

KALIBRASI KAMERA *OMNIVISION* PADA *MOBILE ROBOT* MENGGUNAKAN *MACHINE LEARNING*

Azzam Wildan Maulana

Departemen Teknik Komputer
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia 60111
5024201010@student.its.ac.id

Muhtadin, S.T., M.T.

Departemen Teknik Komputer
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia 60111
muhtadin@te.its.ac.id

Ahmad Zaini, S.T., M.T.

Departemen Teknik Komputer
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia 60111
zaini@te.its.ac.id

Abstrak—Dalam Kompetisi Robot sepak bola beroda, tim IRIS mendapatkan prestasi terbaiknya yaitu menjuarai RoboCup peringkat 3. Dalam permainan, Robot IRIS menggunakan kamera omnivision untuk mendeteksi hal-hal pada lingkungan sekitar. Selama ini, kalibrasi pada kamera omnivision menggunakan regresi polinomial satu arah sehingga hasilnya kurang baik pada arah yang lainnya. Tugas Akhir ini mengusulkan untuk menggunakan metode yang lebih kompleks yaitu menggunakan pendekatan Machine Learning.

Kata kunci—Deep Learning

I. PENDAHULUAN

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

II. TINJAUAN PUSTAKA

A. Hasil penelitian/perancangan terdahulu

Metode Regresi Polinomial adalah sebuah metode pendekatan terhadap data-data yang telah disediakan sebelumnya. Hasil keluaran dari metode Regresi Polinomial ini berupa rumusan matematika berdasarkan data-data yang telah disediakan. Regresi polinomial dapat bekerja secara efisien meskipun dengan model yang non-linear Grondin et al. [1].

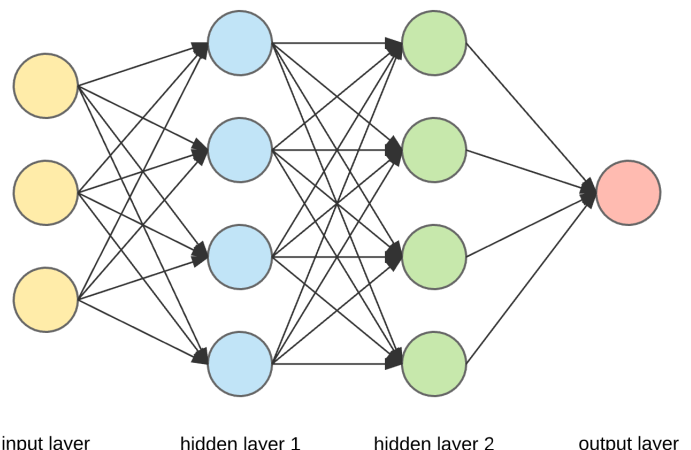
B. Teori/Konsep Dasar

1) Kamera Omnivision: Kamera *omnivision* adalah sebuah kamera yang bisa melihat 360 derajat sekitar kamera tersebut



Gambar 1: Kamera Omnivision.

Chen and Lee [2]. Jarak pandang kamera *omnivision* tidak terbatas tergantung dari resolusi kamera itu sendiri dan konstruksi cerminnya. Pada dasarnya kamera *omnivision* adalah kamera biasa yang ditembakkan ke sebuah cermin cembung sehingga pandangan kamera tersebut bisa ke segala arah.



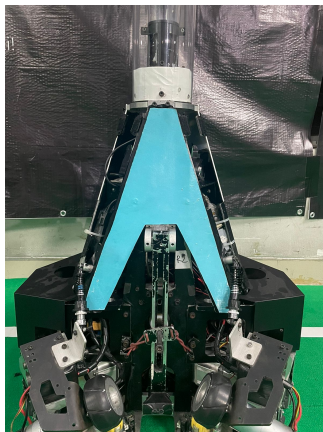
Gambar 2: Neural Network.

2) *Neural Network*: *Neural network* merupakan bagian dari Pembelajaran Mesin. *Neural network* diciptakan untuk mengatasi masalah ketidaklinearan pada sebuah model Tang et al. [3]. Pada dasarnya, *Neural network* hanyalah sekumpulan *Neuron* yang terhubung oleh sebuah *Weight* dan *Bias*. Selain *Weight* dan *Bias*, ada juga namanya *forward propagation* menggunakan *activation function* atau biasa disebut transfer function. Selain *forward propagation*, terdapat *backward propagation* menggunakan *loss function*.

3) *Activation Function*: *Activation function* adalah sebuah fungsi yang digunakan untuk men-transfer data pada masing-masing layer pada *Neural Network*. Dalam *Neural Network*, *Activation function* berperan sebagai *forward propagation* yaitu perjalanan dari layer input menuju layer output. Beberapa *activation Function* memiliki nilai saturasi biasanya bernilai 1 contohnya Sigmoid yang bernilai pada interval 0 sampai 1. Ada juga *activation Function* lain yaitu tanh yang bernilai pada interval -1 sampai 1 Kaloev and Krastev [4].

4) *Loss Function*: *loss function* adalah bagian dari *Neural Network* yang bertujuan untuk memberikan umpan balik pada model tentang baik atau buruknya fase *training*. *loss function* pada *Neural Network* bekerja pada jalur *Backward Propagation* yaitu dari layer output menuju layer input. Ada beberapa macam *loss function* salah satunya adalah MSE (*Mean Squared Error*). MSE *loss function* lebih baik digunakan pada data dengan nilai fluktuasi yang rendah Dohi et al. [5].

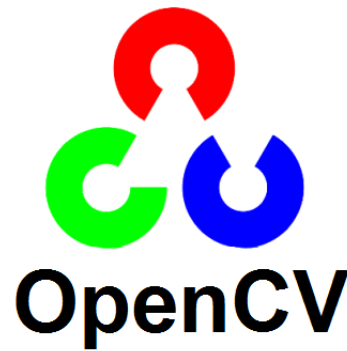
Selain *loss function*, ada sebuah teori lagi yaitu Optimizer. Optimizer adalah sebuah algoritma yang bisa digunakan untuk menentukan *learning rate* sistem training. *Learning rate* pada *Neural Network* digunakan untuk mengatur seberapa cepat model akan konvergen terhadap data-datanya.



Gambar 3: Robot IRIS tampak depan.

5) *Mobile Robot*: *Mobile Robot* adalah sebuah robot yang didesain agar bisa bergerak atau berpindah tempat dengan mudah. Performa *Mobile Robot* banyak dikhususkan pada sistem tracking, lokalisasi, dan algoritma. Ketiga hal tersebut berdasar pada kemampuan sensing yang baik Lee et al. [6]. Pada umumnya, sensor yang digunakan adalah kamera baik itu kamera biasa maupun kamera *omnivision*. Penggunaan kamera *omnivision* dapat membuat robot melihat ke segala arah,

Namun pre-processing datanya yang lebih sulit dibandingkan dengan kamera biasa.



Gambar 4: Logo OpenCV.

6) *OpenCV*: *OpenCV* adalah library open-source yang dikembangkan oleh Intel dengan bahasa pemrograman C/C++. *OpenCV* menyediakan banyak algoritma yang berhubungan dengan Visi Komputer Yildirim et al. [7]. *OpenCV* banyak digunakan untuk deteksi objek baik itu berdasarkan warna, bentuk, ukuran, dan lain lain sesuai kebutuhan program.



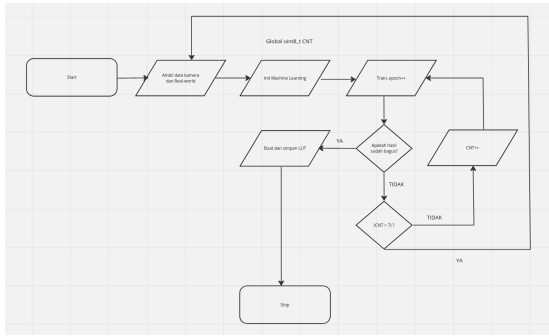
Gambar 5: Logo ROS.

7) *Robot Operating System*: ROS atau *Robot Operating System* adalah sebuah platform yang berdiri diatas Linux dan berguna untuk sinkronisasi bagian bagian dari robot Quang et al. [8]. ROS banyak digunakan sebagai inti pemrosesan data dari sebuah robot mulai dari pemrosesan data sensor hingga menjadi data aktuator. ROS menyediakan konsep modular programming dengan metode publish/subscribe untuk IPC (*Inter Process Communication*) nya. Selain memudahkan untuk transfer data antar proses, ROS juga menyediakan timer dengan scheduler default nya mengikuti Default Linux Scheduler yaitu Priority-based scheduler. Hal itu memungkinkan pengguna untuk mengatur prioritas masing-masing bagian dari robotnya.

8) *Websocket*: *Websocket* adalah jenis protokol komunikasi berbasis protokol TCP (*Transmission Control Protocol*). Protokol websocket membuat kedua pengirim dan penerima untuk selalu membuka socket nya agar bisa saling komunikasi. Dibandingkan dengan HTTP (*Hypertext Transfer Protocol*), protokol Websocket memiliki latensi yang lebih baik Guan et al. [9]. Aplikasi websocket banyak digunakan untuk aplikasi obrolan (chat), game online, aplikasi yang membutuhkan data *realtime*, dan masih banyak lainnya.

III. DESAIN DAN IMPLEMENTASI

A. Desain Sistem



Gambar 6: Desain Sistem Kalibrasi.

Pada desain sistem yang digunakan, terdapat beberapa tahapan proses yang dilakukan. Tahapan tersebut adalah sebagai berikut:

- 1) **Pengambilan Data:** Pada tahapan ini, data yang diambil adalah data koordinat titik pada kamera dan data koordinat titik pada dunia nyata.
- 2) **Machine Learning:** Pada tahapan ini, data koordinat yang telah diambil akan dilakukan proses *training* menggunakan metode *Machine Learning*. Adapun metode yang digunakan adalah *Neural Network*.
- 3) **Pembuatan Lookup Table:** Pada tahapan ini, hasil dari *training* akan digunakan untuk membuat *lookup table* yang akan digunakan untuk kalibrasi kamera. *Lookup table* dibuat dengan menggunakan dua *byte memory* untuk setiap koordinat titik pada kamera.

B. Implementasi pada Robot

Implementasi dari sistem kalibrasi kamera omnivision menggunakan *Machine Learning* dilakukan melalui beberapa tahapan. Tahapan tersebut adalah sebagai berikut:

- 1) **Load Lookup Table:** Pada tahapan ini, *lookup table* yang telah dibuat akan dimuat ke dalam program dalam bentuk *array*.
- 2) **Kalibrasi Kamera:** Pada tahapan ini, kamera akan diarahkan ke titik yang telah ditentukan. Kemudian, kamera akan mengambil gambar dan mengirimkan data koordinat titik ke program. Program akan mengubah data koordinat tersebut menjadi data koordinat yang benar berdasarkan *lookup table*.

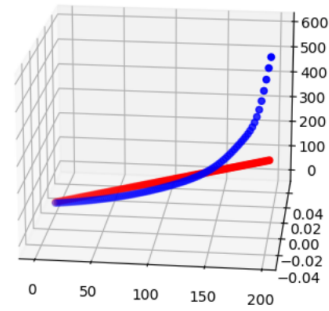
IV. HASIL DAN PEMBAHASAN

A. Visualisasi Data Kalibrasi Lama

Hasil kalibrasi kamera lama akan divisualisasikan dalam bentuk grafik 2D. Karena pada dasarnya, kalibrasi menggunakan metode yang lama hanya mengkalibrasi satu arah saja.

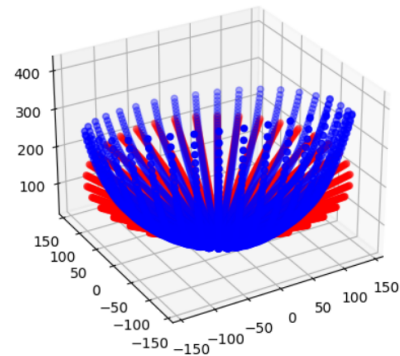
B. Visualisasi Data Kalibrasi Baru

Hasil dari kalibrasi kamera akan divisualisasikan dalam bentuk grafik 3D. Dalam grafik tersebut, terdapat dua elemen yaitu koordinat titik pada kamera dan koordinat titik pada dunia nyata.



Gambar 7: Hasil Kalibrasi Kamera Lama.

- 1) **Warna Merah:** Merupakan koordinat titik pada kamera omnivision.
- 2) **Warna Biru:** Merupakan koordinat titik pada dunia nyata.



Gambar 8: Hasil Kalibrasi Kamera Baru.

- 1) **Warna Merah:** Merupakan koordinat titik pada kamera omnivision.
- 2) **Warna Biru:** Merupakan koordinat titik pada dunia nyata.

C. Pembahasan Perbedaan Hasil Kalibrasi

Dari hasil visualisasi yang disajikan dari dua metode Kalibrasi yang berbeda. Dapat dilihat bahwa metode kalibrasi kamera omnivision menggunakan *Machine Learning* lebih baik daripada metode kalibrasi kamera omnivision menggunakan regresi polinomial. Dilihat dari grafik yang dihasilkan, metode kalibrasi kamera omnivision menggunakan *Machine Learning* dapat mengkalibrasi kamera omnivision pada semua arah. Sedangkan metode kalibrasi kamera omnivision menggunakan regresi polinomial hanya dapat mengkalibrasi kamera omnivision pada satu arah saja.

D. Pembahasan Perbedaan Waktu Eksekusi

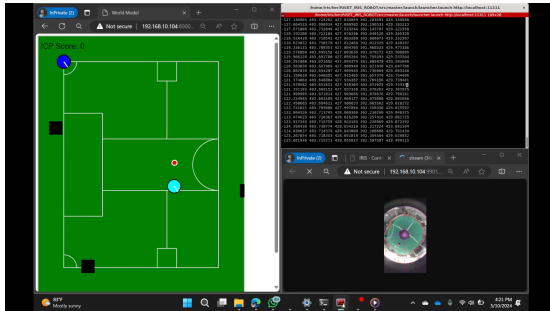
Selain dari hasil visualisasi yang dihasilkan, perbedaan lainnya adalah waktu eksekusi dari kedua metode kalibrasi.

```

=====
Very New method Elapsed time: 48 ns
Old method Elapsed time: 133 ns
=====
Very New method Elapsed time: 47 ns
Old method Elapsed time: 123 ns
=====
Very New method Elapsed time: 40 ns
Old method Elapsed time: 121 ns
=====
Very New method Elapsed time: 47 ns
Old method Elapsed time: 111 ns
=====
Very New method Elapsed time: 36 ns
Old method Elapsed time: 107 ns
=====

```

Gambar 9: Perbedaan Waktu Eksekusi.



Gambar 10: Tampilan Robot.

Metode kalibrasi kamera omnivision menggunakan regresi polinomial membutuhkan waktu yang lebih lama daripada metode kalibrasi kamera omnivision menggunakan *Machine Learning*. Hal ini disebabkan karena metode kalibrasi kamera omnivision menggunakan regresi polinomial membutuhkan proses iterasi yang lebih banyak daripada metode kalibrasi kamera omnivision menggunakan *Machine Learning*.

Berikut adalah perbandingan waktu eksekusi dari kedua metode kalibrasi:

Waktu eksekusi menggunakan metode regresi polinomial membutuhkan waktu lebih lama dikarenakan pada metode tersebut banyak melakukan proses komputasi perkalian sesuai banyaknya orde polinomial yang digunakan. Sedangkan pada metode *Machine Learning* hanya melakukan akses ke *lookup table* yang sudah dibuat sebelumnya.

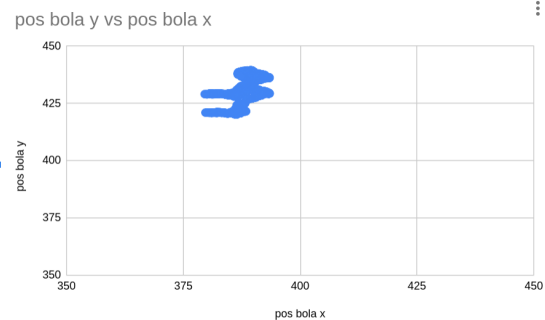
E. Pengujian pada Robot

Pengujian pada robot dilakukan dengan cara mendeteksi bola pada suatu titik lalu memutar robot pada posisinya. Hal itu menyebabkan bola akan terdeteksi melalui berbagi arah yang berbeda. Adapun hasil dari pengujian tersebut adalah sebagai berikut:

Dapat dilihat bahwa posisi bola relatif sama pada setiap arah yang diberikan.

V. KESIMPULAN

Dari hasil penelitian yang telah dilakukan, dapat disimpulkan bahwa metode kalibrasi kamera omnivision menggunakan *Machine Learning* lebih baik daripada metode kalibrasi kamera omnivision menggunakan regresi polinomial. Hal ini dapat dilihat dari kemampuan metode kalibrasi kamera omnivision menggunakan *Machine Learning* yang dapat mengkalibrasi



Gambar 11: Hasil Data Bola.

kamera omnivision pada semua arah. Sedangkan metode kalibrasi kamera omnivision menggunakan regresi polinomial hanya dapat mengkalibrasi kamera omnivision pada satu arah saja. Selain itu, metode kalibrasi kamera omnivision menggunakan *Machine Learning* juga membutuhkan waktu eksekusi yang lebih singkat daripada metode kalibrasi kamera omnivision menggunakan regresi polinomial.

PUSTAKA

- [1] F. Grondin, H. Tang, and J. Glass, "Audio-visual calibration with polynomial regression for 2-d projection using svd-phat," in *ICASSP 2020 - 2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 4856–4860.
- [2] C.-H. L. Chen and M.-F. R. Lee, "Global path planning in mobile robot using omnidirectional camera," in *2011 International Conference on Consumer Electronics, Communications and Networks (CECNet)*, 2011, pp. 4986–4989.
- [3] H. Tang, H. Li, and Z. Yi, "A discrete-time neural network for optimization problems with hybrid constraints," *IEEE Transactions on Neural Networks*, vol. 21, no. 7, pp. 1184–1189, 2010.
- [4] M. Kaloev and G. Krastev, "Comparative analysis of activation functions used in the hidden layers of deep neural networks," in *2021 3rd International Congress on Human-Computer Interaction, Optimization and Robotic Applications (HORA)*, 2021, pp. 1–5.
- [5] N. Dohi, N. Rathnayake, and Y. Hoshino, "A comparative study for covid-19 cases forecasting with loss function as aic and mse in rnn family and arima," in *2022 Joint 12th International Conference on Soft Computing and Intelligent Systems and 23rd International Symposium on Advanced Intelligent Systems (SCIS&ISIS)*, 2022, pp. 1–5.
- [6] G. D. S. Lee, K. S. Lee, H. G. Park, and M. H. Lee, "Optimal path planning with holonomic mobile robot using localization vision sensors," in *ICCAS 2010*, 2010, pp. 1883–1886.
- [7] M. Yildirim, O. Karaduman, and H. Kurum, "Real-time image and video processing applications using raspberry pi," in *2022 IEEE 1st Industrial Electronics Society Annual On-Line Conference (ONCON)*, 2022, pp. 1–6.
- [8] H. D. Quang, T. N. Manh, C. N. Manh, D. P. Tien, M. T. Van, D. H. T. Kim, V. N. T. Thanh, and D. H. Duan, "Mapping and navigation with four-wheeled omnidirectional mobile robot based on robot operating system," in *2019 International Conference on Mechatronics, Robotics and Systems Engineering (MoRSE)*, 2019, pp. 54–59.
- [9] S. Guan, W. Hu, and H. Zhou, "Real-time data transmission method based on websocket protocol for networked control system laboratory," in *2019 Chinese Control Conference (CCC)*, 2019, pp. 5339–5344.