

## **Perancangan Pendeteksi Objek Menggunakan Metode YOLO Dan OpenCV**

**Farhan Dian Irfansyah<sup>1\*</sup>, Nanda Putra Kusuma<sup>2</sup>, Raffly Pramudia Renaldi<sup>3</sup>, Perani Rosyani<sup>4</sup>**

<sup>1-4</sup>Fakultas Ilmu Komputer, Teknik Informatika, Universitas Pamulang, Jl. Raya Puspipetek No. 46,  
Kel. Buaran, Kec. Serpong, Kota Tangerang Selatan. Banten 15310, Indonesia

Email: <sup>1</sup>[farhanbabe854@gmail.com](mailto:farhanbabe854@gmail.com), <sup>2</sup>[nandakusuma889@gmail.com](mailto:nandakusuma889@gmail.com),

<sup>3</sup>[raffly.pramudia@gmail.com](mailto:raffly.pramudia@gmail.com), <sup>4</sup>[dosen00837@unpam.ac.id](mailto:dosen00837@unpam.ac.id)

(\* : coresponding author)

**Abstrak**– Deteksi objek adalah cabang ilmu dari *computer vision*, yang memiliki pengertian objek yang dapat dilihat secara visual dalam sebuah gambar atau video kemudian dapat diteliti, dideteksi, dan dikenali secara langsung oleh komputer. Pada jurnal kami ini menampilkan secara komprehensif tentang pendeteksi objek yang memakai dua kombinasi dari OpenCV dan YOLO, dua kombinasi ini termasuk ke sebuah algoritma pembelajaran mendalam yang canggih atau *Deep Learning*. Pengertian OpenCV yaitu sebuah kumpulan pustaka *computer vision* yang memiliki sumber terbuka yang dapat diartikan dengan beragamnya fungsi algoritma disana. Di lain sisi, YOLO yaitu sebuah metode yang memiliki algoritma pendeteksi objek, berfungsi untuk mencapai kinerja maksimal secara langsung tanpa mengurangi sedikit pun akurasi. Pembelajaran pada kali ini, kami mengkombinasikan kemampuan dari kedua program ini untuk membuat sebuah sistem pendeteksi objek yang memiliki kinerja yang baik. Sekarang kami akan memberikan pendekatan singkat tentang kedua program ini, yang memiliki pusat di bagian komponen utama. Kesimpulan dari jurnal kami ini memberikan pengalaman secara komprehensif tentang pendeteksi objek memakai dua kombinasi yaitu OpenCV dan YOLO. Pada studi ini akan memperhatikan tentang keuntungan dari kombinasi dari kedua kombinasi tersebut dalam hal waktu dan ketepatan serta memberikan implementasi yang mudah dari sistem tersebut. Pada bagian akhir akan menampilkan kinerja sistem untuk aplikasi pendeteksi objek secara langsung.

**Kata Kunci:** Deteksi objek, OpenCV, YOLO

**Abstract**– *Object detection is a branch of computer vision, which means objects that can be seen visually in an image or video can then be studied, detected, and recognized directly by a computer. In this journal, we present a comprehensive overview of object detectors using two combinations of OpenCV and YOLO, these two combinations are included in a sophisticated deep learning algorithm or Deep Learning. The definition of OpenCV is a collection of open source computer vision libraries that can be interpreted as having various algorithm functions there. On the other hand, YOLO is a method that has an object detection algorithm, which functions to achieve maximum performance directly without reducing its accuracy. In this study, we combine the capabilities of these two programs to create an object detection system that has good performance. Now we will provide a brief approach to these two programs, which have a center in the main component section. The conclusion of our journal provides a comprehensive experience of object detectors using two combinations, namely OpenCV and YOLO. This study will pay attention to the advantages of the combination of the two combinations in terms of time and accuracy and provide easy implementation of the system. At the end will show the system performance for the object detection application directly.*

**Keywords:** *Object detection, OpenCV, YOLO*

### **1. PENDAHULUAN**

Kemajuan teknologi pada saat ini sudah mencapai puncaknya sehingga beberapa teknologi didalamnya dapat digunakan untuk mempermudah hidup manusia. Salah satu contoh pengimplementasian teknologi yang dapat dimanfaatkan yaitu pendeteksi sebuah objek hanya dengan kamera, sistem pendeteksi objek mempunyai harapan untuk dapat membantu serta mempermudah kehidupan manusia dengan berbagai objek yang sudah dikonfirmasi sebelumnya. Pendeteksi objek adalah cabang *ilmu computer vision* memiliki fungsi untuk mencari sebuah objek anomali di sebuah gambar maupun video. Pendeteksi objek merupakan salah satu metode guna untuk mengetahui serta memberitahu lokasi sebuah objek dikelas tertentu. Adanya fungsi pendeteksi objek yaitu untuk memudahkan manusia menjalankan kehidupan sehari-hari.

Program ini bertujuan untuk membuat pendeteksi dari banyaknya objek bersifat umum dikenal dan mudah ditemukan di kehidupan sehari-hari, contoh manusia, motor, tumbuhan, maupun objek anomali lainnya. Satu sebab penting diperlukannya sebuah pendeteksi ini yaitu untuk memberi

tahukan tentang posisi objek dan identifikasi objeknya dari sebuah gambar atau video, dan alasan utama dari program ini yaitu untuk mengenal objek anomali yang terdapat disekitar pengguna. YOLO merupakan satu dari banyaknya metode yang mungkin digunakan untuk pengaplikasian pendeteksi objek, pada proses penyusunannya juga memanfaatkan program lain.

Bagi kehidupan, pengenalan suatu objek memiliki kategori yang mudah, tetapi untuk komputer, hal itu termasuk kedalam kategori sulit, dikarenakan melewati beberapa tahapan yang diawali dengan mengenali objek, lalu mengidentifikasi, dan yang terakhir menampilkan objek di presisi acak. Dalam YOLO dapat memperkirakan objek yang muncul di sekitar serta letak dari objek anomali itu berada, algoritma ini juga memiliki frame rate yang lebih tinggi untuk pendeteksian.

YOLO termasuk sebuah metode pendeteksi objek yang mempunyai kinerja terbagus dengan bermodal fitur yang sangat unik dan tingkatan pencaharian yang lebih memadai dan juga mempunyai penyusunan jaringan yang simpel. Dalam memahirkan gambar objek serta pengujian, metode ini melewati metode pendeteksi lainnya seperti RCNN dan DPM. Aplikasi pendeteksi bermacam objek ini dibuat melalui bahasa *python* dan *library* untuk memproses didalamnya. OpenCv yaitu sebuah pustaka algoritma untuk fungsi pemrograman komputer *vision* secara langsung dan fungsi lainnya yaitu dapat membedakan objek hanya bermodalkan kamera. OpenCv berlisensi BSD memiliki sifat terbuka untuk pemakai biasa maupun berbayar.

## **2. METODE PENELITIAN**

### **2.1 Metode YOLO**

Metode berikut yaitu sebuah metode yang telah menjadi pendekatan yang populer dalam deteksi objek, karena dapat menghasilkan deteksi secara real-time dengan akurasi yang tinggi. Metode ini juga dapat digunakan untuk memperoleh sistem pemberi tahu sebuah objek. Metode ini hanya dapat diakses dari *open source neural network*. Lalu kelebihanannya mudah memproses objek dalam real-time pada empat puluh lima (45) *frames per second*. Cara kerja YOLO yaitu diawali dengan melihat gambar secara keseluruhan yang hanya dilakukan sekali, lalu melewati pemrosesan *neural* sekali, dan terakhir YOLO dapat menampilkan objek yang terlihat.

### **2.2 Kotak Bounding**

Di sebuah kasus yang digarap (Putra et al., 2021), diberi penjelasan yaitu di dalam kotak imajiner guna meliputi objek terdeteksi. Kotak ini mempunyai bentuk kotak serta ukurannya yang menyamakan dengan ukuran objek yang muncul. Cara membuatnya diharuskan mempunyai koordinat piksel.

### **2.3 Skor Mobilitas**

Dalam penyelesaian kasus oleh (Hutauruk et al., 2020), YOLO membutuhkan skor pada setiap kotak. Ini bertujuan memobilisasikan kotak itu mempunyai objek, lalu dijumlahkan dengan IoU per kotak prediksi dan yang akan dihasilkan selama waktu pengujian cobaan.

## **3. ANALISA DAN PEMBAHASAN**

### **3.1 Analisa**

Performa pada YOLO mudah di uji dengan beberapa percobaan meliputi rata-rata dari presisi, daya memori, dan kegesitannya. Pada percobaan kali ini menghasilkan mungkin dapat wawasan tambahan dari model berikut.

#### **a. Kemampuan pada Akurasi**

YOLO memiliki pengakurasian cukup baik, meskipun performanya terasa lebih dibawah daripada dengan model YOLO lainnya. Skor mAP yang diperoleh untuk mengukur presisi rata-rata di beragamnya kategori objek, mendapati nilai 20-30%. Nilai ini mungkin terasa lebih dibawah dibanding dengan versi lainnya, namun keunggulan dari YOLO terletak pada kegesitan inferensi yang sangat cepat.

### b. Kemampuan Deteksi Objek

YOLO dapat menemui objek apapun dengan kelebihanannya yang dimiliki menghasilkan performa yang cukup baik. Hal itu memperoleh hasil jawaban tepat dengan mengeksplorasi objek di beragamnya kategori, seperti halnya makhluk hidup, barang, dan lainnya pada kehidupan. Kemampuannya dapat menemukan benda yang ukurannya lebih mungil dan mungkin akan merasa sedikit terganggu oleh pengecilan arsitekturnya yang akan diperkecil mengikuti bentuk objek, akan tetapi kinerjanya terasa cukup baik.

### c. Kecepatan Inferensi

Hal ini merupakan keunggulan utama yang dimiliki oleh YOLO yaitu kegesitan inferensi yang sangat dibutuhkan. YOLO sengaja dibentuk untuk pendeteksian objek secara real-time yang mempunyai sumber daya terbatas, contoh hal ini yaitu perangkat seluler. Keunggulan inilah memakan waktu yang jauh sangat sedikit dibanding dengan program lain, memungkinkan dapat memproses stream video atau hasil kamera langsung.

## 3.2 Pembahasan

Perancangan ini diawali dengan penggunaan libraries lalu dilanjutkan oleh percobaan program dengan menu YOLO yang dikombinasikan oleh Open Cv

### a) Pendeteksi Gambar

```
1 import cv2
2 import numpy as np
3
4 # Load Yolo
5 net = cv2.dnn.readNet("weight/yolov3.weights", "cfg/yolov3.cfg")
6 classes = []
7 with open("coco.names", "r") as f:
8     classes = [line.strip() for line in f.readlines()]
9 layer_names = net.getLayerNames()
10 output_layers = [layer_names[i - 1] for i in net.getUnconnectedOutLayers()]
11 colors = np.random.uniform( low=0, high=255, size=(len(classes), 3))
12
13 # Loading image
14 img = cv2.imread("pic/pic1.jpg")
15 img = cv2.imread("pic/pic2.jpg")
16 img = cv2.resize(img, dsize=None, fx=0.6, fy=0.6)
17 height, width, channels = img.shape
18
19 # Detecting objects
20 blob = cv2.dnn.blobFromImage(img, scalefactor=0.00392, size=(416, 416), mean=(0, 0, 0), swapRB=True, crop=False)
21
22 net.setInput(blob)
23 outs = net.forward(output_layers)
24
25 # Showing informations on the screen
26 class_ids = []
27 confidences = []
28 boxes = []
```

**Gambar 1.** Source Code Pendeteksi Gambar

### b) Pendeteksi Gambar Real-Time

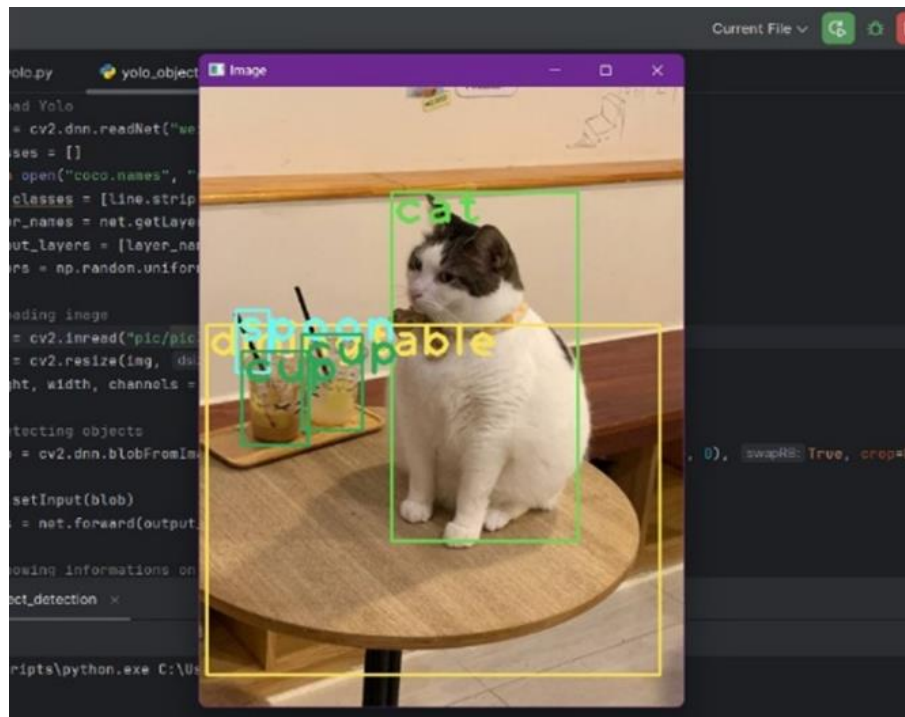
```
1 import cv2
2 import numpy as np
3 import time
4
5 # Load Yolo
6 net = cv2.dnn.readNet("weight/yolov3-tiny.weights", "cfg/yolov3-tiny.cfg")
7 classes = []
8 with open("coco.names", "r") as f:
9     classes = [line.strip() for line in f.readlines()]
10 layer_names = net.getLayerNames()
11 output_layers = [layer_names[i - 1] for i in net.getUnconnectedOutLayers()]
12 colors = np.random.uniform( low=0, high=255, size=(len(classes), 3))
13
14 # Loading image
15 cap = cv2.VideoCapture(0)
16
17 font = cv2.FONT_HERSHEY_PLAIN
18 starting_time = time.time()
19 frame_id = 0
20 while True:
21     frame = cap.read()
22     frame_id += 1
23
24     height, width, channels = frame.shape
25
26     # Detecting objects
27     blob = cv2.dnn.blobFromImage(frame, scalefactor=0.00392, size=(416, 416), mean=(0, 0, 0), swapRB=True, crop=False)
```

**Gambar 2.** Source Code Pendeteksi Gambar Real-Time

### 3.3 Implementasi

YOLO dapat mendeteksi gambar atau video maupun gambar secara langsung atau *real-time* melalui penggunaan kamera. Dengan sistem yang ada YOLO memanfaatkan kemampuan mendeteksinya untuk mendeteksi berbagai objek apa saja yang ada di depannya, berikut gambar implementasi penggunaan YOLO dan OpenCv

#### a) Mendeteksi Objek dalam Sebuah Gambar



**Gambar 3.** Mendeteksi Objek Dalam Gambar

#### b) Mendeteksi Objek Secara *Real-Time*



**Gambar 4.** Mendeteksi Objek Secara *Real-Time*

#### 4. KESIMPULAN

Sistem yang dibuat mempunyai tampilan yang mudah dimengerti sehingga mempermudah pengguna dalam pengaksesan. Setelah aplikasi dibuka, maka kamera memulai perekaman video. Aplikasi ini membutuhkan pengkoneksian internet dan nilai akurasi maksimal di angka 50%. Sistem ini bekerja dengan tepat dalam mengeksplor objek yang dalam halnya berbeda, namun dalam beberapa kemungkinan mungkin performanya tidak dapat berfungsi dengan optimal jika objek yang dideteksi sangat mendekat kamera dan juga objek yang dideteksi tidak ada pada sekumpulan data yang hadir. Objek disarankan menjauhi bingkai dari kamera sangat direkomendasikan menjauhkan jarak pandang dengan posisi yang lebih jauh. Sekumpulan data objek yang akan dipergunakan aplikasi berikut menggarap 60 jenis. Saran kelompok kami saat ini yaitu mengharap pada era masa depan setelahnya dapat melaksanakan perbaharui untuk kelengkapan fitur seperti penyajian informasi detail objek, oleh sebab itu tidak hanya menampilkan keluaran yang tertulis penyebutan objek, mungkin diperbolehkan melihatkan letak pasti posisi berada. Untuk ketinggian akurasi pada saat ini terbilang rendah maka dari itu dapat *diupgrade* untuk menambah pengalaman baik pengguna. Pada tingkatan sistem deteksi kondisi kurang cahaya atau pada kegelapan sangat membutuhkan peningkatan, agar kedepannya dapat diakses secara maksimal untuk bermacam kondisi serta dimanapun. Penjarakan antar kamera menjadikan fitur utama yang butuh dipertimbangkan tahap peningkatan aplikasi kedepannya.

#### REFERENCES

- Rosyani, P., Wirandi1, D. S., Permadi, E. D., Ardiyansyah, Prasetyo, D., & Rudin, M. (2022). Kecerdasan Buatan Alat Pendeteksi Maling Berbasis Arduino Menggunakan Sensor Ultrasonic Melalui SMS. *Scientia Sacra: Jurnal Sains, Teknologi Dan Masyarakat*, 2(2).
- Rosyani, P., Sundawa, E., Utami, M. N., Putra, A. S., & Nur, M. I. (2022). Analisis Perbandingan Metode Logika Fuzzy Untuk Menentukan Harga Penjualan/Pembelian Sepeda Motor. *BISIK : Jurnal Ilmu Komputer, Hukum, Kesehatan, Dan SosHum Hasil Penelitian Dan Pengabdian Masyarakat*, 1(1).
- Randy Moh, Aldof Faris, Dinda Desmonda Muslimah, Sriyani Mentari, Widya Ningrum Permana, Shindi Yuliani. "PENDETEKSIAN OBJEK MENGGUNAKAN OPENCV DAN METODE YOLOv4-TINY UNTUK MEMBANTU TUNANETRA" *Journal of Computer Science and Information Technology (JCSIT) Volume 1, No 2 – Maret 2024 e-ISSN : 3031-8467*.
- Taufik Hidayat, Restu Fajar Firmansyah, Muhammad Ilham, Muhammad Naufal Yazid, & Perani Rosyani. (2023). Analisis Kinerja Dan Peningkatan Kecepatan Deteksi Kendaraan Dalam Sistem Pengawasan Video Dengan Metode YOLO . *JRIIN :Jurnal Riset Informatika Dan Inovasi*, 1(2), 504–509. Retrieved from <http://jurnalmahasiswa.com/index.php/jriin/article/view/206>
- Nisan Yedidiya Sorayana Mendrofa, Ahya Mahfuzie, Muhammad Faisal, Ahmad Haidar, & Perani Rosyani. (2023). PERBANDINGAN METODE YOLO DAN FAST R-CNN DALAM SISTEM DETEKSI PENGENALAN KENDARAAN. *JRIIN :Jurnal Riset Informatika Dan Inovasi*, 1(2), 431–436. Retrieved from <http://jurnalmahasiswa.com/index.php/jriin/article/view/170>
- Rosyani, P., Maulida, A., Rahmatulloh, A., Ahussalim, I., & Mulia, R. A. J. (2023). Analisis Metode Forward Chaining pada Sistem Pakar: Systematic Literature Review. *Jurnal Manajemen, Ekonomi, Hukum, Kewirausahaan, Kesehatan, Pendidikan Dan Informatika (MANEKIN)*, 1(4).