

Penerapan Algoritma Greedy untuk Mendeteksi Bola pada Robot Sepak Bola Beroda

Muhammad Hilmi Asyrofi 13515083

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

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

13515083@std.stei.itb.ac.id

Abstract—Robot sepak bola merupakan robot yang mampu bergerak *autonomous*. Sebagai robot yang bergerak *autonomous*, diperlukan sistem persepsi yang berfungsi untuk mendeteksi bola. Algoritma greedy dapat digunakan sebagai dasar pembuatan algoritma deteksi bola karena terdapat berbagai tahapan pada pengolahan citra yang dapat dioptimasi. Oleh karena itu, dalam makalah ini akan dibahas algoritma untuk mendeteksi bola pada robot sepak bola beroda berbasis algoritma greedy. Himpunan kandidat berisi setiap *pixel* dari citra kamera. Himpunan solusi merupakan *pixel* yang terpilih dari himpunan kandidat berdasarkan *threshold* warna bola. Untuk mencapai solusi optimal, maka dipilih blob yang memiliki bentuk lingkaran. Kendala yang tidak boleh dilanggar adalah blob yang terpilih harus ada dalam lapangan. Jika dalam lapangan terdeteksi lebih dari satu blob, maka solusi yang dipilih adalah blob yang memiliki jari-jari terbesar. Berdasarkan skenario pengujian yang dilakukan, algoritma greedy berhasil diimplementasikan pada setiap skenario pengujian. Dengan demikian robot mampu menjalankan visinya yaitu bermain bola di lapangan.

Keywords—robot sepak bola; *autonomous*; persepsi; algoritma greedy

I. PENDAHULUAN

Kontes Robot Sepak Bola Indonesia (KRSBI) Beroda merupakan salah satu kompetisi untuk mahasiswa yang diadakan oleh Ristekdikti. KRSBI Beroda diadakan untuk meningkatkan keilmuan dan kreativitas mahasiswa di bidang robotika dan kecerdasan buatan. Dalam kontes ini, tim yang terdiri dari 5 buah robot *full autonomous* beroda bertanding melawan tim lain. Selama pertandingan berlangsung, tidak diperbolehkan adanya campur tangan manusia [1].

Sebagai robot yang harus mampu bergerak secara *autonomous*, diperlukan sistem persepsi yang berfungsi untuk mendeteksi objek yang berada di lapangan permainan sepak bola. Tanpa adanya sistem persepsi, robot tidak mungkin mampu menjalankan misinya bermain bola mengikuti aturan umum sepak bola. Dengan kata lain, keberadaan sistem persepsi pada robot sepak bola adalah syarat wajib bagi suatu *platform* robot sepak bola.

Sistem persepsi berbasis citra merupakan salah satu sistem persepsi yang banyak digunakan oleh robot sepak bola luar negeri. Dalam sistem persepsi berbasis citra, digunakan *omnidirectional* kamera yang mampu menangkap citra dari berbagai sudut sekaligus. Objek-objek yang harus dideteksi

oleh sistem persepsi berbasis citra adalah bola, lapangan, dan gawang.

Algoritma untuk mendeteksi bola merupakan algoritma yang paling krusial untuk diimplementasikan karena robot tidak mungkin bisa bermain tanpa informasi posisi bola. Namun algoritma deteksi bola masih kurang optimal karena dalam pertandingan terdapat berbagai *noise* yang mengganggu citra. Algoritma greedy dapat digunakan sebagai dasar pembuatan algoritma deteksi bola karena terdapat berbagai tahapan pada pengolahan citra yang dapat dioptimasi. Oleh karena itu, dalam makalah ini akan dibahas algoritma untuk mendeteksi bola pada robot sepak bola beroda berbasis algoritma greedy sebagai optimasi.



Gambar 1. Robot Sepak Bola Beroda Bermain di Lapangan

Sumber: www.robocup2013.org

II. TINJAUAN PUSTAKA

A. Algoritma Greedy

Algoritma greedy adalah salah satu algoritma yang paling populer untuk menyelesaikan masalah optimasi [2]. Algoritma greedy memecahkan masalah langkah demi langkah. Pada setiap langkah algoritma greedy mengambil pilihan terbaik yang dapat diambil pada langkah tersebut. Algoritma greedy tidak peduli dengan konsekuensi dari langkah tersebut di masa depan karena algoritma greedy mempunyai prinsip “*take what you can get now*” atau ambil apa yang bisa didapat sekarang. Pada setiap langkah tersebut algoritma greedy mengasumsikan pilihan terbaik saat itu akan memberi hasil terbaik pada akhirnya. Setiap

langkah akan menghasilkan hasil optimum lokal yang diasumsikan akan menghasilkan optimum global yang merupakan solusi dari permasalahan optimasi tersebut.

Penyelesaian permasalahan optimasi menggunakan algoritma greedy mempunyai elemen-elemen sebagai berikut:

- Himpunan kandidat yang berisi elemen-elemen pembentuk solusi
- Himpunan solusi yang berisi kandidat-kandidat terpilih sebagai solusi persoalan
- Fungsi seleksi yaitu fungsi yang memilih kandidat paling memungkinkan mencapai solusi optimal dalam setiap langkah
- Fungsi kelayakan (*feasible*) yang memeriksa apakah suatu kandidat yang telah dipilih dapat memberikan solusi yang layak, yaitu kandidat tersebut bersama himpunan solusi yang sudah ada tidak melanggar kendala yang diberikan
- Fungsi obyektif yaitu fungsi yang memaksimalkan atau meminimumkan nilai solusi.

B. Gaussian Blur

Dalam pengolahan citra, *gaussian blur* adalah penyamaran citra oleh fungsi *gaussian*. Efek ini banyak digunakan pada perangkat lunak grafis untuk mengurangi derau dan mengaburkan detail pada gambar. Efek visual dari teknik ini adalah citra dengan blur yang halus dengan distribusi yang rata.

Gaussian blur umumnya digunakan pada tahap awal dalam pengolahan citra digital untuk memperbaiki struktur gambar pada skala tertentu. Secara matematis, penerapan *gaussian blur* pada gambar identik dengan menerapkan konvolusi pada citra dan fungsi *gaussian*. Karena transformasi fourier dari fungsi *gaussian* adalah fungsi *gaussian* juga, penerapan *gaussian blur* berefek pada penghilangan komponen dengan frekuensi tinggi. Dengan kata lain, *gaussian blur* tergolong *lowpass filter*.

Persamaan fungsi *gaussian* untuk dua dimensi adalah

$$G(x) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \quad (1)$$

Dengan x adalah jarak dari titik asal pada sumbu horizontal, y adalah jarak dari titik asal pada sumbu vertikal, dan σ adalah standar deviasi distribusi *gaussian*. Bila diaplikasikan dalam dua dimensi, persamaan ini menghasilkan permukaan yang konturnya berbentuk melingkar dan terkonsentrasi pada pusat lingkaran mengikuti distribusi *gaussian*. Nilai distribusi tersebut kemudian digunakan untuk membentuk matriks konvolusi yang diterapkan pada citra asli [3].



Gambar 2. Contoh Pasangan Citra Asli (Kiri) dan Citra Setelah Melalui Gaussian Blur (Kanan)

Sumber: <http://docs.opencv.org/2.4/doc/tutorials/imgproc>

C. Canny Edge Detector

Canny edge detector adalah metode deteksi tepi yang menggunakan algoritma multi tahap untuk mendeteksi tepi objek pada citra. *Canny edge detector* mampu mengekstrak informasi struktural berbagai objek sekaligus mengurangi jumlah data yang akan diproses. Di antara berbagai metode deteksi tepi yang telah dikembangkan sejauh ini, algoritma *canny edge detector* tergolong metode yang paling baik yang memberikan hasil deteksi. Kriteria yang mudah dan proses yang sederhana menjadikan algoritma ini salah satu algoritma deteksi tepi yang paling populer [4]. Terdapat lima tahap dalam algoritma *canny edge detector*. Tahap pertama adalah penggunaan *gaussian blur* untuk menghaluskan citra. Tahap kedua adalah penentuan gradien pada citra. Tahap ketiga menerapkan *non-maximum suppression* pada citra untuk menghilangkan bagian citra yang tidak diinginkan. Tahap keempat adalah penerapan *double threshold* untuk menyeleksi bagian tertentu pada *pixel* yang tersisa. Tahap terakhir merupakan penentuan tepi menggunakan histerisis [5].



Gambar 3. Contoh Pasangan Citra Asli (Kiri) dan Citra Setelah Melalui Canny Edge Detector (Kanan)

Sumber: <http://docs.opencv.org/2.4/doc/tutorials/imgproc>

D. Hough Circle Transform

Transformasi hough adalah teknik ekstraksi *fitur* yang digunakan pada analisis pengolahan citra digital. Tujuan dari penggunaan teknik ini adalah untuk menemukan objek berbentuk tertentu melalui prosedur voting. Prosedur voting ini dilakukan dalam cakupan parameter tertentu, dengan objek kandidat yang diperoleh disebut sebagai maksimum lokal pada ruang akumulator. Transformasi hough klasik berkaitan dengan identifikasi garis pada citra, namun saat ini transformasi hough telah diperluas penggunaannya untuk mengidentifikasi berbagai bentuk seperti lingkaran atau elips [3]. Metode *hough circle transform* adalah metode yang robust yang dapat digunakan

untuk mengandung banyak *noise*. Metode ini pernah diterapkan pada pengenalan bola secara otomatis dan deteksi obyek buah kelapa [6].

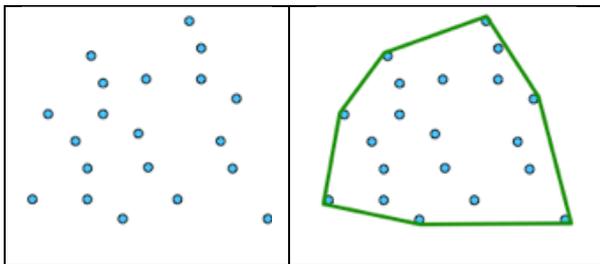


Gambar 4. Contoh Citra Cetelah Melalui Hough Circle

Sumber: <http://docs.opencv.org/2.4/doc/tutorials/imgproc>

E. Convex Hull

Pada sebuah himpunan yang berisi titik-titik misalnya $S = \{A, B, C, D, E, F\}$, *convex hull* adalah bidang *convex* terkecil yang mencakup semua titik yang ada pada set S . Jika setiap titik pada set S dihubungkan satu sama lain dengan garis lurus, maka *convex hull* S adalah garis terluar dari sejumlah garis lurus yang dihasilkan [7].



Gambar 5. Contoh Pasangan Citra Asli (Kiri) dan Citra Setelah Melalui Convex Hull (Kanan)

Sumber: <http://axon.cs.byu.edu>

III. METODE PENELITIAN

A. Kebutuhan Sistem

Program yang digunakan dalam penelitian ini diimplementasikan menggunakan *library* OpenCV 3.2 dalam bahasa pemrograman C++. Robot sepak bola yang digunakan adalah Chandrika, yaitu robot sepak bola milik Tim KRSBI Beroda ITB. Kamera yang digunakan pada robot tersebut adalah kamera *fish eye* 180°. Spesifikasi komputer yang digunakan untuk mengolah citra adalah sebagai berikut:

- *Single Board Computer*: Odroid XU4
- Sistem Operasi: Ubuntu Mate 15.04
- Prosesor: Samsung Exynos5422 Cortex™-A15 @ 2GHz
- RAM: 8 GB



Gambar 5. Robot Chandrika ITB

Sumber: Dokumen Pribadi

B. Rancangan Algoritma Greedy

Rancangan algoritma greedy yang digunakan untuk mendeteksi bola pada robot sepak bola adalah sebagai berikut:

- Himpunan Kandidat

Himpunan kandidat berisi setiap *pixel* yang dapat dipilih dari citra kamera. Himpunan kandidat direpresentasikan sebagai kelas `Mat` yang berisi pengambilan gambar tiap *frame* dari kamera. Gambar bertipe RGB yang diperoleh dari kamera diubah menjadi tipe HSV dengan prosedur `cvtColor` untuk memudahkan *thresholding*.

- Himpunan Solusi

Himpunan solusi merupakan *pixel* yang terpilih dari himpunan kandidat berdasarkan *threshold* warna bola. Untuk *thresholding* warna digunakan prosedur `inRange`. Prosedur `inRange` menerima input berupa gambar HSV, batas bawah (*lower bound*) HSV, dan batas atas (*upper bound*) HSV. Hasil dari proses *thresholding* ini berupa *binary image* yang merepresentasikan himpunan solusi.

- Fungsi Seleksi

Untuk mencapai solusi optimal, maka dipilih blob (kumpulan *pixel* yang berdekatan) yang memiliki bentuk lingkaran. Objek berbentuk lingkaran dideteksi dengan prosedur `HoughCircle`. Prosedur `HoughCircle` menerima input berupa *edge* hasil dari `Canny`.

- Fungsi Kelayakan

Kendala yang tidak boleh dilanggar adalah blob yang terpilih harus ada dalam lapangan sehingga blob yang terdeteksi di luar lapangan tidak akan dipertimbangkan lagi.

- Fungsi Objektif

Jika dalam lapangan terdeteksi lebih dari satu blob, maka solusi yang dipilih adalah blob yang memiliki jari-jari terbesar.

C. Rancangan Percobaan

Untuk mengetahui keberhasilan algoritma greedy yang diimplemntasikan, dilakukan 5 kali skenario pengujian. Berikut skenario pengujian:

- 1) Ada satu bola tetapi di luar lapangan
- 2) Ada satu bola di lapangan dan tidak ada *noise* dari citra yang diperoleh
- 3) Ada satu bola di lapangan dan ada *noise* dari citra yang diperoleh
- 4) Ada satu bola di lapangan dan satu bola lain di luar lapangan
- 5) Ada dua bola di lapangan dengan jarak berbeda terhadap robot

IV. HASIL DAN PEMBAHASAN

A. Hasil Percobaan

Berikut hasil percobaan yang diperoleh berdasarkan 5 skenario pengujian:

- 1) Ada satu bola tetapi di luar lapangan

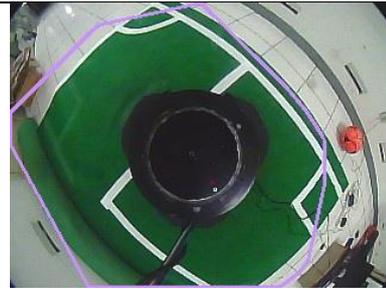


Gambar asli sebelum dilakukan pengolahan citra. Terlihat bahwa bola berada di luar lapangan dan tidak ada warna lain yang menyerupai warna bola.

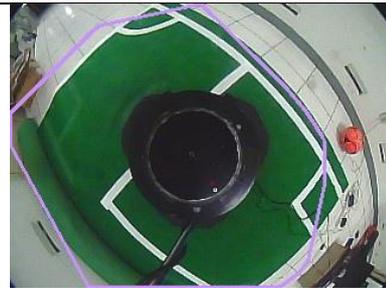


Dengan *thresholding* warna bola diperoleh *binary image*. Blob warna putih menandakan bahwa *pixel* tersebut terpilih dari citra awal.

Warna hitam menandakan bahwa *pixel* tersebut tidak terpilih dan akan diabaikan.



Garis warna ungu menandakan fungsi kelayakan, yaitu blob yang sudah terpilih dari fungsi seleksi akan divalidasi apakah ada dalam lapangan.

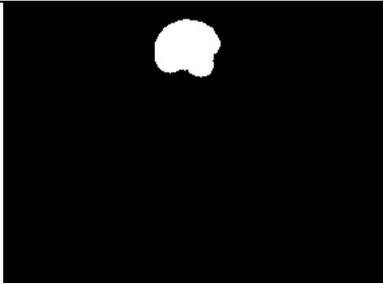


Bola dianggap tidak terdeteksi karena berada di luar lapangan.

- 2) Ada satu bola di lapangan dan tidak ada *noise* dari citra yang diperoleh



Gambar asli sebelum dilakukan pengolahan citra. Terlihat bahwa bola berada di dalam lapangan dan tidak ada warna lain yang menyerupai warna bola.



Dengan *thresholding* warna bola diperoleh *binary image*. Blob warna putih menandakan bahwa *pixel* tersebut terpilih dari citra awal. Warna hitam menandakan bahwa *pixel* tersebut tidak terpilih dan akan diabaikan.



Garis warna ungu menandakan fungsi kelayakan, yaitu blob yang sudah terpilih dari fungsi seleksi akan divalidasi apakah ada dalam lapangan.

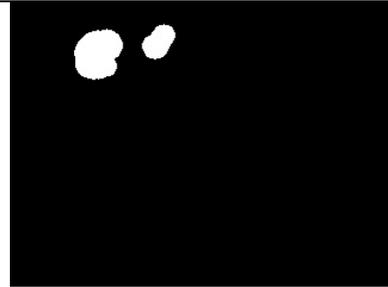


Bola terdeteksi karena berada di dalam lapangan.

- 3) Ada satu bola di lapangan dan ada noise dari citra yang diperoleh



Gambar asli sebelum dilakukan pengolahan citra. Terlihat bahwa bola berada di dalam lapangan dan ada warna lain yang menyerupai warna bola yang lebih dekat terhadap robot.



Dengan *thresholding* warna bola diperoleh *binary image*. Blob warna putih menandakan bahwa *pixel* tersebut terpilih dari citra awal. Warna hitam menandakan bahwa *pixel* tersebut tidak terpilih dan akan diabaikan. Karena bentuk *noise* kurang menyerupai lingkaran sehingga tidak lolos fungsi seleksi.



Garis warna ungu menandakan fungsi kelayakan, yaitu blob yang sudah terpilih dari fungsi seleksi akan divalidasi apakah ada dalam lapangan.



Noise tidak terdeteksi karena bentuknya tidak menyerupai lingkaran. Bola terdeteksi meskipun jaraknya lebih jauh dari *noise*.

- 4) Ada satu bola di lapangan dan satu bola lain di luar lapangan



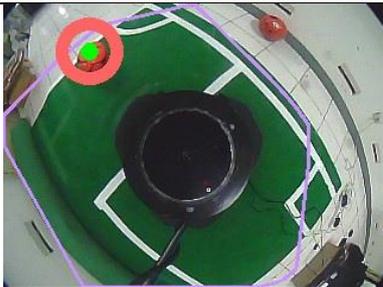
Gambar asli sebelum dilakukan pengolahan citra. Terlihat bahwa bola berada di dalam lapangan dan ada warna lain yang menyerupai warna bola yang lebih dekat terhadap robot.



Dengan *thresholding* warna bola diperoleh *binary image*. Blob warna putih menandakan bahwa *pixel* tersebut terpilih dari citra awal. Warna hitam menandakan bahwa *pixel* tersebut tidak terpilih dan akan diabaikan. Terdapat 2 blob yang dideteksi oleh *hough circle*.



Garis warna ungu menandakan fungsi kelayakan, yaitu blob yang sudah terpilih dari fungsi seleksi akan divalidasi apakah ada dalam lapangan. Bola yang di luar lapangan tidak layak sebagai solusi.

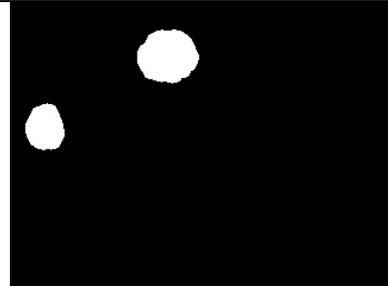


Terdeteksi bola yang ada di lapangan.

5) Ada dua bola di lapangan dengan jarak berbeda terhadap robot



Gambar asli sebelum dilakukan pengolahan citra. Terlihat bahwa terdapat 2 bola berada di dalam lapangan dengan jarak berbeda terhadap titik pusat robot.



Dengan *thresholding* warna bola diperoleh *binary image*. Blob warna putih menandakan bahwa *pixel* tersebut terpilih dari citra awal. Warna hitam menandakan bahwa *pixel* tersebut tidak terpilih dan akan diabaikan. Terdapat 2 blob yang dideteksi oleh *hough circle*.



Dua bola layak menjadi solusi karena berada dalam lapangan.



Bola dengan jari-jari terbesar dipilih sebagai fungsi optimasi.

B. Pembahasan

Berdasarkan 5 skenario pengujian di atas, dapat dilihat bahwa algoritma deteksi bola berhasil diimplementasikan pada robot sepak bola beroda. Hasil pengujian untuk setiap skenario sesuai dengan harapan yang ada, yaitu agar sistem persepsi robot berjalan dengan baik. Namun demikian algoritma tersebut masih diimplementasikan pada citra statis. Padahal pada kenyataannya, robot sepak bola bergerak setiap saat sehingga citra akan berubah tiap waktu. Algoritma ini perlu di uji coba lagi pada citra dinamis, sehingga hasilnya akan merepresentasikan kebutuhan robot pada aslinya.

V. KESIMPULAN

Algoritma deteksi bola berhasil diimplementasikan pada robot sepak bola beroda. Hasil pengujian untuk setiap skenario sesuai dengan harapan yang ada, yaitu agar sistem persepsi robot berjalan dengan baik. Dengan demikian robot mampu menjalankan visinya yaitu bermain bola di lapangan.

VI. UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih kepada Tuhan Yang Maha Esa atas terselesaikannya pembuatan makalah ini. Penulis juga berterima kasih kepada dosen mata kuliah IF2211 Strategi

Algoritma, yakni Dr. Ir. Rinaldi Munir, MT., Dr. Nur Ulfa Maulidevi, ST., MSc., dan Dr. Mesayu Lelya Khodra, ST., MT. yang telah memberikan materi dan bimbingan, baik di dalam maupun di luar kelas, yang bermanfaat dalam pembuatan tulisan ini. Tak lupa penulis juga berterima kasih kepada teman-teman Tim KRSBI Beroda ITB yang telah memberi semangat dalam penyelesaian tugas ini, di tengah perlombaan Kontes Robot Indonesia Regional II di Universitas Telkom.

DAFTAR PUSTAKA

- [1] Heru, Endra Pitowarno, dan Kusprasapta Mutijarsa. "Buku Panduan Kontes Robot Sepak Bola Indonesia Beroda". 2017
- [2] R. Munir, Diktat Kuliah IF2211, Strategi Algoritma, Program Studi Teknik Informatika, STEI, ITB, 2009
- [3] D.B. Kusumawardhana, K. Mutijarsa, W Adiprawita. "Sistem Persepsi Dua Kamera Berbasis Citra pada Robot Sepak Bola". 2017
- [4] N. Abdullah Al, K. Yinan, H. Md Nazmul, "Performance Analysis of Canny's Edge Detection Method for Modified Threshold Algorithms", in Proc. ICEEE, 2015, India
- [5] L. Juseong, T. Hoyoung, P. Jongsun, "Energy Efficient Canny Edge Detector for Advanced Mobile Vision Applications" in Proc. IEEE Transactions on Circuits and Systems for Video Technology, 2016
- [6] Ike Mardiyah Sari, Agus Zainal Arifin, dan Anny Yuniarti, "Implementasi Circular Hough Transform untuk Deteksi Kemunculan Bulan Sabit", Jurnal Teknik POMITS Vol.1 No.1, 2012.
- [7] A.M. Primabudi, A.Y. Wijaya, dan R. Soelaiman, "Desain dan Analisis Algoritma Pembangkitan Convex Hull 3 Dimensi dan Visualisasinya". Jurnal Teknik POMITS Vol.2 No.1, 2014.

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, 18 Mei 2017

Muhammad Hilmi Asyrofi - 13515083