunakan pada robot beroda umumnya adalah motor DC dengan magnet permanen. Motor DC jenis ini memiliki dua buah magnet permanen sehingga timbul medan magnet di antara kedua magnet tersebut. Di dalam medan magnet inilah jangkar/rotor berputar. Jangkar yang terletak di tengah motor memiliki jumlah kutub yang ganjil dan pada setiap kutubnya terdapat lilitan. Lilitan ini terhubung ke area kontak yang disebut komutator. Sikat (*brushes*) yang terhubung ke kutub positif dan negatif motor memberikan daya ke lilitan sedemikian rupa sehingga kutub yang satu akan ditolak oleh magnet permanen yang berada di dekatnya, sedangkan lilitan lain akan ditarik ke magnet permanen yang lain sehingga menyebabkan jangkar berputar. Ketika jangkar berputar, komutator mengubah lilitan yang mendapat pengaruh polaritas medan magnet sehingga jangkar akan terus berputar selama kutub positif dan negatif motor diberi daya. Kecepatan putar motor DC (N) adalah pembagian antara tegangan terminal (V_{TM}) dikurangi perkalian arus jangkar motor dan hambatan jangkar motor dengan persamaan berikut[5].

$$N = \frac{V_{TM} - I_A R_A}{K\phi} \quad (1)$$

Pengendalian kecepatan putar motor DC dapat dilakukan dengan mengatur besar tegangan terminal motor V_{TM} . Metode lain yang biasa digunakan untuk mengendalikan kecepatan motor DC adalah dengan teknik modulasi lebar pulsa

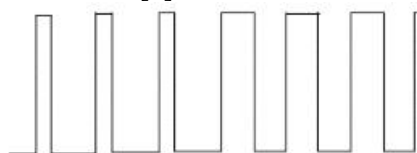
atau *Pulse Width Modulation* (PWM) [5].

PWM (*Pulse Width Modulation*) adalah salah satu teknik untuk mengatur kecepatan motor DC yang umum digunakan. Dengan menggunakan PWM kita dapat mengatur kecepatan yang diinginkan dengan mudah. Teknik PWM untuk pengaturan kecepatan motor adalah, pengaturan kecepatan motor dengan cara merubah-ubah besarnya *duty cycle* pulsa. Pulsa yang berubah ubah *duty cycle*-nya inilah yang menentukan kecepatan motor. Besarnya amplitudo dan frekuensi pulsa adalah tetap, sedangkan besarnya *duty cycle* berubah-ubah sesuai dengan kecepatan yang diinginkan, semakin besar *duty cycle* maka semakin cepat pula kecepatan motor, dan sebaliknya semakin kecil *duty cycle* maka semakin pelan pula kecepatan motor. Sebagai contoh bentuk pulsa yang dikirimkan adalah seperti pada gambar 4, pulsa kotak dengan *duty cycle* pulsa 50%. Sedangkan sebagai contoh bentuk pulsa PWM adalah seperti pada gambar 5 [6].



Gambar 4. Pulsa dengan *duty cycle* 50%

Seperti pada gambar 7, semakin besar *duty cycle* pulsa kotak, maka semakin lama pula posisi logika *high*. Jika motor diatur agar berjalan ketika diberi logika *high*, maka jika memberi pulsa seperti pada gambar 2 diatas, maka motor akan berada pada kondisi “nyala-mati-nyala-mati” sesuai dengan bentuk pulsa tersebut. Semakin lama motor berada pada kondisi “nyala” maka semakin cepat pula kecepatan motor tersebut. Motor akan berputar dengan kecepatan maksimum jika mendapat pulsa dengan *duty cycle* 100%. Dengan kata lain motor mendapat logika *high* terus menerus [6].



Gambar 5. Pulsa PWM

Dengan mengatur besarnya *duty cycle* pulsa kotak yang dikirimkan, kita dapat mengatur banyaknya logika *high* yang diberikan pada motor, dengan kata lain mengatur lamanya waktu motor untuk berputar dalam satu periode pulsa. Jika lamanya waktu motor untuk berputar dalam satu periode pulsa ini berubah maka kecepatan putaran motor juga akan berubah, sesuai dengan

duty cycle atau waktu motor untuk berputar dalam satu periode pulsa [6].

Kendali Logika Fuzzy

Logika *fuzzy* adalah suatu cara yang tepat untuk memetakan suatu ruang *input* kedalam suatu ruang *output*. Titik awal dari konsep modern mengenai ketidakpastian adalah paper yang dibuat oleh Lofti A Zadeh (1965), dimana Zadeh memperkenalkan teori yang memiliki obyek-obyek dari himpunan *fuzzy* yang memiliki batasan yang tidak presisi dan keanggotaan dalam himpunan *fuzzy*, dan bukan dalam bentuk logika benar (*true*) atau salah (*false*), tapi dinyatakan dalam derajat (*degree*). Konsep seperti ini disebut dengan *Fuzziness* dan teorinya dinamakan *Fuzzy Set Theory*. *Fuzziness* dapat didefinisikan sebagai logika kabur berkenaan dengan semantik dari suatu kejadian, fenomena atau pernyataan itu sendiri. Seringkali ditemui dalam pernyataan yang dibuat oleh seseorang, evaluasi dan suatu pengambilan keputusan[7].

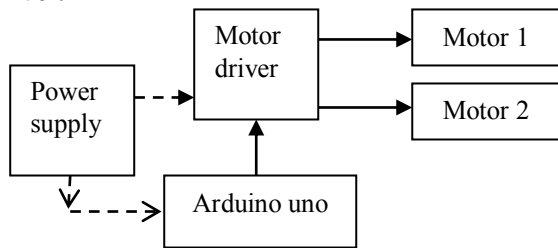
Fuzzy system (sistem kabur) didasari atas konsep himpunan kabur yang memetakan domain *input* kedalam domain *output*. Perbedaan mendasar himpunan tegas dengan himpunan kabur adalah nilai keluarannya. Himpunan tegas hanya memiliki dua nilai *output* yaitu nol atau satu, sedangkan himpunan kabur memiliki banyak nilai keluaran yang dikenal dengan nilai derajat keanggotaannya[7].

Logika *fuzzy* adalah peningkatan dari logika *Boolean* yang berhadapan dengan konsep kebenaran sebagian. Dimana logika klasik (*crisp*) menyatakan bahwa segala hal dapat diekspresikan dalam istilah *binary* (0 atau 1, hitam atau putih, ya atau tidak). Logika *fuzzy* menggantikan kebenaran *Boolean* dengan tingkat kebenaran. Logika *fuzzy* memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk *linguistic*, konsep tidak pasti seperti “sedikit”, “lumayan”, dan “sangat”. Logika ini diperkenalkan oleh Dr. Lotfi Zadeh dari Universitas California, Barkeley pada tahun 1965. Logika *fuzzy* telah digunakan pada bidang-bidang seperti taksonomi, topologi, linguistik, teori automata, teori pengendalian, psikologi, *pattern recognition*, *pengobatan*, *hukum*, *decision analysis*, *system theory and information retrieval*. Pendekatan *fuzzy* memiliki kelebihan pada hasil yang terkait dengan sifat kognitif manusia, khususnya pada situasi yang melibatkan pembentukan konsep, pengenalan pola, dan pengambilan keputusan dalam lingkungan yang tidak pasti atau tidak jelas[7].

Teori logika *fuzzy* yang diajukan oleh Zadeh pada pertengahan tahun 1960 (Nikola K, 1998

dari Setiyowati, M.I dan Seta, B.A, 2007), memberikan suatu pemecahan masalah terhadap persoalan yang tidak pasti ini. Sehingga sistem informasi yang akan dibuat menggunakan model DBMS dan *query* yang berbasis *fuzzy* karena model DBMS konvensional, *non fuzzy* kurang dapat memenuhi kebutuhan sistem informasi ini. Banyak model DBMS dan *query fuzzy* yang ada, salah satunya adalah model Tahani yang ditemukan pada tahun 1977. Prof. Lutfi Zadeh berpendapat bahwa logika benar dan salah dari logika *boolean*/konvensional tidak dapat mengatasi masalah gradasi yang ada di dunia nyata. Untuk mengatasi masalah gradasi tersebut maka ia mengembangkan sebuah himpunan samar (*fuzzy*)[7].

Alat penghapus *whiteboard* memiliki satu sistem pergerakan yaitu sistem pergerakan horizontal yang digerakkan oleh motor 1 dan motor 2. Gerakan motor 1 dan motor 2 ini diprogram di dalam mikrokontroler yang terdapat pada arduino uno yang mengendalikan motor driver.



Gambar 6. Skema rangkaian motor

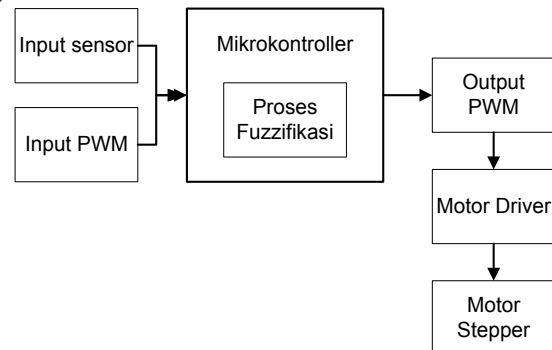
Sistem program pada arduino uno diatur dengan menggunakan 2 kontroler logika *fuzzy* (*Fuzzy Logic Controllers, FLC*). Gerakan X dari alat penghapus *whiteboard* akan didapatkan dari putaran motor 1 dan motor 2. Putaran motor 1 ini akan menggerakkan penghapus kearah X+ dan X-.

Kecepatan putaran motor DC diatur dengan menggunakan sistem *fuzzy logic*. Pengaturan putaran motor stepper ini adalah dengan merubah lebar sinyal tegangan PWM. PWM adalah singkatan dari *Pulse Width Modulation*, merupakan suatu metode yang digunakan untuk mengontrol daya yang berkaitan dengan *power supply*. Pembacaan metode *fuzzy* dimulai dengan menerima *input* dari sensor gaya yang ada di motor DC dimana nilainya akan dihitung untuk kemudian hasil dari keluarannya akan dikonversi menjadi PWM untuk motor DC.

Dalam sistem ini sensor gaya berfungsi sebagai masukan, dimana sensor ini akan mendeteksi besarnya gaya gesek yang bekerja pada penghapus yang kemudian akan memberikan signal analog mode pwm (pulse width modulation) yang ada di arduino uno, yang selanjutnya akan dikirim ke

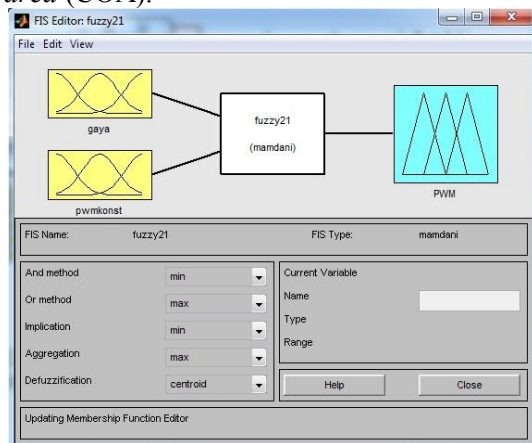
driver motor untuk menentukan kecepatan yang akan diberikan sinyal ke motor DC.

Diagram alir untuk menentukan pengaturan kecepatan motor dengan metode *fuzzy* pada gambar 7.



Gambar 7. Diagram alir proses *fuzzy* dalam pengaturan kecepatan motor

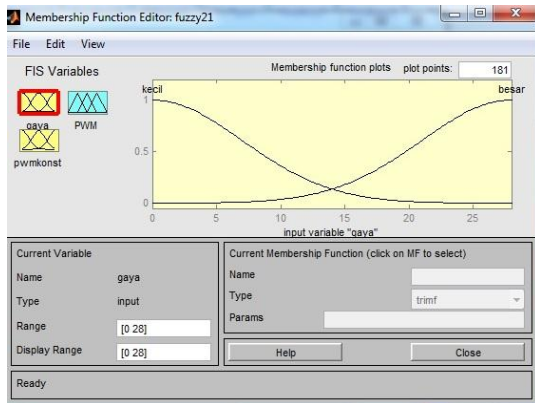
Dalam mengatur kecepatan putaran motor menggunakan metode *fuzzy* ini kondisi acuannya adalah pada kondisi *pwmkons* “sedang” dan kondisi gaya gesek sebenarnya, untuk mendapatkan hasil keluaran dari sensor gaya yang berupa nilai untuk *pwm* motor digunakan metode *mamdani* dan *defuzzifikasi* menggunakan *centroid of area* (COA).



Gambar 8. Setting parameter masukan dan keluaran pada editor *fuzzy*

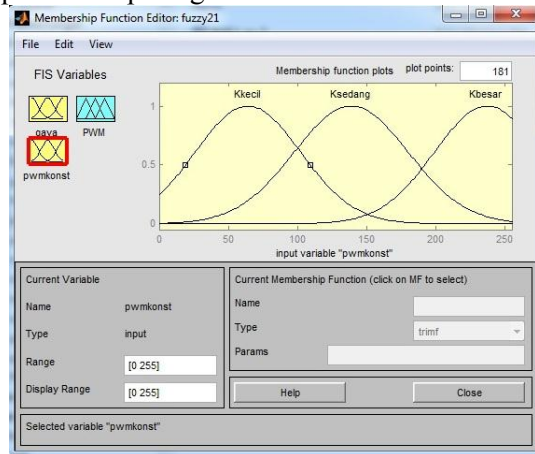
Dengan menggunakan *Fuzzy Inference System* (FIS) pada perangkat lunak Matlab, disetting dua masukan yaitu gaya gesek dengan kondisi “kecil” dan “besar” serta masukan *pwm konstan* dengan kondisi “Kkecil”, “Ksedang” dan “Kbesar” seperti yang terlihat pada gambar 8.

Masukan pertama yaitu gaya gesek dengan kondisi “kecil” dan “besar”, menggunakan *membership function* tipe *gaussmf* dapat dilihat pada gambar 9. Angka 0 – 28 menyatakan besarnya gaya gesek pada *whiteboard* oleh penghapus.



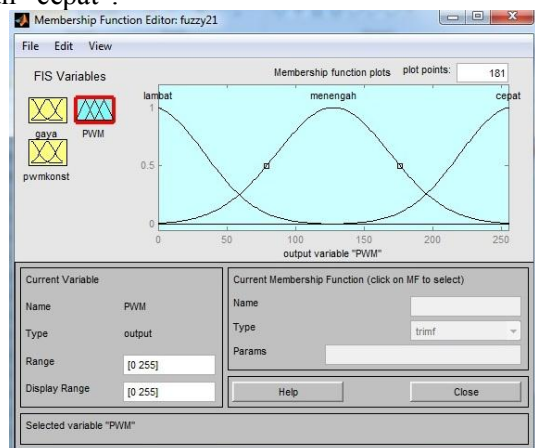
Gambar 9. Masukan gaya pada editor fuzzy

Masukan kedua yaitu pwm konstan dengan kondisi “Kkecil”, “Ksedang” dan “Kbesar”, menggunakan *membership function* tipe *trimf* dapat dilihat pada gambar 10.



Gambar 10. Masukan pwmkons pada editor fuzzy

Untuk keluaran yang berupa PWM, menggunakan *membership function* tipe *trimf* dapat dilihat pada gambar 11. Kondisi keluaran PWM ini ada tiga yaitu “lambat”, “menengah” dan “cepat”.



Gambar 11. Keluaran pwm pada editor fuzzy

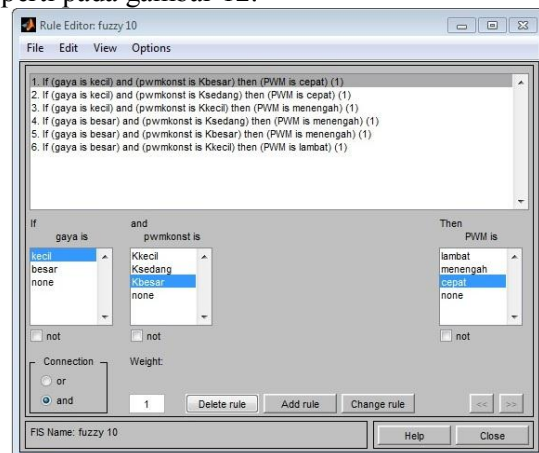
Keputusan yang diberikan oleh *fuzzy controller* berasal dari *rule-rule* yang ada pada basis data. Keputusan-keputusan ini disimpan sebagai

kumpulan *rule*. Dasarnya *rule-rule* tersebut adalah sebuah *rule if-then* yang intuitif dan mudah dimengerti, karena hanya merupakan kata-kata. Ada 6 rule yang digunakan dalam pengaturan kecepatan putaran motor pada alat penghapus whiteboard ini.

Tabel 1. Aturan Dasar (*Base Rules*)

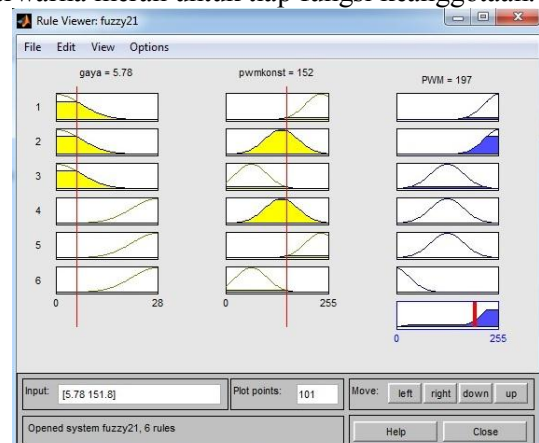
Motor	gaya	pwmkons	pwm
Motor 1 dan Motor 2	kecil	besar	cepat
	kecil	sedang	cepat
	kecil	kecil	menengah
	besar	Kbesar	menengah
	besar	Ksedang	menengah
	besar	Kkecil	lambat

Rule ini dimasukkan ke dalam “*rule editor*” seperti pada gambar 12.



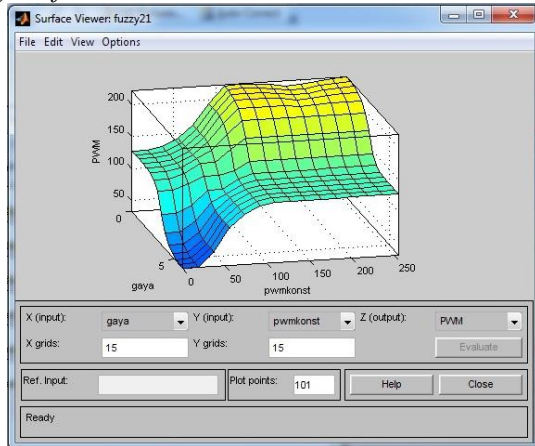
Gambar 12. Setting rule pada editor fuzzy

Rule yang telah dimasukkan pada *rule editor* dapat dilihat pada opsi *rule viewer* seperti terlihat pada gambar 13. *Rule viewer* ini digunakan untuk mengatur kecenderungan apakah kecepatan putaran motor adalah cepat atau lambat, yaitu tergantung dari besar kecilnya nilai pwm pada keluaran. Untuk mengatur kecenderungannya dapat dilakukan dengan menggeser garis tengah berwarna merah untuk tiap fungsi keanggotaan.



Gambar 13. Rule viewer pada editor fuzzy

Hasil dari *rule viewer* ini bisa dilihat dalam bentuk gambar 3 dimensi yang bernama *surface viewer*. Gambar *surface viewer* dapat dilihat pada gambar 14. *Surface viewer* ini merupakan plot grafik dari data-data yang dikeluarkan pada proses *defuzzifikasi*.



Gambar 14. *Surface viewer* pada editor fuzzy

Untuk menguji *fuzzy* yang telah dibuat, caranya adalah kembali ke *command window*, kemudian ketik: **fis=readfis('fuzzy21')** lalu enter maka matlab akan meload engine FIS yang telah dibuat. Engine FIS yang diload oleh Matlab dapat dilihat sebagai berikut

```
>>fis=readfis('fuzzy21')
fis =
    name: 'fuzzy21'
    type: 'mamdani'
    andMethod: 'min'
    orMethod: 'max'
    defuzzMethod: 'centroid'
    impMethod: 'min'
    aggMethod: 'max'
    input: [1x2 struct]
    output: [1x1 struct]
    rule: [1x6 struct]
```

File engine FIS yang dibuat dapat dilihat pada command editor seperti dibawah ini

```
[System]
Name='fuzzy21'
Type='mamdani'
Version=2.0
NumInputs=2
NumOutputs=1
NumRules=6
AndMethod='min'
OrMethod='max'
ImpMethod='min'
AggMethod='max'
DefuzzMethod='centroid'
```

```
[Input1]
Name='gaya'
```

```
Range=[0 28]
NumMFs=2
MF1='kecil':gaussmf,[7 0]
MF2='besar':gaussmf,[7 28]
```

```
[Input2]
Name='pwmkonst'
Range=[0 255]
NumMFs=3
MF1='Kkecil':gaussmf,[38.3 64.11]
MF2='Ksedang':gaussmf,[41.4 138.76]
MF3='Kbesar':gaussmf,[38.3 237.047]
```

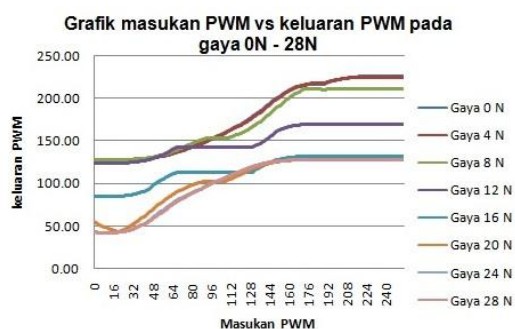
```
[Output1]
Name='PWM'
Range=[0 255]
NumMFs=3
MF1='lambat':gaussmf,[38.3 -4.49]
MF2='menengah':gaussmf,[41.4 127.5]
MF3='cepat':gaussmf,[38.3 258.09]
```

```
[Rules]
1 3, 3 (1) : 1
1 2, 3 (1) : 1
1 1, 2 (1) : 1
2 2, 2 (1) : 1
2 3, 2 (1) : 1
2 1, 1 (1) : 1
```

Hasil *defuzzifikasi* bisa diuji dengan memasukkan *rule* yang telah dibuat dengan mengetik instruksi *evalfis*.

```
>> out = evalfis([1 1],fis)
out =
    127.5000
....
....
>> out = evalfis([1 255],fis)
out =
    225.5750
....
....
>> out = evalfis([28 1],fis)
out =
    43.1637
....
....
>> out = evalfis([15 255],fis)
out =
    127.5000
```

Data dari *evalfis* ini sebenarnya adalah data yang digunakan untuk membuat plot pada *surface viewer* dalam bentuk 3 dimensi. Grafik masukan PWM dan keluaran PWM pada gaya 0 N sampai dengan 28 N dalam bentuk 2 dimensi dapat dilihat pada gambar berikut ini.



Gambar 15. Grafik masukan PWM dan keluaran PWM pada gaya 0N-28N

Pada gambar diatas, ketika nilai gaya 28N, nilai keluaran PWM cenderung kecil dibandingkan dengan gaya-gaya yang lebih kecil nilainya. Nilai keluaran PWM yang paling besar diperoleh dari kombinasi gaya yang kecil (0N) dan masukan PWM yang besar (255).

Kesimpulan

Sistem kendali dengan menggunakan kendali logika fuzzy dapat diterapkan untuk pengontrolan gerakan motor pada mikrokontroler yang terdapat pada arduino uno. Pada motor driver menggunakan sistem *aktiv low*, sehingga apabila diberi masukan *low* pada program, maka motor driver akan aktif. Gerakan sumbu X dari alat penghapus *whiteboard* didapatkan dari putaran motor 1 dan motor 2 yang akan menggerakkan penghapus kearah X+ dan X-. Dengan kendali logika fuzzy ini dapat diatur kecepatan putaran motor yang diinginkan dalam mencapai suatu posisi yang diharapkan. Dengan demikian kecepatan motor DC dalam merespon suatu perintah dapat pula diatur dengan menggunakan kendali logika *fuzzy*.

Referensi

- [1] Anonim, Papan Tulis (white board) (<http://www.sentraoffice.com/papan-tulis-whiteboard> . diakses pada 9 November 2013)
- [2] Supardi. Penghapus Papan Tulis Otomatis Berbasis Mikrokontroller ATmega 16 [Tesis]. Universitas Negeri Yogyakarta. Yogyakarta (2013)
- [3] Farhan Yasyirli Amri, Mulia Kausar, Henry Manurung, Aditya Effendi, Dimas Dayyuna Kusuma. Penghapus Papan Tulis Kaca Otomatis. (http://wiryadinata.web.id/?page_id=156 diakses pada 9 November 2013). Jurusan Teknik Elektro, Universitas Sultan Ageng Tirtayasa. Cilegon.
- [4] Feri Djuandi. Pengenalan Arduino. www.tobuku.com/docs/Arduino-Pengenalan.pdf? (2011)

- [5] Fahmizal, Driver Motor DC pada Robot Beroda dengan Konfigurasi H-BRIDGE MOSFET(<http://fahmizaleeits.wordpress.com/2011/12/04/driver-motor-dc-pada-robot-beroda-dengan-konfigurasi-h-bridge-mosfet/> diakses pada 10 Februari 2014)
- [6] Anonim, Pengaturan Kecepatan Motor Dengan PC oleh DST-52 (<http://delta-electronic.com/article/wp-content/uploads/2008/09/an0082.pdf> diakses pada 6 April 2014)
- [7] Anonim, Chapter II Tinjauan Teoritis (<http://repository.usu.ac.id/bitstream/123456789/25178/4/Chapter%20II.pdf> diakses pada 29 Maret 2014)