

## **Sistem Navigasi Mobil Listrik dengan Faster R-CNN untuk Transportasi Pasien Anak**

### *Navigation System of Electric Car with Faster R-CNN for Pediatric Patient Transportation*

**Rosida Amalia Nurul Qoyima, Bambang Sri Kaloko, Khairul Anam<sup>\*)</sup>, Muchamad Arif Hana Sasono, Dicky Yusril Efendi**

*Jurusan Teknik Elektro, Fakultas Teknik, Universitas Jember*

#### **Article info**

##### **Keywords**

Navigation,  
Electric car,  
Faster R-CNN,  
semi-autonomous navigation

##### **Article history**

Received 16-07-2024

Accepted 03-10-2024

<sup>\*)Corresponding author</sup>

khairul@unej.ac.id

#### **ABSTRACT**

The use of electric cars as a means of transportation for pediatric patients has the main purpose of having a positive effect on the psychology of pediatric patients before surgery. Therefore, it is expected to accelerate the healing process. An electric car navigation system that can recognize the environment is needed. This article aims to develop a camera-based semi-autonomous navigation system using the faster R-CNN method to detect markers as electric car direction. This method optimizes the range of interest (RoI) layer to produce optimal features. Faster R-CNN is faster in generating accurate region proposals compared to R-CNN and Fast R-CNN. Various Faster R-CNN models were tested in image data processing for marker detection as the electric car steering system. Test results on FPS variations show that the best results were obtained when using the Faster R-CNN MobileNet V3 Large 320 FPN model with a value of 11.3f ps for the forward marker, 18.9 fps for the stop marker, 22.6 fps for the left turn marker and 11.1 fps for the right turn marker. With this model, the results obtained are quite good in testing the performance of the car navigation system. The results obtained in the success of the test are 70% for the forward marker test, 100% for the stop marker test, 90% for the left turn marker and 100% for the right turn marker.

#### **Article info**

##### **Kata kunci**

Navigasi,  
Mobil listrik,  
Faster R-CNN,  
Navigasi semi autonomus,

##### **Riwayat artikel**

Received 16-07-2024

Accepted 03-10-2024

<sup>\*)Corresponding author</sup>

khairul@unej.ac.id

#### **ABSTRAK**

Penggunaan mobil listrik sebagai alat transportasi pasien anak memiliki tujuan utama untuk memberikan efek positif terhadap psikologi pasien anak sebelum operasi, sehingga diharapkan dapat mempercepat proses penyembuhan. Sistem navigasi mobil listrik yang dapat mengenali lingkungan sangat diperlukan. Artikel ini bertujuan untuk mengembangkan sistem navigasi semi autonomus berbasis kamera menggunakan metode faster R-CNN untuk mendeteksi marker sebagai penanda arah mobil listrik. Metode ini mengoptimalkan lapisan *range of interest* (RoI) untuk menghasilkan fitur yang optimal. Faster R-CNN lebih cepat dalam menghasilkan proposal wilayah yang akurat dibandingkan dengan R-CNN dan Fast R-CNN. Beberapa model Faster R-CNN yang berbeda digunakan dalam pemrosesan data gambar untuk deteksi penanda sebagai sistem kemudi mobil listrik

anak. Hasil pengujian pada variasi FPS menunjukkan bahwa hasil terbaik didapatkan ketika menggunakan model Faster R-CNN MobileNet V3 Large 320 FPN dengan nilai 11.3f ps untuk marker maju, 18.9 fps untuk marker berhenti, 22.6 fps untuk marker belok kiri dan 11.1 fps untuk marker belok kanan. Dengan model ini, hasil yang didapatkan cukup baik dalam pengujian performa sistem navigasi mobil. Hasil yang didapatkan dalam keberhasilan pengujian adalah 70% untuk pengujian marker maju, 100% untuk pengujian marker berhenti, 90% untuk marker belok kiri dan 100% untuk marker belok kanan.

---

## PENDAHULUAN

Perasaan takut terhadap rumah sakit sering dialami oleh sebagian orang, terutama anak-anak, ketika harus menjalani prosedur dan perawatan. Kecemasan dan ketakutan yang dirasakan oleh anak-anak sebelum operasi dapat mempengaruhi kelancaran dan hasil dari operasi tersebut. Untuk membuat pasien anak merasa nyaman dan tidak takut sebelum operasi, diperlukan terapi bermain yang dapat memberikan dampak positif terhadap kondisi psikologis anak, sehingga membantu mempercepat proses penyembuhan. Terapi bermain dapat diterapkan selama proses pemindahan dari ruang rawat inap ke ruang operasi dengan menggunakan mobil listrik yang didesain menarik seperti mobil mainan anak-anak.

Mobil listrik dikembangkan sebagai alat bantu manusia dalam kehidupannya (Ghobadpour *et al.*, 2022) dengan berbagai inovasinya. Salah satu contohnya adalah kendaraan listrik yang dilengkapi dengan sistem kemudi otomatis menggunakan *deep learning* (Jebamikyous & Kashef, 2022). Kendaraan dengan sistem kemudi otomatis akan membantu orang yang tidak bisa mengemudi, baik karena alasan medis atau terkait usia (Adem & Kilicarlan, 2019). Penelitian pada pengemudian otomatis telah banyak berkembang dalam beberapa tahun kebelakang ini. Banyak para peneliti menggunakan model CNN sebagai pemrosesan data dalam pendeteksian citra. Penggunaan CNN dapat menghasilkan prediksi sudut kemudi yang akurat dari hasil pembacaan gambar untuk menjaga kendaraan tetap pada jalur (Mygapula *et al.*, 2021). Penggunaan CNN juga memiliki hasil yang cukup baik dalam melakukan klasifikasi gambar untuk mengatasi lokalisasi dan navigasi kendaraan didalam lingkungan ruangan (Foroughi *et al.*, 2021). Adapun pembuatan arsitektur model dengan menggabungkan CNN dan RNN (Dhruv & Naskar, 2020) untuk mendapatkan performa yang lebih baik (Setta *et al.*, 2020).

Faster R-CNN adalah salah satu sistem pendeteksi objek (Rani *et al.*, 2022). Faster R-CNN terdiri dari 3 tahapan yaitu *Convolutional Neural Network* (CNN) (EIngar *et al.*, 2021), *Region Proposal Network* (RPN) (Liu *et al.*, 2020), dan *Region of Interest* (RoI) *Pooling* (Wang *et al.*, 2022). Faster R-CNN dapat menggunakan banyak arsitektur CNN seperti ResNet50 (Parvathi & Tamil Selvi, 2021), MobileNet (Saenong *et al.*, 2023), VGG16 (Malini *et al.*, 2021) dan masih banyak lagi. Penggunaan RPN pada Faster R-CNN meningkatkan performa secara keseluruhan khususnya dari segi kecepatan pendeteksian (Li, 2021). Metode yang diusulkan ini menggunakan teknik pengelompokan baru berdasarkan jaringan Faster R-CNN, sebuah metode yang lebih efektif untuk mendeteksi kelompok pengukuran dengan kemiripan yang berkelanjutan. Metode ini mengoptimalkan lapisan RoI untuk menghasilkan fitur yang dioptimalkan (Sevugan *et al.*, 2022). Faster R-CNN lebih cepat dalam melakukan pembuatan proposal wilayah yang akurat dibandingkan dengan R-CNN (Maity *et al.*, 2021) dan Fast R-CNN (Othmani, 2022).

Artikel ini berfokus pada kendaraan listrik dengan kemudi otomatis menggunakan model Faster R-CNN (Albahli *et al.*, 2021). Kendaraan ini nantinya akan digunakan sebagai alat pemindahan pasien anak dari ruang rawat inap ke ruang operasi bedah anak. Kendaraan yang akan digunakan adalah mobil mainan anak yang telah dimodifikasi sedemikian rupa agar dapat memindahkan pasien secara otomatis dengan pendeteksian

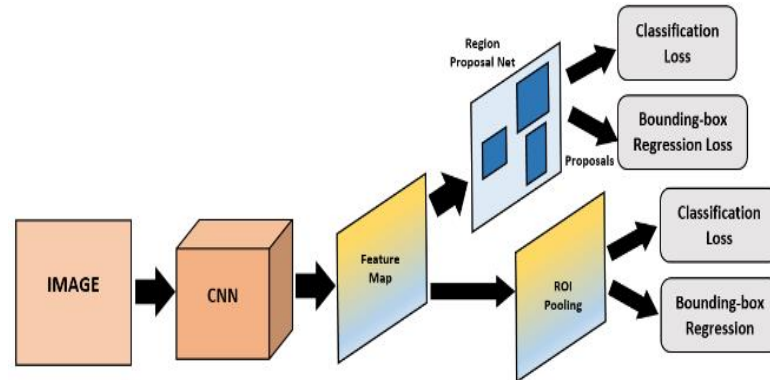
objek. Hal ini bertujuan agar dapat memberikan efek positif pada psikologis pasien anak sehingga diharapkan dapat mempercepat proses penyembuhan (Lwin *et al.*, 2020).

## METODE

### Desain dan Analisis Model CNN

#### Faster R-CNN

Faster *Region Convolutional Neural Network* (R-CNN) adalah salah satu sistem pendeteksi objek. Faster R-CNN terdiri dari 3 tahapan yaitu *Convolutional Neural Network* (CNN), *Region Proposal Network* (RPN) dan *Region of Interest (RoI) Pooling*.



Gambar 1. Arsitektur Faster R-CNN

Arsitektur Faster R-CNN yaitu pada bagian CNN dapat diisi berbagai arsitektur CNN seperti ResNet dan VGG (Gambar 1). Hasil dari CNN adalah *feature map* yang akan digunakan pada tahapan RPN. RPN berperan sebagai *region proposer* untuk menentukan *bounding box*. Pada proses RPN dilakukan pengambilan data atau citra dalam ukuran apapun sebagai *input* dan menghasilkan *frame proposals* yang masing-masingnya memiliki nilai objek. Untuk menentukan *region proposal*, dilakukan proses *sliding window* menggunakan *anchor*. Dengan *sliding window* tersebut, pengklasifikasi menentukan probabilitas proposal yang memiliki objek yang sesungguhnya dan meregresi koordinat proposal. Hasil dari tahap ini yaitu hilangnya *bounding box* yang tidak terpakai di suatu objek karena telah ditetapkan *bounding box* asli objek tersebut didalam suatu gambar.

#### ResNet

ResNet adalah struktur *Convolutional Neural Network* kompleks yang digunakan dalam *deep learning*, khususnya di bidang *computer vision*. Model ini mengatasi kesulitan yang dihadapi selama pelatihan *deep neural network*, seperti degradasi dan kehilangan gradien, dengan memperoleh pemetaan sisa. Komponen utama ResNet adalah blok sisa (*residual block*) yang terdiri dari dua atau tiga level. Blok ini menggabungkan luarannya dengan luaran lapisan bertumpuk dan menyertakan tautan pintasan untuk pemetaan identitas. Arsitektur ResNet ditunjukkan berdasarkan jumlah lapisannya, seperti ResNet-18, ResNet-50, ResNet-101, dan ResNet-152.

Penelitian ini menggunakan ResNet50 dalam penelitian ini. ResNet50, dengan struktur 50 lapisan, memperkenalkan teknik pembelajaran yang disebut pembelajaran residual. Setiap komponen ResNet50 menggabungkan koneksi *skip*, yang juga disebut sebagai koneksi *shortcut*. Karena adanya koneksi *skip* ini, jaringan saraf memiliki kemampuan untuk mempelajari pemetaan identitas yang dapat membantu proses pelatihan.

#### Faster R-CNN Nano

Faster R-CNN Nano merupakan variasi dari Faster R-CNN yang menggunakan *backbone* sangat kecil untuk mengurangi kompleksitas dan mempercepat proses inferensi. Model ini dirancang untuk digunakan pada perangkat dengan keterbatasan sumber daya seperti *edge devices* atau *embedded systems*. Arsitektur Faster R-CNN Nano terdiri dari beberapa komponen utama yang saling berinteraksi untuk melakukan deteksi objek dengan efisien: R-CNN Nano *Backbone* terdiri dari beberapa lapisan konvolusional dengan berbagai parameter untuk meningkatkan efisiensi (Tabel 1). Arsitektur dirancang untuk mengekstrak fitur dari input.

Tabel 1. Arsitektur CNN yang digunakan

Lapisan CNN	Ukuran Filter dan Jumlahnya
Pertama	Ukuran kernel 7x7 dengan 64 filter
Kedua	Ukuran kernel 3x3 dan 1x1 dengan 128 dan 256 filter
Ketiga	Ukuran kernel 1x1 dengan 128 dan 256 filter

*Region Proposal Network* (RPN) menggunakan *anchor generator* untuk membuat *bounding box* dengan berbagai ukuran dan indeks serta modul *MultiScaleRoIAlign* untuk *cropping* peta fitur dari *backbone* untuk menemukan *Region of Interest* (RoI). *RoI Head* menggunakan *Fully Connected* (FC) dan dua lapisan MLP untuk menciptakan representasi RoI yang lebih ringkas. Model *Faster R-CNN Nano* menggabungkan *Backbone*, RPN, dan *RoI head* untuk membentuk model deteksi objek *end-to-end*. Model ini menghasilkan prediksi kelas dan koordinat *bounding box* untuk setiap objek yang terdeteksi dalam gambar.

### MobileNet

MobileNet adalah arsitektur jaringan saraf yang dirancang untuk aplikasi vision pada perangkat *mobile* dan *embedded system*. Arsitektur ini menggunakan konvolusi yang dapat dipisahkan secara mendalam untuk membangun jaringan saraf yang ringkas. Arsitekturnya memiliki tiga versi: MobileNetV2, MobileNetV3, dan MobileNetV3 Large. MobileNetV3 Large menekankan peningkatan akurasi dengan memperkenalkan aktivasi *hard-swish*, desain blok hierarki, dan blok *squeeze-and-excitation*. MobileNetV3 juga memiliki desain blok hierarki berdasarkan MobileNetV2, yang menerapkan lapisan *squeeze* dan *excitation*. Modul *squeeze* dan *excitation* membantu membuat peta fitur keluaran dengan memberikan bobot yang tidak sama pada saluran masukan yang berbeda, tidak seperti CNN biasa. Blok ResNet/*inception* sering kali diperkenalkan secara terpisah, namun dalam model ini, blok tersebut digunakan bersama dengan lapisan ResNet.

### VGG16

Faster R-CNN berbasis VGG16 dibagi menjadi empat bagian utama: model dasar algoritma VGG16, *Region Proposal Network*, *ROI Pooling Layer*, dan *Fully Connected Layer*. Tahap VGG16 memproses data dengan ukuran 600 x 800 piksel, mengubah ukuran gambar untuk pra-pemrosesan. Tahapan ini menghasilkan *feature map* untuk tahap *Region Proposed Network* selanjutnya. Proses *Region Proposed Network* menggunakan proses *sliding window* menggunakan *anchor* dengan dimensi 512, menentukan probabilitas suatu proposal memiliki objek nyata dan melakukan regresi koordinat proposal. *Anchor* yang digunakan dalam penelitian ini memiliki skala 128, 256, dan 512 yang proporsional secara regional terhadap suatu objek dalam gambar. Tahap *ROI Polling Layer* memproses *input* dari hasil *Region Proposal Network* menggunakan *max pooling layer* untuk mengubah fitur-fitur di *region* yang telah ditentukan menjadi *map* fitur yang lebih kecil atau memperbaiki peta fitur. *Fully Connected Layer* Sepenuhnya berfokus pada *softmax* dan regresi *bounding box*, menggunakan *softmax* dan aktivasi *linier* di *library tensorflow*. Tahapan ini bertujuan untuk memprediksi objek kelas dan regresi *bounding box*.

### Desain Mekanik Mobil Listrik

Mobil listrik yang digunakan adalah modifikasi mobil listrik mainan (Gambar 2.) Beberapa modifikasi dilakukan. Pada bagian bawah kursi pasien diletakkan rangkaian elektronika penggerak kendaraan, diantaranya terdapat Arduino, aki, dan receiver remot kontrol ketika kendaraan akan dijalankan secara manual. Kemudian pada bagian depan mobil atau kap mobil diletakkan rangkaian elektronika dari Jetson yang digunakan sebagai sistem deteksi pada kendaraan. Kamera sebagai pendeteksi lingkungan sekitar diletakkan pada bagian depan bawah kendaraan dan diarahkan untuk menghadap ke jalan atau lintasan. Kendaraan ini mampu membawa beban kurang lebih 38 kg (Tabel 2).



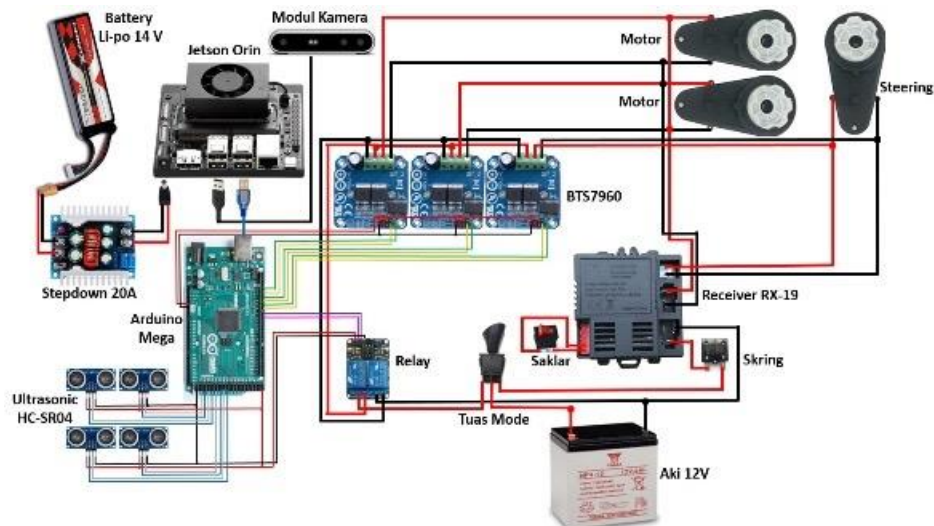
Gambar 2. Desain Mekanik Mobil

Tabel 2. Bagian-bagian pada mobil listrik mainan

No	Nama
1	Sensor Ultrasonic HC-SR04
2	Tablet HMI
3	Rangkaian Elektronika Penggerak Kendaraan
4	Saklar Otomatis dan Manual
5	Rangkaian Elektronika Sistem Deteksi
6	RealSense Dept Camera D415
7	Setir

### Perancangan Elektrikal

Pendeteksian sistem elektronika dan kelistrikan mobil listrik dilakukan oleh RealSense Dept Camera D415 yang kemudian diproses pada Jetson AGX Orin. Jetson AGX Orin menggunakan GPU berarsitektur *NVIDIA Ampere 2048-core* dengan 64 *Tensor Core* dan frekuensi maksimum 1,3GHz. Selanjutnya pengontrolan navigasi dilakukan oleh Arduino Mega 2560.



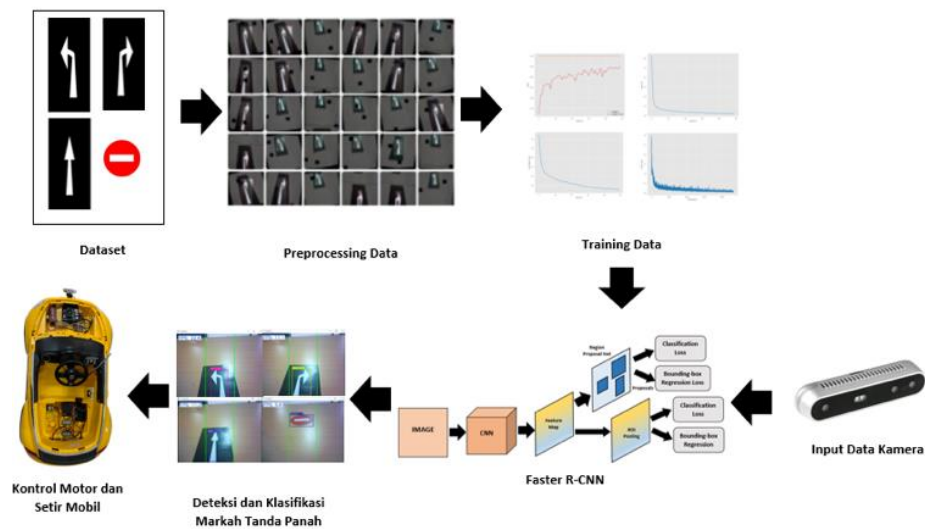
Gambar 3. Sistem Perancangan Elektronik

Pada sistem perancangan eleftrikal terdapat tuas mode yang digunakan sebagai *switch* mode otomatis atau mode *remote control* (Gambar 3). Rangkaian driver penggerak mobil listrik terdiri dari beberapa komponen. Yang pertama adalah tiga *gearbox 390 12V* atau

dinamo mobil mainan anak, dua *gearbox* digunakan sebagai motor penggerak, dan satu *gearbox* digunakan sebagai *steering*. Komponen selanjutnya adalah *driver* motor BTS 7960 yang terhubung dengan *gearbox*, *relay* dan *receiver* RX-19 yang merupakan *receiver* bawaan pabrik dari mobil mainan anak. Selanjutnya terdapat sensor ultrasonic HC-SR04 sebagai sensor jarak. Secara keseluruhan dari sistem elektronika ini akan dikontrol oleh Arduino Mega setelah mendapat kiriman serial data hasil dari pembacaan gambar oleh kamera dan melalui pemrosesan data oleh model Faster R-CNN pada Jetson Orin. Jetson Orin mendapatkan daya listrik dari baterai Li-Po 14V yang kemudian dihubungkan dengan *stepdown* 20A sebelum nantinya dihubungkan dengan Jetson Orin.

### Perancangan Sistem Navigasi

Perancangan sistem kemudi semi autonomous terdiri dari beberapa tahapan proses. Serangkaian proses tersebut meliputi data set, preprocessing data, Training data, input data kamera, faster R-CNN, deteksi dan klasifikasi markah dan kontrol motor (Gambar 4).



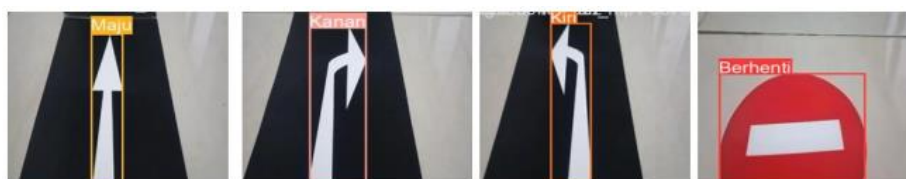
Gambar 4. Perancangan Sistem Navigasi

Sistem navigasi mobil listrik berdasarkan pada marka – marka jalan yang diletakkan di lantai. Penggunaan dataset mengacu dengan model marka panah yang biasa digunakan pada sistem lalu lintas di Indonesia. Dataset marka yang digunakan terdiri dari 4 kelas yaitu maju, belok kanan, belok kiri, dan berhenti (Gambar 5).



Gambar 5. Dataset marka jalan

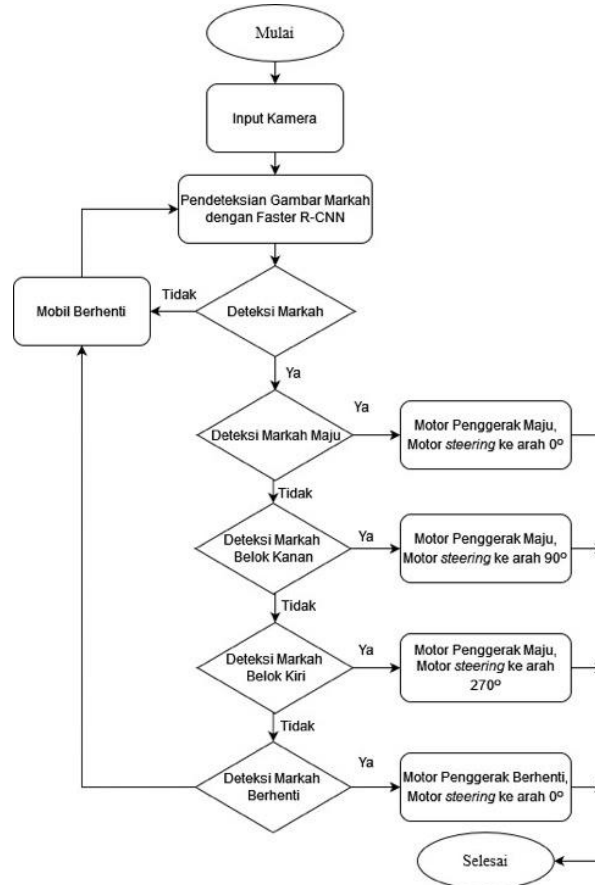
Pada *preprocessing* data, pengumpulan data set diambil dari sudut pandang kamera yang dipasang di bagian depan mobil menghadap ke arah lantai dengan sudut 80 derajat kebawah. Selanjutnya dijadikan beberapa gambar supaya dapat dilakukan anotasi dataset sesuai dengan kelasnya menggunakan website <https://roboflow.com/>, dataset yang digunakan sebanyak 480 gambar. Kemudian dilakukan anotasi data untuk membantu model mengenali dan membedakan berbagai jenis objek sesuai dengan kelasnya (Gambar 6).



Gambar 6. Anotasi Dataset Marka



Setelah dilakukan anotasi pada 480 dataset yang sudah dibagi menjadi 70% data pelatihan, 20% data validasi, dan 10% *data tes atau uji*. Selanjutnya dilakukan proses augmentasi untuk meningkatkan jumlah data dan keragaman dataset menggunakan data yang sudah ada. Setelah dilakukannya proses augmentasi data, didapatkan data sebanyak 1152 data gambar. Kemudian akan dilanjutkan dengan melatih model dengan 5 arsitektur Faster R-CNN yang berbeda.



Gambar 7. Diagram alir Sistem Navigasi

Setelah mendapatkan input dari kamera, sistem akan melakukan deteksi gambar dengan model Faster R-CNN. Gambar marka yang berhasil dideteksi akan menggerakkan motor penggerak dan menentukan arah sudut dari motor *steering* (Gambar 7).

## HASIL DAN PEMBAHASAN

### Pelatihan Model

Pada pelatihan model dilakukan menggunakan 5 arsitektur yang berbeda. Arsitektur yang digunakan adalah ResNet50 FPN, MobileNet V3 Large FPN, Nano, MobileNet V3 Large 320 FPN dan VGG16. Untuk parameter yang digunakan adalah *epoch* sebesar 50, *learning rate* sebesar 0,001 dan *batch size* sebesar 8.

Tabel 3. Perbandingan hasil pelatihan model dari model faster R-CNN yang berbeda

Model	mAP	Training Time (jam)
Faster R-CNN ResNet50 FPN	0,9242	4,7
Faster R-CNN MobileNet V3 Large FPN	0,9446	1,545
Faster R-CNN Nano	0,8216	1,45
Faster R CNN MobilenetV3 Large 320 FPN	0,9364	1,55
Faster R CNN VGG16	0,8204	6,4

Pengujian pelatihan model dengan menggunakan model faster R-CNN yang berbeda menunjukkan hasil yang bervariasi. Penggunaan model Faster R-CNN MobileNet V3 Large FPN memiliki nilai mAP terbaik yaitu 0,9446 namun membutuhkan waktu 1,545 jam atau sekitar 93 menit dalam melakukan pemrosesan data trainingnya. Waktu tersebut tidak lebih cepat dibandingkan dalam penggunaan model Faster R-CNN Nano yakni 1,45 jam atau sekitar 87 menit. Waktu tersebut merupakan waktu tercepat jika dibandingkan dengan model lainnya. Namun hasil mAP yang dihasilkan oleh Faster R-CNN Nano sebesar 0,8216. Kemudian pada hasil model Faster R-CNN VGG16 didapatkan nilai mAP terendah sebesar 0,8204 dan waktu training terlama yakni 6,4 jam atau sekitar 384 menit (Tabel 3).

#### Pengujian *Frame per Second (fps)* dengan model yang berbeda

Pengujian dilakukan secara *realtime* menggunakan kamera untuk mengetahui performa dari setiap model yang telah diujikan sebelumnya. Pengujian dilakukan dengan menggunakan 4 marka panah untuk dideteksi yaitu marka panah maju, berhenti, belok kiri dan belok kanan (Gambar 8).



Gambar 8. Pengujian FPS dengan model yang berbeda

Hasil pengujian terbaik dengan menggunakan Faster R-CNN MobileNet V3 Large 320 FPN memiliki nilai fps yang unggul pada masing-masing marka yang digunakan, yaitu 11,3 fps untuk marka maju, 18,9 fps untuk marka berhenti, 22,6 fps untuk marka belok kiri dan 11,1 fps untuk marka belok kanan. Sedangkan pada pengujian dengan model Faster R-CNN ResNet50 FPN dan model Faster R CNN VGG16 memberikan hasil yang kurang baik (Tabel 4).

Tabel 4. Perbandingan hasil pengujian FPS dengan model yang berbeda

Model	<i>Frame per Second (fps)</i>			
	Maju	Berhenti	Belok Kiri	Belok Kanan
Faster R-CNN ResNet50 FPN	0,5	0,5	0,8	1,0
Faster R-CNN MobileNet V3 Large FPN	6,1	5,6	5,9	5,7
Faster R-CNN Nano	5,9	5,9	5,5	5,6
Faster R CNN MobilenetV3 Large 320 FPN	11,3	18,9	22,6	11,1
Faster R CNN VGG16	0,6	0,3	0,6	0,6



### Pengujian Performa Model Faster R-CNN Terbaik dalam Navigasi

Pada pengujian ini digunakan Faster R-CNN dengan hasil terbaik pada pengujian sebelumnya, yakni Faster R-CNN MobileNet V3 Large 320 FPN dengan nilai fps tercepat dibandingkan model lainnya. Pengujian dilakukan 10 kali percobaan untuk pendeteksian setiap marka yang digunakan. Marka panah yang digunakan adalah marka panah maju, berhenti, belok kiri dan belok kanan. Pengujian dilakukan secara *realtime* pada mobil sampai mobil bergerak atau berpindah berdasarkan dari marka panah yang terdeteksi.



Gambar 9. Pengujian dengan marka panah maju

Dari 10 kali pengujian marka maju didapatkan presentase keberhasilan 70%, yang artinya 7 dari 10 pengujian mobil berhasil bergerak maju sesuai dengan marka yang terdeteksi. Pada 3 pengujian yang tidak berhasil, sistem tetap mendeteksi objek namun sebagai marka belok kanan, sehingga mobil bergerak belok kanan (Gambar 9).



Gambar 10. Pengujian Belok Kanan

Untuk pengujian belok kanan, dari 10 kali pengujian mobil didapatkan presentase keberhasilan 100%. Hal ini menunjukkan bahwa semua pengujian berhasil bergerak sesuai marka belok kanan (Gambar 10).



Gambar 11. Pengujian Belok Kiri

Adapun pengujian belok kiri, dari 10 kali pengujian marka belok kiri seperti didapatkan presentase keberhasilan 90%, yang artinya 9 dari 10 pengujian mobil berhasil bergerak belok kiri sesuai dengan marka yang terdeteksi (Gambar 11). Pada 1 pengujian yang tidak berhasil, sistem tetap mendeteksi objek namun sebagai marka belok kanan, sehingga mobil bergerak belok kanan. Sementara itu, untuk pengujian berhenti, dari 10 kali pengujian marka berhenti didapatkan presentase keberhasilan 100% (Gambar 12).



Gambar 12. Pengujian Berhenti

### Perbandingan hasil pengujian model Faster R-CNN dengan Yolo8

Hasil pengujian terbaik pada model Faster R-CNN didapatkan ketika menggunakan Faster R-CNN MobileNet V3 Large 320 FPN. Selanjutnya hasil terbaik yang didapatkan dibandingkan dengan model Yolo8.

Tabel 5. Perbandingan Hasil Pengujian Model Faster R-CNN dengan Yolo8

Model	Epoch	Batch Size	Learning Rate	mAP
Faster R-CNN	50	8	0.001	0,9346
Yolo8	500	64	0.0001	0,995

Hasil pengujian menggunakan Yolo8 menghasilkan nilai rata-rata presisi atau mAP yang lebih baik dibandingkan dengan penggunaan Faster R-CNN, namun demikian nilai tersebut tidak terlalu signifikan atau berbeda jauh. Hal yang berkontribusi terhadap perbedaan hasil adalah parameter yang digunakan berbeda.

### KESIMPULAN

Faster R-CNN telah mampu bekerja dengan baik untuk membantu sistem navigasi mobil listrik. Dari berbagai model Faster R-CNN yang diujikan, model Faster R-CNN VGG16 direkomendasikan untuk tidak digunakan karena memiliki waktu pelatihan yang lama yaitu sekitar 6,4 jam.

### DAFTAR PUSTAKA

- Adem, K., & Kilicarlan, S. 2019. Performance Analysis of Optimization Algorithms on Stacked Autoencoder. *2019 3rd International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT)*, 1-4. <https://doi.org/10.1109/ISMSIT.2019.8932880>
- Albahli, S., Nawaz, M., Javed, A., & Irtaza, A. 2021. An improved faster-RCNN model for handwritten character recognition. *Arabian Journal for Science and Engineering*, 46(9): 8509-8523. <https://doi.org/10.1007/s13369-021-05471-4>
- Dhruv, P., & Naskar, S. 2020. *Image Classification Using Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN): A Review* (pp. 367-381). [https://doi.org/10.1007/978-981-15-1884-3\\_34](https://doi.org/10.1007/978-981-15-1884-3_34)
- EIngar, A. A., Arafa, M., Fathy, A., Moustafa, B., Mahmoud, O., Shaban, M., & Fawzy, N. 2021. Image Classification Based On CNN: A Survey. *Journal of Cybersecurity and Information Management*, PP. 18-50. <https://doi.org/10.54216/JCIM.060102>
- Foroughi, F., Chen, Z., & Wang, J. 2021. A CNN-Based System for Mobile Robot Navigation in Indoor Environments via Visual Localization with a Small Dataset. *World Electric Vehicle Journal*, 12(3): 134. <https://doi.org/10.3390/wevj12030134>
- Ghobadpour, A., Monsalve, G., Cardenas, A., & Mousazadeh, H. 2022. Off-Road Electric Vehicles and Autonomous Robots in Agricultural Sector: Trends, Challenges, and Opportunities. *Vehicles*, 4(3): 843-864. <https://doi.org/10.3390/vehicles4030047>
- Jebamikyous, H.-H., & Kashef, R. 2022. Autonomous Vehicles Perception (AVP) Using Deep Learning: Modeling, Assessment, and Challenges. *IEEE Access*, 10: 10523-10535. <https://doi.org/10.1109/ACCESS.2022.3144407>
- Li, W. 2021. Analysis of Object Detection Performance Based on Faster R-CNN. *Journal of Physics: Conference Series*, 1827(1): 012085. <https://doi.org/10.1088/1742-6596/1827/1/012085>
- Liu, B., Luo, J., & Huang, H. 2020. Toward automatic quantification of knee osteoarthritis severity using improved Faster R-CNN. *International Journal of Computer Assisted Radiology and Surgery*, 15(3): 457-466. <https://doi.org/10.1007/s11548-019-02096-9>
- Lwin, M. N., Serhal, L., Holroyd, C., & Edwards, C. J. (2020). Rheumatoid Arthritis: The Impact of Mental Health on Disease: A Narrative Review. *Rheumatology and Therapy*, 7(3): 457-471. <https://doi.org/10.1007/s40744-020-00217-4>
- Maity, M., Banerjee, S., & Sinha Chaudhuri, S. 2021. Faster R-CNN and YOLO based

- Vehicle detection: A Survey. *2021 5th International Conference on Computing Methodologies and Communication (ICCMC)*, 1442-1447. <https://doi.org/10.1109/ICCMC51019.2021.9418274>
- Malini, A., Priyadharshini, P., & Sabeena, S. 2021. An automatic assessment of road condition from aerial imagery using modified VGG architecture in faster-RCNN framework. *Journal of Intelligent & Fuzzy Systems*, 40(6): 11411-11422. <https://doi.org/10.3233/JIFS-202596>
- Mygapula, D. V. P., S, A., V, S. V. V, & P, S. K. 2021. CNN based End to End Learning Steering Angle Prediction for Autonomous Electric Vehicle. *2021 Fourth International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, 1-6. <https://doi.org/10.1109/ICECCT52121.2021.9616875>
- Othmani, M. (2022). A vehicle detection and tracking method for traffic video based on faster R-CNN. *Multimedia Tools and Applications*, 81(20): 28347-28365. <https://doi.org/10.1007/s11042-022-12715-4>
- Parvathi, S., & Tamil Selvi, S. 2021. Detection of maturity stages of coconuts in complex background using Faster R-CNN model. *Biosystems Engineering*, 202, 119-132. <https://doi.org/10.1016/j.biosystemseng.2020.12.002>
- Rani, S., Ghai, D., & Kumar, S. 2022. Object detection and recognition using contour based edge detection and fast R-CNN. *Multimedia Tools and Applications*, 81(29): 42183-42207. <https://doi.org/10.1007/s11042-021-11446-2>
- Saenong, A., Zainuddin, Z., & Niswar, M. 2023. Identification of Poultry Reproductive Behavior Using Faster R-CNN with MobileNet V3 Architecture in Traditional Cage Environment. *2023 International Seminar on Intelligent Technology and Its Applications (ISITIA)*, 456-461. <https://doi.org/10.1109/ISITIA59021.2023.10221017>
- Setta, I. G. A., Shehata, O. M., & Awad, M. A. 2020. Multivariate Prediction of Correct Lane for Autonomous Electric Vehicle Using Deep Learning Models. *2020 8th International Conference on Control, Mechatronics and Automation (ICCMA)*, 127-130. <https://doi.org/10.1109/ICCMA51325.2020.9301610>
- Sevugan, A., Karthikeyan, P., Sarveshwaran, V., & Manoharan, R. (2022). Optimized Navigation of Mobile Robots Based on Faster R-CNN in Wireless Sensor Network. *International Journal of Sensors, Wireless Communications and Control*, 12(6): 440-448. <https://doi.org/10.2174/2210327912666220714091426>
- Wang, H., Jiang, S., & Gao, Y. 2022. Improved Object Detection Algorithm Based on Faster RCNN. *Journal of Physics: Conference Series*, 2395(1): 012069. <https://doi.org/10.1088/1742-6596/2395/1/012069>