

## ALAT BANTU INSPEKSI PENAMBAT REL KERETA BERDASARKAN KLASIFIKASI VIDEO *PROCESSING* BERBASIS YOLO CNN

**Dhimas Rizky Priyambodo<sup>1</sup>, Isa Rachman<sup>2</sup>, Ryan Yudha Aditya<sup>3</sup>**

<sup>123</sup>Program Studi Teknik Otomasi, Jurusan Teknik Kelistrikan Kapal

Politeknik Perkapalan Negeri Surabaya

Jalan Teknik Kimia, Kampus ITS Sukolilo, Surabaya 60111

*E-mail* : dhimasrizkypriyambodo@gmail.com

### Abstrak

Di Indonesia terdapat kurang lebih 7.000 km rel yang ada di Pulau Jawa dan Sumatra. Dalam pengecekan rel terdapat beberapa masalah yang ditemui salah satunya yaitu kondisi penambat rel yang longgar dalam pemasangannya bahkan hilang. Petugas PPJ (Petugas Penilik Jalur Kereta) adalah profesi yang bertanggung jawab atas pengecekan rel kereta di Indonesia, PPJ diharuskan berjalan 8 – 12 kilometer tiap harinya untuk mengecek kondisi rel dengan cara manual, metode tersebut dirasa kurang efektif dan membahayakan bagi petugas PPJ itu sendiri. Untuk mengatasi masalah tersebut penulis mencoba menyematkan teknologi pada kereta inspeksi berupa kamera pendeteksi kondisi penambat rel. Dengan menggunakan metode YOLO (*You Only Look Once*) akan mendeteksi kondisi penambat secara *realtime* sekaligus akan memberikan lokasi ketika mendeteksi penambat yang longgar maupun hilang. Hasil dari percobaan yang telah dilakukan mencapai 86,7% keberhasilan. Hasil dari alat ini bisa langsung dimonitoring lewat *interface* pada layar PC petugas dan data yang masuk ke *interface* secara otomatis akan direkap pada Microsoft excel, dengan data tersebut PT KAI dapat dijadikan acuan penjadwalan inspeksi rel dengan melihat lokasi mana yang banyak kondisi penambat hilang/longgar, lokasi tersebutlah yang akan diprioritaskan.

**Kata Kunci:** YOLO, *Image Processing*.

### 1. PENDAHULUAN

Kereta merupakan moda transportasi utama yang sering kita gunakan, kereta sendiri bisa dimanfaatkan sebagai alat untuk pengiriman barang dan mobilisasi penumpang, transportasi ini sangat unik yang dan memiliki jalur tersendiri berupa rangkaian besi baja melintang hingga beratus kilometer. Dalam memeriksa ataupun *maintenance* rel kereta api, PT KAI masih menggunakan tenaga manual yaitu dengan tenaga Petugas Penilik Jalur Kereta Api (PPJ). Setiap hari PPJ dalam melaksanakan tugasnya dengan berjalan kaki menyusuri rel sejauh 8 – 12 km dari satu titik ke satu titik yang sudah ditentukan sebelumnya. PPJ memeriksa secara detail seperti kondisi kalayakan rel. sebagainya Dengan beban tugas seperti demikian petugas suatu saat akan mengalami kelelahan, kelelahan itu sendiri penyebab menurunnya performa kerja dan dapat menimbulkan ketidaknyamanan, mengurangi produktivitas, meningkatnya kesalahan hingga kecelakaan (Sari, 2019). Dalam melakukan inspeksi pada rel kereta banyak faktor yang harus dilihat dan dicermati. Salah satunya yaitu kondisi penambat rel, penambat rel kereta adalah pengikat rel ke bantalan rel kereta

api (Hadi et al., 2018). Dalam penambat memiliki 3 kelas kondisi yaitu penambat normal, penambat rusak dan penambat hilang (Xiukun Wei et al., 2020). Pada paper ini diperkenalkan sebuah alat bantu inspeksi penambat rel kereta api berbasis citra digital menggunakan metode YOLO yang dikombinasikan dengan GPS secara *realtime*. YOLO dipilih karena telah diuji pada berbagai dataset dan terbukti memberi hasil yang cepat dan cukup akurat (Pramestya, R.H., 2018). Mengingat Indonesia sudah mempunyai kereta khusus inspeksi jalur rel alat ini dapat diaplikasikan pada kereta tersebut, dengan alat ini petugas bisa langsung memantau kondisi penambat sepasang rel melalui monitor yang ada pada kereta inspeksi tanpa harus meneliti keadaan penambat rel secara manual. Ketika alat mendeteksi penambat yang longgar maupun hilang maka secara otomatis akan memberikan peringatan dan lokasi ke monitor agar bisa segera ditindak lanjuti dan dapat dijadikan data acuan penjadwalan lokasi mana yang akan diprioritaskan karena banyak terdapat penambat yang hilang/longgar

### 2. METODOLOGI

#### 2.1 Penambat Rel Kereta

Penambat rel adalah pengikat rel ke bantalan rel kereta api (Hadi et al., 2018). Dalam pengecekan rel kereta, penambat rel juga salah satu faktor yang dicek kondisinya karena biasanya sering ditemukan penambat rel yang rusak maupun tidak standart. Dalam penambat memiliki 3 kelas kondisi yaitu penambat normal, penambat rusak dan penambat hilang (Xiukun Wei et al, 2020). Dalam kondisi penambat rusak memiliki beberapa model kasus dilapangan seperti penambat patah, rusak dan memuai. Dalam paper ini menggunakan kelas penambat rusak karena terjadinya pemuaian yang akan mengakibatkan penambat itu sendiri menjadi longgar. Sehingga pada



paper ini memiliki 3 kelas yang akan dideteksi yaitu penambat normal, hilang dan longgar

**Gambar 2.1** Penambat Rel Kereta

## 2.2 Desain Perancangan Sistem

Sistem menggunakan kamera sebagai pengambil data *input* berupa gambar yang diperoleh dari kamera tentang keadaan penambat apakah penambat tersebut normal, longgar ataupun hilang, lalu diproses laptop dengan menggunakan video *processing* menggunakan metode YOLO untuk mengetahui keadaan dari penambat pada rel kereta, mengapa dipilih YOLO sebagai metode karena YOLO telah diuji pada berbagai dataset dan terbukti memberi hasil yang cepat dan cukup akurat (Pramestya, R.H., 2018). Bila YOLO mendeteksi kondisi penambat tersebut hilang ataupun longgar maka modul GPS akan mengirimkan lokasi ke arduino dan arduino akan mengirimkan data lokasi ke laptop lalu ditampilkan pada *interface*. Begitu juga dengan hasil dari image keadaan penambat akan dikirimkan ke *interface* untuk dimonitoring melalui PC petugas.

## 2.3 Convolutional Neural Network (CNN)

Dalam metode YOLO yang digunakan pada penelitian ini CNN juga termasuk dalam tahapan dari YOLO itu sendiri. *Convolutional Neural Network* (CNN) adalah salah satu pengembangan dari *Multilayer Perceptron*

(MLP) yang memiliki tujuan untuk mengolah data dua dimensi (Putra, 2019). CNN biasa digunakan untuk mengenali atau monitoring suatu objek. CNN hampir sama dengan *Neural Network* (NN) seperti *weight*, bias dan *activation function* ada keduanya tapi yang membedakan arsitektur CNN dibagi menjadi dua bagian besar, yaitu *Feature Extraction Layer* dan *Fully Connected Layer* (Abi Rachman, dkk, 2018).

### 2.3.1 Feature Extraction Layer

Pada tahap ini dilakukan proses *encoding* dari awalnya sebuah gambar di *convert* menjadi angka – angka yang sesuai gambar itu sendiri (*Feature Extraction*). *Feature Extraction Layer* terdiri dari dua bagian yaitu *Convolutional Layer* dan *Pooling*.

#### o Convolutional Layer

*Convolutional Layer* adalah filter yang terdiri dari neuron yang memiliki panjang, tinggi, kedalaman. Contoh *Convolutional Layer* memiliki ukuran  $5 \times 5 \times 3$  berarti *Convolutional Layer* tersebut memiliki panjang 5 pixels, tinggi 5 pixels, tebal atau jumlah 3 buah.

#### o Pooling Layer Pooling

Layer ini terjadi proses *downsampling* dimana memiliki fungsi untuk menjaga ukuran data ketika konvolusi, selain menjaga ukuran pada layer ini juga berfungsi untuk meningkatkan invariasi posisi dari fitur. Pada dasarnya bagian ini menjelaskan bagaimana penelitian itu dilakukan. Materi pokok bagian ini adalah:

### 2.3.2 Fully Connected Layer

Layer ini bertujuan untuk melakukan transformasi pada dimensi data agar dapat diklasifikasikan secara linear. Sebelum masuk ke dalam layer ini setiap neuron yang berasal dari convolution layer harus di ubah menjadi satu dimensi terlebih dahulu agar saat masuk ke fully connected layer informasi dari setiap neuron tidak hilang. Begitu juga dengan activation map yang dihasilkan feature extraction tidak bisa langsung diproses di layer ini karena masih dalam bentuk multidimensional array, sehingga harus diproses reshape activation map menjadi sebuah vector

## 2.4 YOLO (You Only Look Once)

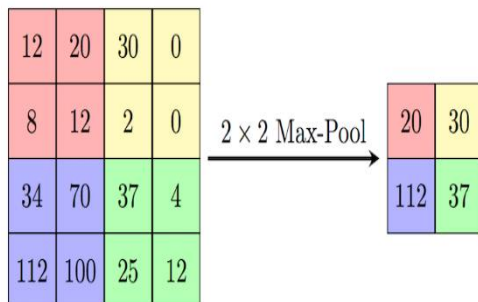
Yolo (*You Only Look Once*) merupakan algoritma yang dikembangkan agar bisa memudahkan pemantauan dan

pengidentifikasi secara *real time*. Untuk melakukan pengidentifikasi secara real time sendiri harus menggunakan metode yang cepat agar suatu objek bisa secara cepat dan tepat diidentifikasi. Yolo sendiri memiliki 4ms/gambar sehingga metode ini dirasa efektif untuk pengidentifikasi secara *real time*. Sistem pendeteksian yang dilakukan adalah dengan menggunakan *repurpose classifier* atau localizer untuk melakukan deteksi. Sebuah model diterapkan pada sebuah citra di beberapa lokasi dan skala. Daerah dengan citra yang diberi score paling tinggi akan dianggap sebagai sebuah pendeteksian (Unsky, 2017).

#### 2.4.1 Konvolusi & Max Pooling

Konvolusi & max pooling adalah termasuk proses dari CNN, konvolusi itu sendiri adalah sebuah proses dimana *input* citra dimanipulasi dengan menggunakan subwindow untuk menghasilkan citra yang baru menggunakan kernel 3x3 yang bertujuan untuk mengurangi dimensi fitur gambar dengan menghilangkan beberapa nilai pixel.

Max pooling adalah proses dimana penyusutan *input* citra dengan cara mengurangi jumlah dari parameter dengan cara mengambil nilai terbesar dari bagian tersebut atau biasa juga disebut operasi *down sampling* dengan menggunakan kernel 2x2 filters dan stride 2 seperti *input* citra semula 416x416 menjadi 208x208 dst.



Gambar 2.2 Max Pooling

Jadi sebelum masuk kedalam 2 proses tersebut terdapat *input* citra yang sudah diubah ukurannya menjadi 416x416x3 dengan *input* citra berupa RGB, dari *input* citra tersebut masuklah ke tahap konvolusi sebanyak 9 kali dan max pooling sebanyak 6 kali.

Layer	Type	Filters	Size/stride	Input	Output
0	Convolutional	16	3×3/1	416×416×3	416×416×16
1	Maxpooling		2×2/2	416×416×16	208×208×16
2	Convolutional	32	3×3/1	208×208×16	208×208×32
3	Maxpool		2×2/2	208×208×32	104×104×32
4	Convolutional	64	3×3/1	104×104×32	104×104×64
5	Maxpool		2×2/2	104×104×64	52×52×64
6	Convolutional	128	3×3/1	52×52×64	52×52×128
7	Maxpool		2×2/2	52×52×128	26×26×128
8	Convolutional	256	3×3/1	26×26×128	26×26×256
9	Maxpool		2×2/2	26×26×256	13×13×256
10	Convolutional	512	3×3/1	13×13×256	13×13×512
11	Maxpool		2×2/1	13×13×512	13×13×512
12	Convolutional	1024	3×3/1	13×13×512	13×13×1024
13	Convolutional	1024	3×3/1	13×13×1024	13×13×1024
14	Convolutional	40	1×1/1	13×13×1024	13×13×40

#### 2.4.2 Confidence & Class Prediction

*Confidence score* adalah salah satu parameter yang diperlukan, *confidence score* terdapat pada setiap *bounding box* yang terbentuk. Dalam menentukan *confidence score* dalam setiap *bounding box* Yolo mendefinisikan dengan :

$$confident\ score = Pr(Object) \times IoU_{Pred}^{truth}$$

Dimana :

Pr(object) = Probabilitas Objek

IoU = Intersection Over Union

Setelah setiap *bounding box* memiliki *score* akan dilanjutkan dengan memprediksi kelas dari setiap *bounding box* proses tersebut di sebut dengan *class prediction*, dari semua kelas yang terprediksi tidak semua akan dipergunakan untuk menentukan kelas dari *input* citra tersebut masih ada proses selanjutnya untuk menyeleksi kelas – kelas tersebut.

#### 2.4.3 Non Max Suppression

Metode ini digunakan untuk menyeleksi *bounding box* yang muncul lebih dari satu dalam satu objek yang sama dengan membandingkan nilai *confidence* masing-masing *bounding box*, hanya nilai *confidence* yang paling tinggi (maksimal) yang akan dipertahankan.

Dimana sebelum masuk pada *non – max suppression* *input* citra memiliki banyak *bounding box* yang terprediksi, setelah diproses maka akan dihasilkan *bounding box* yang memiliki nilai *confidence* yang paling tinggi. Standart untuk mempertahankan skor dari *bounding box* disebut dengan *threshold*, *threshold* sendiri dapat diatur tergantung dari

seberapa akurat deteksi yang diinginkan.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Hasil Pengujian Metode YOLO

Dari **Gambar 3.1** didapatkan mAP *Mean Average Precision* tertinggi yaitu di angka 86%, angka tersebut didapat dari hasil proses *training* dataset yang telah dilakukan. Dengan mAP yang terbaik tersebut digunakan sebagai file bobot yang akan digunakan untuk prototipe. Sementara rata – rata presisi dari setiap kelas yang sudah *training* untuk kelas normal memiliki rata – rata presisi memiliki nilai 91,63%, pada kelas hilang memiliki rata – rata presisi 99,36%, pada kelas longgar memiliki rata – rata 66,41%. Untuk nilai yang lain – lain sudah dicantumkan pada table 3.1



**Gambar 3.1** mAP

Dalam **Gambar 3.1** tersebut disajikan grafik hasil dari *training*, grafik biru menunjukkan proses dataset di *training* mulai dari 0 literasi, 1000 literasi, 1200 literasi sampai 6000 literasi. Sementara grafik merah menunjukkan rata – rata presisi dari setiap proses literasi mulai dari 0 literasi sampai 6000 literasi hingga dihasilkan rata – rata presisi 86% tertinggi pada saat 6000 literasi. Grafik tersebut dapat dipantau ketika melakukan *training* dengan durasi berjam- jam sehingga dapat dilakukan pemantauan sudah sampai mana dataset di *training*.




**Tabel 3.1** Data Hasil *Training*

Load Model		Hasil
Normal	AP	91,63%
	TP	304
	FP	46
Longgar	AP	66,41%
	TP	36
	FP	9
Hilang	AP	99,36%
	TP	11
	FP	0
FN		49
Waktu Pemrosesan (s)		95
Precision		0,86
Recall		0,88
F1-score		0,87
IoU		68,23%
mAP@0.5		85,80%

#### 3.2 Hasil Pengujian *Deteksi Real Time*

Pengujian dilakukan di tempat terbuka dengan menggunakan prototype yang sudah dibuat. Pengujian dilakukan sekitar pukul 9 pagi dengan cara setiap kelas di uji coba sebanyak 10 kali percobaan lalu dihitung berapa kali sukses dan berapa kali gagal . Jadi dari setiap kelas pada penambat memiliki masing persentase untuk kelas normal 90%, hilang 90% dan longgar 80% sehingga memiliki rata – rata keberhasilan dalam pembacaan suatu kondisi penambat rel kereta mencapai 86.7%.

**Tabel 3.2** Hasil Pengujian Deteksi Realtime

	Hasil Pengujian	Hasil Prediksi	Keterangan
1	FPS: 12.28 	Normal	Benar
2	FPS: 12.16 	Hilang	Benar
3	FPS: 12.15 	Longgar	Benar

Keterangan :

AP : Average Precision

TP : True Postif

FP : False Positif

FN : False Negatif

### 3.3 Hasil Pengujian GPS

	Kondisi Penambat	Modul GPS		Konversi Perhitungan Manual	Jarak Sebenarnya (m)	Simpangan (m)
		Latitude	Longitude			
1.	Longgar	-7,284298	112,78981	0 (titik awal)	0 (titik awal)	0
2.	Hilang	-7,284329	112,789817	3,53	5	1,47
3.	Longgar	-7,284368	112,789817	7,83	10	2,17
4.	Hilang	-7,284422	112,789825	13,90	15	1,1
5.	Hilang	-7,284455	112,789825	17,55	20	2,45
6.	Hilang	-7,28452	112,789802	24,72	25	0,25
7.	Hilang	-7,284557	112,789802	28,84	30	1,16
8.	Hilang	-7,284592	112,789787	32,82	35	2,18
9.	Longgar	-7,284653	112,789787	39,60	40	0,4
10	Hilang	-7,284683	112,789794	42,89	45	2,11
Rata - rata						1,47

**Gambar 3.2** Hasil Pengujian GPS

Perhitungan konversi menggunakan rumus Euclidean Distance, rumus tersebut digunakan untuk mengukur jarak antar 2 titik *Euclidean space* (Pamungkas canggih,2019) .

Berikut adalah rumus Euclidean Distance :

$$n = \sqrt{(Lat1 - Lat2)^2 + (Long1 - Long2)^2} \times 111,319$$

Keterangan :

Lat1 = Latitude dari titik awal

Lat2 = Latitude dari titik ke dua

Long1 = Longitude dari titik awal

Long2 = Longitude dari titik ke dua

111,319 = Ketetapan berupa jarak dari 1° bumi

## 4. PENUTUP

### 4.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, kesimpulan yang dapat di ambil yaitu sebagai berikut.

1. Metode YOLO yang digunakan dirasa sudah cukup tepat dibuktikan dengan nilai persentase rata - rata pada kelas normal 91,63% , pada kelas hilang 99,36% ,pada kelas longgar 66,41% serta rata -rata keseluruhan sebesar 85,80% .
2. Dengan alat yang sudah dirancang serta ujicoba yang telah dilakukan dengan bantuan prototype serta adanya rekaman data melalui interface ke dalam Microsoft excel akan dapat menjadi data acuan .
3. petugas untuk memprioritaskan daerah mana yang harus segera dilakukan perbaikan sehingga petugas tidak perlu melakukan pengecekan secara manual langsung pada rel kereta dan dampaknya akan meminimalisir kecelakaan kerja bagi petugas.

Metode YOLO yang digunakan dirasa sudah cukup tepat diimplementasikan pada monitoring kondisi penambat secara *realtime* dibuktikan dengan nilai persentase rata – rata keberhasilan didapatkan sebesar 86.7% serta fps yang cukup lancar diangka 10 – 12 fps.

### 4.1 Saran

Dalam pengujian yang telah dilakukan dalam penelitian ini, maka terdapat beberapa hal yang harus diperhatikan dan dikembangkan lagi untuk penelitian selanjutnya yaitu:

1. Penggunaan PC/Laptop dengan spesifikasi yang tinggi agar hasil pembacaan keadaan penambat agar lebih cepat dan akurat.
2. Menambah data *training* agar pembacaan kamera semakin baik
3. Disarankan menggunakan modul GPS yang me

## 5. DAFTAR PUSTAKA

- [1] Hadi, T., Sugiharto, , Supriyo, S., & Sudarmono. (2018). *Rancang bangun*

*komponen pendorong alat pasang  
pendrol untuk uji penambat rel. 04, 36–  
44.*

- [2] Herlangga, I., & Nugroho, H. (n.d.).  
*Perancangan Portabel Komputer kali  
Linux Berbasis Raspberry Pi 3 Model B  
Manfaat Untuk Test Penetrasi.*
- [3] Husni, M., I, R. M., & Bisaptanto, J. (n.d.).  
*Prototype Sistem Monitoring Rumah  
Menggunakan Webcam 105 - 111.*
- [4] Putra, F. (2019). *Pengawasan Jumlah  
Barang Elektronok Berdasarkan  
Klasifikasi Video Processing Dengan  
Metode YOLO - COnvolutional Neural  
Network.*
- [5] Xiukun, W dkk(2020) Multi-Target Defect  
Identification for Railway Track Line  
Based on Image Processing and  
Improved YOLOv3 Model School of  
Traffic and Transportation, Beijing  
Jiaotong University, Beijing 100044,  
China