

Algoritma Greedy pada Robot Penjaga Gawang

Nurwanto 13511085

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

13511085@std.stei.itb.ac.id

Abstrak—Robot merupakan suatu bentuk teknologi yang dapat mempermudah pekerjaan manusia dalam kehidupan sehari-hari. Pada makalah ini akan dijelaskan bagaimana robot dapat menggantikan peran manusia sebagai penjaga gawang dalam suatu permainan sepak bola. Strategi algoritma yang digunakan pada robot penjaga gawang tersebut menggunakan algoritma greedy untuk menentukan keputusan. Algoritma greedy merupakan algoritma yang menentukan solusi tahap demi tahap dan memilih keputusan yang paling optimal. Robot akan menentukan jarak bola dan menentukan arah bola berdasarkan perpindahan posisi bola. Pada makalah ini akan dibandingkan bagaimana kinerja robot dalam melakukan perannya sebagai penjaga gawang dengan menggunakan algoritma greedy berdasarkan jarak dan berdasarkan sudut pandangan robot terhadap bola.

Kata kunci — Robot, algoritma greedy, sensor, trigonometri, kontroler, aktuator.

I. PENDAHULUAN

Dalam perkembangan teknologi bidang robotika, robot dapat melakukan berbagai macam peran dalam kehidupan sehari-hari. Peran tersebut dapat kita jumpai dalam bidang akademik, hiburan, olahraga, maupun dalam bidang industri. Di era yang modern ini, robot dapat membantu pekerjaan manusia, seperti diterapkan sebagai robot pembantu rumah tangga. Robot dapat digunakan sebagai ajang adu kecerdasan dan sebagai hiburan bagi manusia. Salah satu peran robot yang memberikan hiburan dan berkembang dalam bidang olahraga yaitu robot sepak bola. Robot sepak bola sangat menarik perhatian bagi peneliti bidang robotika dan semakin banyaknya kompetisi tentang robot sepak bola tersebut. Dalam kenyataannya robot sepak bola tersebut masih belum optimal dalam melakukan perannya, sehingga kinerja dari robot ini masih belum sesuai dengan yang kita harapkan.

Dengan adanya masalah optimasi tersebut, dalam makalah ini akan dibahas bagaimana robot dapat berfungsi sebagai penjaga gawang dalam suatu permainan sepak bola dengan menggunakan algoritma greedy. Tujuannya robot mampu menghalau bola yang datang kearah gawangnya. Pada makalah ini, sensor robot yang digunakan yaitu hanya menggunakan sensor kamera sebagai pengganti indera mata pada manusia.

Sensor kamera tersebut digunakan untuk mengolah citra

gambar yang ditangkapnya. Kemampuan dasar yang harus ada dalam robot ini yaitu mendeteksi bola dengan menggunakan kamera. Selain itu, robot harus mampu menentukan jarak bola dari posisi robot berada. Jika robot tidak menemukan objek bola, maka robot akan melakukan pencarian bola sampai ditemui objek bola. Setelah robot berhasil menemukan objek bola, maka robot akan melakukan perhitungan jarak dengan memanfaatkan rumus trigonometri. Nilai jarak pertama yang didapatkan akan menjadi jarak inisial yang akan dijadikan sebagai pembanding untuk jarak bola selanjutnya. Hasil perbandingan jarak inisial dengan jarak sekarang maka akan diperoleh data perpindahan bola. Hasil gambar yang diperoleh tersebut akan dibagi menjadi beberapa segmen yang berguna untuk mempermudah dalam menentukan arah perpindahan bola. Data jarak dan arah perpindahan bola tersebut akan digunakan untuk menentukan keputusan bagi robot penjaga gawang dalam melakukan aksi. Terdapat empat status keputusan pada robot tersebut yang meliputi, tangkap bola ke kiri, tangkap bola ke kanan, tangkap bola di tengah, dan posisi diam.

II. DASAR TEORI

A. Robot dan Perkembangannya dalam Sepak Bola.

Robot merupakan suatu alat yang dapat melakukan tugas fisik, baik dijalankan secara otomatis oleh program tertentu maupun dengan kontrol manual oleh manusia. Istilah robot berawal bahasa Cheko “*robota*” yang mempunyai arti pekerja atau kuli yang dapat melakukan pekerjaan secara terus menerus tanpa mengurangi performansinya. Di era yang modern ini, robot banyak diterapkan dalam bidang industri. Penerapan robot dapat kita jumpai dalam *search and rescue*, serta untuk deteksi bom.

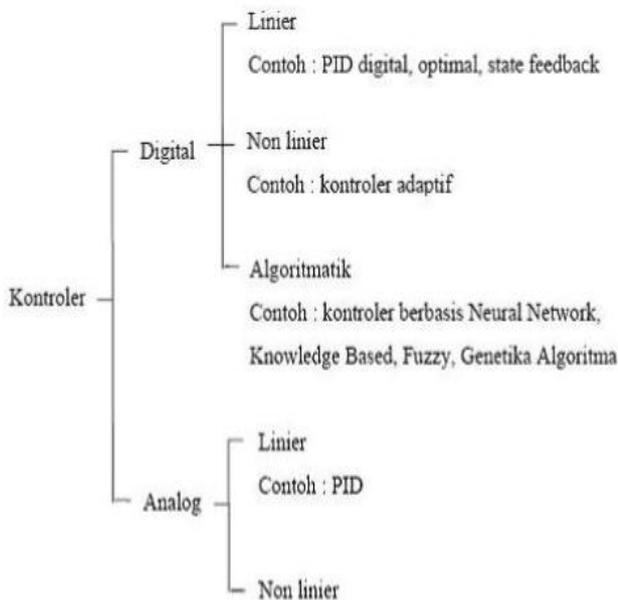
Komponen dasar penyusun robot yaitu manipulator, sensor, aktuator, dan kontroler. Manipulator merupakan perangkat yang digunakan untuk memanipulasi bahan tanpa kontak langsung. Sensor merupakan jenis transduser yang digunakan untuk mengubah besaran mekanis, magnetis, panas, sinar, dan kimia menjadi tegangan dan arus listrik. Transduser adalah alat yang berfungsi untuk mengubah

suatu energi tertentu kedalam bentuk energi lain. Terdapat dua jenis transduser, yaitu transduser aktif dan transduser pasif. Transduser aktif dapat menghasilkan energi listrik sendiri, contohnya foto sel. Sedangkan transduser pasif membutuhkan catu daya eksternal untuk dapat bekerja. contohnya thermistor.

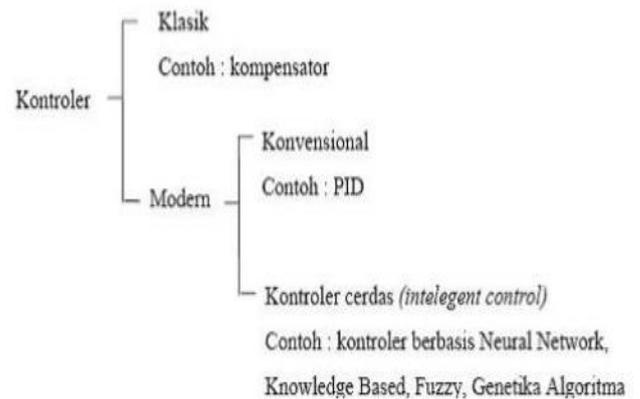
Beberapa jenis sensor yang sering digunakan antara lain sensor cahaya, sensor suhu, dan sensor tekanan.

Aktuator merupakan sebuah alat mekanik untuk menggerakkan atau mengontrol sebuah mekanisme atau sistem. Aktuator diaktifkan dengan menggunakan lengan mekanis yang biasanya digerakkan oleh motor listrik, yang dikendalikan oleh media pengontrol otomatis yang terprogram di antaranya mikrokontroler. Aktuator merupakan elemen yang mengkonversikan besaran listrik analog menjadi besaran lainnya misalnya kecepatan putaran dan merupakan perangkat elektromagnetik yang menghasilkan daya gerakan sehingga dapat menghasilkan gerakan pada robot. Jenis energi penggerak pada aktuator meliputi aktuator energi elektrik, aktuator energi hidrolis, aktuator energi *pneumatic*, aktuator energi *piezoelectric*, dan jenis lainnya. Keunggulan aktuator elektrik diantaranya pengontrolan yang mudah, mempunyai kecepatan yang cukup tinggi, mempunyai akurasi yang tinggi serta mempunyai efisiensi yang baik.

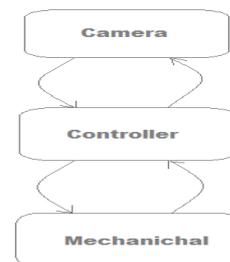
Kontroler merupakan salah satu komponen sistem pengaturan yang berfungsi mengolah sinyal umpan balik dan sinyal masukan acuan atau sinyal error menjadi sinyal kontrol. Sinyal error adalah selisih antara sinyal umpan balik yang dapat berupa sinyal keluaran sebenarnya atau sinyal keluaran terukur dengan sinyal masukan acuan. Berdasarkan aksi kontrolnya, kontroler dapat diklasifikasikan sebagai berikut :



Berdasarkan periodanya, kontroler dapat diklasifikasikan sebagai berikut :

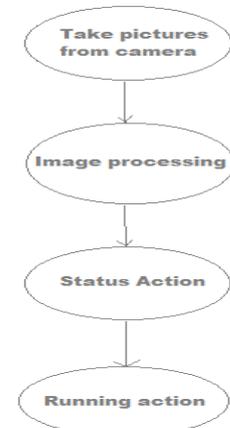


Dalam robot sepak bola, bentuk sistem koordinasi dasar dari robot tersebut dapat dijelaskan pada gambar flow dibawah ini.



Flow 1. Flow kerja sistem

Pada alir flow 1 tersebut dapat dilihat bahwa terdapat tiga komponen penting yang saling berhubungan. Sensor dalam hal ini kamera akan mengambil data berupa gambar yang kemudian akan diolah oleh controller. Dari controller tersebut akan mengirimkan status pada mekanik dalam hal ini motor. Untuk robot penjaga gawang maka proses statenya akan diperlihatkan pada flow diagram berikut.



Flow 2. Proses pengambilan keputusan

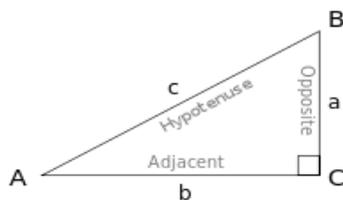
Dari flow 2 tersebut. Menjelaskan bagaimana robot akan bekerja. Pertama kali robot akan mengambil

gambar dengan kamera, lalu dari gambar tersebut akan diolah pada kontroler. Pengolahan gambar tersebut meliputi pendeteksian warna dengan pengelompokan *cluster* atau menjadi bagian-bagian tertentu sesuai dengan kesamaan warna. Sistem akan melakukan *scanning* gambar dan mencari warna atau bentuk yang dideteksi sebagai bola. Setelah mendapatkan posisi bola tersebut maka sistem akan memberikan status aksi tertentu yang telah didefinisikan sebelumnya. Lalu dari status aksi tersebut akan menjalankan aksi yang diberikan, dalam hal ini yang menjalankan aksi adalah bagian aktuator.

B. Trigonometri

Dalam ilmu matematika, teorema trigonometri sudah tidak asing lagi bagi kita. Hal ini banyak sekali diterapkan pada kehidupan sehari-hari.

Dasar dari trigonometri adalah konsep kesebangunan segitiga siku-siku. Sisi-sisi yang bersesuaian pada dua bangun datar yang sebangun memiliki perbandingan yang sama. Pada geometri *euclid*, jika masing-masing sudut pada dua segitiga memiliki besar yang sama, maka kedua segitiga itu pasti sebangun. Hal ini adalah dasar untuk perbandingan trigonometri sudut lancip. Konsep ini lalu dikembangkan lagi untuk sudut-sudut non lancip (lebih dari 90 derajat dan kurang dari nol derajat).



Gambar 1. Segitiga siku-siku

Berikut beberapa fungsi dasar dalam trigonometri :

$$\sin A = \frac{a}{c}$$

$$\cos A = \frac{b}{c}$$

$$\tan A = \frac{\sin A}{\cos A} = \frac{a}{b}$$

$$\cot A = \frac{1}{\tan A} = \frac{\cos A}{\sin A} = \frac{b}{a}$$

$$\sec A = \frac{1}{\cos A} = \frac{c}{b}$$

$$\csc A = \frac{1}{\sin A} = \frac{c}{a}$$

C. Greedy Algorithm

Algoritma *greedy* merupakan algoritma yang memecahkan masalah secara langkah per langkah. Dari setiap langkah yang dilakukan tersebut, terdapat berbagai macam solusi yang dapat kita peroleh untuk mencapai tujuan. Pada strategi algoritma *greedy* ini akan mengambil suatu keputusan yang terbaik dalam setiap langkah pemecahan masalah. Solusi terbaik yang telah diambil tersebut tidak dapat dirubah, dalam hal ini solusi tersebut merupakan solusi final. Prinsip algoritma *greedy* yaitu "*take what you can get now*", yang berarti dalam algoritma ini akan mengambil pilihan yang terbaik yang dapat diperoleh pada saat itu tanpa memperhatikan konsekuensi ke depan. Hal tersebut dilakukan dengan harapan bahwa dengan memilih optimum lokal pada setiap langkah akan berakhir dengan optimum global.

Terdapat lima elemen dalam algoritma *greedy* sebagai berikut :

- Himpunan kandidat, C, berisi elemen-elemen pembentuk solusi. Pada setiap langkah *greedy*, satu element akan dipilih untuk mendapatkan optimum lokal.
- Himpunan solusi, S, berisi kandidat-kandidat yang terpilih sebagai solusi persoalan.
- Fungsi seleksi, memilih kandidat yang paling memungkinkan emncapai solusi optimal. Kandidat yang sudah dipilih pada suatu langkah tidak pernah dipertimbangkan lagi pada langkah berikutnya.
- Fungsi kelayakan, memeriksa apakah suatu kandidat yang telah dipilih dapat memberikan solusi yang layak , yakni kandidat tersebut bersama-sama dengan himpunan solusi yang sudah terbentuk tidak melanggar kendala (constrains) yang ada. Kandidat yang layak dimasukan kedalam himpunan solusi, sedangkan kandidat yang tidak layak dibuang dan tidak dipertimbangkan lagi untuk langkah selanjutnya.
- Fungsi obyektif, yaitu fungsi yang memaksimumkan atau meminimumkan nilai solusi(misalnya panjang lintasan, keuntungan, dan lain-lain).

Dengan kata lain, algoritma *greedy* melibatkan pencarian sebuah himpunan bagian S dari himpunan kandidat C. Dalam hal ini, S harus memenuhi beberapa kriteria yang ditentukan, yaitu menyatakan suatu solusi dan S dioptimisasi oleh fungsi obyektif. Algoritma *greedy* tidak selalu memberikan solusi optimum global. Alasannya yaitu *greedy* tidak beroperasi secara menyeluruh terhadap semua alternatif solusi yang ada. Selain itu, terdapat beberapa fungsi solusi yang berbeda, sehingga kita harus memilih fungsi yang tepat jika menginginkan hasil solusi yang optimal. Namun, algoritma ini mempunyai kecepatan eksekusi yang baik dalam menyelesaikan masalah.

Berikut skema umum algoritma greedy :

```

function greedy(input C: himpunan_kandidat)→
  himpunan_kandidat
{ Mengembalikan solusi dari persoalan optimasi
  dengan algoritma greedy
  Masukan: himpunan kandidat C
  Keluaran: himpunan solusi yang bertipe
  himpunan_kandidat
}
Deklarasi
x : kandidat
S : himpunan_kandidat
Algoritma:
S ← {} { inisialisasi S dengan kosong }
while (not SOLUSI(S) and (C ≠ {})) do
  x ← SELEKSI(C) { pilih sebuah
  kandidat dari C}
  C ← C - {x} { elemen himpunan
  kandidat berkurang satu }
  if LAYAK(S ∪ {x}) then
    S ← S ∪ {x}
  endif
endwhile
{SOLUSI(S) or C = {} }
if SOLUSI(S) then
  return S
else
  write('tidak ada solusi')
endif

```

III. METODE PENYELESAIAN MASALAH

Lingkungan uji coba yang digunakan pada penyelesaian masalah ini yaitu lapangan sepak bola robot. Lapangan robot tersebut berwarna hijau dengan bola berwarna orange. Dalam melakukan tugasnya sebagai penjaga gawang, beberapa langkah yang dilalui oleh robot dalam mencari bola sampai melakukan aksi atau tindakan, sebagai berikut :

- Mencari bola
- Menghitung jarak
- Mendeteksi arah bola
- Melakukan aksi menangkap atau menghalau bola

Berikut langkah-langkah secara detail :

1. Mencari bola

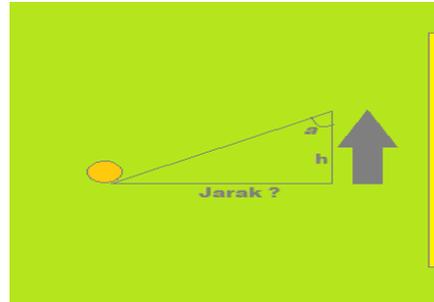
Robot akan mencari bola dengan mengenali bentuk dan warna bola. Pencarian bola akan terus berulang sampai bola dapat ditemukan.

2. Menentukan jarak bola

Setelah robot berhasil menemukan bola maka robot akan melakukan perhitungan jarak robot dengan bola tersebut. Dengan menggunakan konsep trigonometri pada segitiga maka didapatkan

suatu jarak. Dengan mengetahui sudut pada robot dan mengetahui tinggi kamera dari permukaan maka didapat jarak bola dari posisi robot tersebut berada.

Berikut cara menghitung jarak bola dari robot. Mula-mula tinggi kamera dari permukaan didefinisikan terlebih dahulu, data sudut pada robot dapat langsung diambil saat program sedang berjalan.



Gambar 2. Menentukan jarak

Dari gambar tersebut, robot direpresentasikan dengan anak panah berwarna abu-abu. Bola direpresenrasikan dengan bentuk lingkaran berwarna orange dan gawang berwarna kuning. Untuk menghitung jarak tersebut dapat dilakukan perhitungan dengan menggunakan konsep trigonometri. Berikut persamaan matematis yang digunakan pada kasus penentuan jarak tersebut.

$$\tan a = \text{Jarak} / h$$

Sehingga didapat,

$$\text{Jarak} = h \tan a$$

Contoh kasusnya yaitu misalkan tinggi kamera dari permukaan yaitu 50 cm, dan saat melihat bola sudut yang dibentuk oleh kamera dengan garis vertikal yaitu 50°, maka nilai jarak akan didapat dengan melakukan perhitungan sebagai berikut,

$$\text{Jarak} = h \tan a$$

$$\text{Jarak} = 50 \text{ cm} \cdot \tan 50$$

$$\text{Jarak} = 50 \times 1,192$$

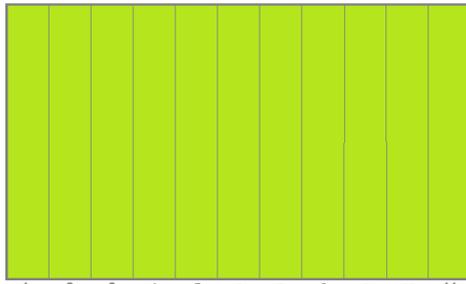
$$\text{Jarak} = 59,6 \text{ cm}$$

Dengan menggunakan persamaa tersebut diperoleh jarak dari posisi robot ke bola sebesar 59,6 cm.

3. Mendeteksi arah bola

Untuk mendeteksi arah bola, hal yang kita lakukan pertama kali yaitu membagi frame gambar menjadi beberapa segmen. Hal tersebut dilakukan untuk memudahkan dalam menentukan arah bola.

Pembagian segmen gambar tersebut dapat dilihat pada gambar berikut ini.



Gambar 3. Pembagian segmen gambar

Pada gambar 3 menunjukkan gambar yang diambil oleh kamera. Gambar dibagi-bagi menjadi beberapa segment, dalam gambar tersebut dibagi menjadi 11 segmen. Untuk memudahkan dalam menentukan segmen maka tiap segmen diberi indeks (pada gambar diberi indeks dari 1 sampai 11).

Setelah berhasil membuat segmen, maka sistem siap untuk melakukan pendeteksian bola dan menentukan arah bola. Aksi menangkap bola dilakukan ketika bola melakukan perpindahan mendekati gawang, jika bola menjauh maka robot hanya mempertahankan untuk tetap dapat mendeteksi bola. Untuk dapat berperan sebagai mana mestinya dalam mengambil keputusan sebagai penjaga gawang, maka dalam paper ini menggunakan algoritma Greedy yang akan menentukan keputusan berdasarkan bobot yang paling mungkin.

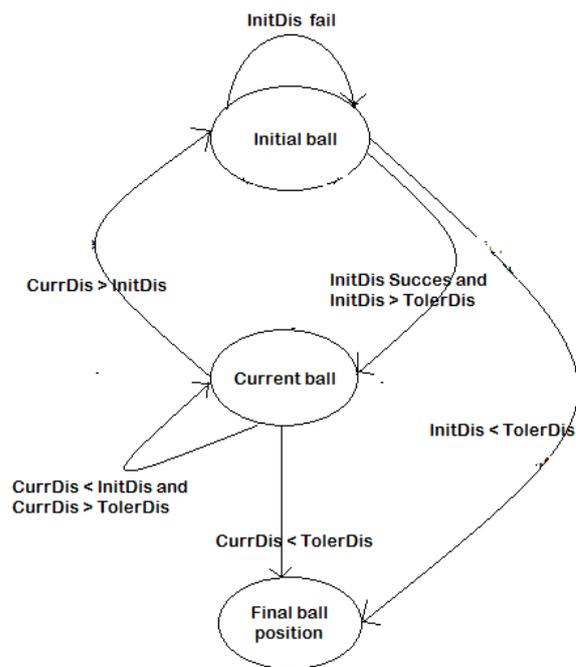
Dalam menyelesaikan masalah ini kita dapat menggunakan greedy berdasarkan jarak dan greedy berdasarkan sudut ang diberntuk oleh kamera.

a) Greedy by Jarak

Greedy by jarak merupakan suatu pengambilan keputusan berdasarkan posisi bola pada segmen dilayar pada setiap perubahan jarak yang ditentukan.

Pertama kali kita mendefinisikan jarak toleransi untuk melakukan suatu aksi. Jarak toleransi merupakan jarak dimana robot harus melakukan tindakan dan tidak mempertimbangkan faktor lainnya. Pada greed by jarak ini terdapat tiga istilah yaitu *initDis* (*Initial distance*), *CurrDis* (*Current Distance*), dan *TolerDis* (*Tolerance Distance*). *Initial distance* merupakan jarak awal yang disimpan

ketika sensor kamera mendeteksi bola dan menyimpan posisi bola pada indek segmen yang keberapa. *Current distance* merupakan jarak sekarang antara robot dengan bola. Pada flow 3 dibawah ini terdapat tiga *state* yaitu *initial ball*, *current ball*, dan *final ball position*. Dalam *initial ball* akan dilakukan pencarian objek bola. *Current ball* merupakan state yang mana robot telah berhasil menentukan *initial distance* dan melakukan *update* jarak bola dengan robot saat ini. Pada *final ball position* dilakukan suatu keputusan final yang harus diambil untuk melakukan aksi. Berikut akan dijelaskan bagaimana mekanisme kerja robot dalam mengambil keputusan.

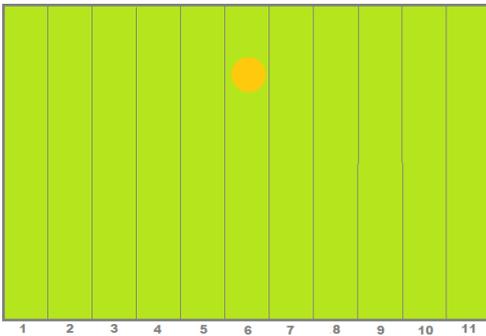


Flow 3. Diagram flow greedy by jarak

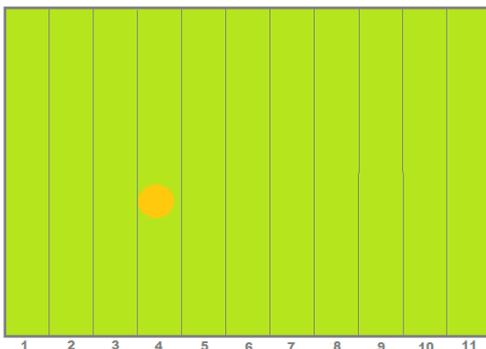
Pada flow tersebut, hal yang pertama kali dilakukan oleh robot yaitu melakukan pencarian bola. Jika bola berhasil terdeteksi maka dilakukan perhitungan jarak bola dari robot. Jarak tersebut akan dimasukkan dalam *InitDis*, lalu *initDis* tersebut dicek dengan *TolerDis* yang kita tentukan sebelumnya. Jika *InitDis* lebih besar dari *TolerDis* maka akan melanjutkan ke *state current ball*. Sedangkan jika *InitDis* kurang dari *TolerDis* maka akan melanjutkan ke *state "final ball position"*. Saat berada pada *current ball* maka akan dilakukan pengecekan segmen robot dan jarak saat ini yang akan dimasukkan kedalam *CurrDis*. Jika *CurrDis* kurang dari *TolerDis* yang berarti bola dekat dengan posisi robot berada maka akan melakukan perbandingan indek segmen pada *InitDis* dan segmen pada *CurrDis*. Jika indek segmen pada *CurrDis* lebih kecil dari segmen pada *initDis* maka robot akan melakukan aksi untuk

menangkap bola kearah kiri. Jika indeks segmen pada CurrDis lebih besar dari indek segmen pada InitDis maka robot akan melakukan aksi dengan menangkap bola kearah kanan. Sedangkan jika indek segmen posisi bola pada CurrDis sama dengan indeks segmen pada InitDis sama maka akan melakukan aksi untuk menangkap bola kedepan. Pada gambar 4 mengilustrasikan proses pengambilan keputusan, yang mana pada kasus tersebut robot melakukan aksi dengan mengambil keputusan menangkap bola kearah kiri.

Posisi bola di segmen 6 pada InitDis



Posisi bola di segmen 4 pada CurrDis



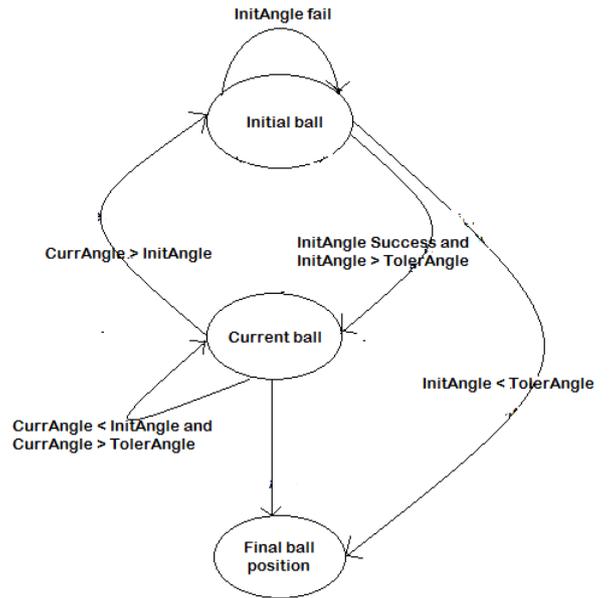
Gambar 4. Ilustrasi pengambilan keputusan

b) **Greedy by Sudut**

Greedy by sudut merupakan suatu pengambilan keputusan berdasarkan posisi bola pada segmen dilayar pada setiap perubahan sudut yang ditentukan.

Pertama kali mendefinisikan sudut toleransi untuk melakukan suatu aksi. Sudut toleransi merupakan sudut dimana robot harus melakukan tindakan dan tidak mempertimbangkan faktor lainnya. Pada greedy by sudut, terdapat tiga istilah yaitu *initAngle* (*Initial Angle*), *CurrAngle* (*Current Angle*), dan *TolerAngle* (*Tolerance Angle*). *Initial angle* merupakan sudut awal yang dibentuk oleh kamera terhadap garis vertikal yang disimpan ketika sensor kamera mendeteksi bola dan menyimpan posisi bola pada indek segmen yang keberapa. *Current distance* merupakan sudut sekarang antara robot dengan bola. Pada flow 4 dibawah ini terdapat tiga *state* yaitu *initial ball*, *current ball*, dan *final ball position*. Ketiga *state* tersebut fungsinya sama seperti pada greedy by jarak. Berikut akan dijelaskan bagaimana

mekanisme kerja robot dalam mengambil keputusan.



Flow 4. Alur greedy by Sudut

Pada alur flow 4 tersebut, saat program pertama kali *dirunning* maka akan mencari bola pada *state initial ball*. Selanjutnya sistem akan selalu mengupdate data indeks segmen dan sudut kamera selama *CurrAngle* lebih besar dari *TolerAngle*. Untuk proses selanjutnya sama seperti proses pada greedy b jarak yang membandingkan nilai jarak dan segmen. Namun, yang membedakan dalam greedy by sudut, nilai perbandingan yang digunakan yaitu sudut dan indeks segmen.

4. **Melakukan aksi**

Pada langkah ini, hasil processing dari kontroler akan mengirimkan *output* status gerakan. Yang kemudian akan diteruskan ke aktuator. Aktuator akan menjalankan aksi sesuai dengan status yang diberikan oleh kontroler. Status kontroler yang diberikan pada actuator meliputi status "on" atau "off" dan status lain yang akan memberikan *task* secara detail pada perangkat tertentu.

IV. ANALISIS PEMBAHASAN

Metode algoritma greedy by jarak dengan greedy by sudut mempunyai cara yang sama, namun mempunyai sudut pandang yang berbeda dalam menentukan keputusan. Dengan menggunakan greedy by jarak maka hasil akan lebih maksimal dan data jarak bisa kita gunakan untuk berbagai kebutuhan lainnya. Dengan menggunakan data jarak tersebut maka untuk proses *debugging* lebih mudah dan dapat digunakan pada robot lainnya. Namun greedy by jarak memiliki kekurangan yaitu jarak yang dihasilkan dari perhitungan kadang memiliki galat yang besar, hal ini tergantung dari

pengembangnya sendiri dalam menggunakan metode penentuan jarak.

Penggunaan algoritma greedy by sudut akan menentukan atau mengambil keputusan berdasarkan sudut yang dibentuk antara garis vertical dengan arah kamera. Hal ini memiliki keakuratan yang lebih baik dibandingkan dengan greedy by jarak. Pada greedy by jarak setelah mendapatkan sudut maka data dolah lagi menjadi jarak. Sedangkan pada greedy by sudut hanya dengan memanfaatkan sudut terhadap garis vertical saja untuk menentukan suatu keputusan. Penggunaan greedy by sudut ini dapat melakukan suatu aksi dengan lebih cepat dibandingkan dengan jarak, namun untuk penggunaan pada robot lain harus dilakukan *tuning* terlebih dahulu.

V. KESIMPULAN

Robot merupakan suatu teknologi yang dapat membantu manusia dalam kehidupan sehari-hari. Berbagai macam peran robot yang dapat kita temui diantaranya dalam bidang industri, militer, pendidikan, olahraga, maupun hiburan. Dalam perkembangannya robot dapat berperan sebagai pemain sepak bola. Selayaknya permainan sepak bola pada mestinya, robot terbagi menjadi robot penjaga gawang dan robot penyerang. Untuk robot penjaga gawang harus mampu mempertahankan gawangnya dari serangan lawan. Salah satu algoritma yang sesuai untuk diterapkan dalam robot penjaga gawang yaitu algoritma greedy. Pada makalah ini terdapat dua algoritma greedy, yaitu greedy by Jarak dan greedy by Sudut. Penggunaan greedy by jarak sangat baik untuk diterapkan terutama dalam beberapa robot. Selain itu, greedy by jarak selain untuk keperluan dalam mengambil keputusan untuk menangkap bola, dapat digunakan untuk kebutuhan lainnya. Sedangkan penggunaan greedy by sudut memiliki waktu komputasi yang lebih cepat dan akurat dibandingkan dengan greedy by jarak.

REFERENSI

- [1] Rinaldi Munir, Diktat Kuliah IF3051 Strategi Algoritma, Bandung: Program Studi Teknik Informatika ITB, 2009.
- [2] Craig, J.J. (2005). Introduction to Robotics. Pearson Prentice Hall. Upper Saddle River, NJ.
- [3] Brooks, Rodney. "Achieving Artificial Intelligence through Building Robots." Boston: Massachusetts Institute of Technology, 1986.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 19 Desember 2013



Nurwanto
13511085