

## PENERAPAN DEEP LEARNING YOLO UNTUK PENGUKURAN JARAK OBJEK MENGUNAKAN MONO KAMERA

**Herdianto<sup>✉</sup>, Darmeli Nasution, Niko Surya Atmaja, Syahrul Ramadhan**

Program Studi Sistem Komputer, Universitas Pembangunan Panca Budi, Medan, Indonesia

Email: [herdianto@dosen.pancabudi.ac.id](mailto:herdianto@dosen.pancabudi.ac.id)

DOI: <https://doi.org/10.46880/jmika.Vol8No1.pp51-56>

### ABSTRACT

*Mobile robots are a type of robot that can move from one place to another. Therefore, this type of robot has been given the ability to detect objects and measure estimated distances to objects around it to then carry out actions to turn left and right, forward, backward or even stop to avoid collisions. In general, to measure the distance of objects on mobile robots, ultrasonic sensors such as the HC-SR04 are used and some also use cameras, although they can be used to measure distances, but the use of these sensors has disadvantages, such as the maximum distance that can be measured is 4 meters. Given this deficiency, the research that will be carried out will try a YOLO deep learning method that can detect objects and then measure the distance of objects around them. From the results of the tests that have been carried out, it is known that the distance to the object that can be measured is 31,400 mm with the actual object height being 1750 mm.*

**Keyword: Object Detection, Deep Learning, Distance Measurement, Estimation, Monocamera.**

### ABSTRAK

*Robot mobile merupakan salah satu jenis robot yang dapat berpindah dari satu tempat ke tempat yang lain. Oleh karenanya pada robot jenis ini telah diberi kemampuan untuk mendeteksi objek dan melakukan pengukuran estimasi jarak terhadap objek di sekelilingnya untuk selanjutnya melakukan aksi berbelok ke kiri kanan, maju, mundur atau pun berhenti untuk menghindari tabrakan. Pada umumnya untuk mengukur jarak objek pada robot mobile menggunakan sensor ultrasonic seperti HC-SR04 dan ada juga yang menggunakan kamera meskipun dapat digunakan untuk melakukan pengukuran jarak tetapi penggunaan sensor ini memiliki kekurangan seperti jarak yang dapat diukur maksimum 4 meter. Dengan adanya kekurangan ini maka pada penelitian yang akan dilakukan dicoba sebuah metode deep learning YOLO yang dapat mendeteksi objek untuk selanjutnya melakukan pengukuran jarak objek yang ada di sekitarnya. Dari hasil uji coba yang telah dilakukan diketahui jarak objek yang dapat diukur sejauh 31400 mm dengan tinggi objek sebenarnya 1750 mm.*

**Kata Kunci: Deteksi Objek, Deep Learning, Pengukuran Jarak, Estimasi, Monokamera.**

### PENDAHULUAN

Servoing Visual (VS) adalah teknik untuk mengendalikan pergerakan robot menggunakan informasi umpan balik yang dikirim dari sensor penglihatan (kamera). Pada VS sangat sulit untuk melacak pergerakan objek jika data jarak objek tidak diketahui. Ada tiga jenis kontrol servo visual yaitu : servo visual berbasis posisi kontrol (PBVS), kontrol servo visual berbasis gambar (IBVS) dan hybrid (wikipedia, 2023), (Alizadeh, 2015), (Herdianto, 2008), (Herdianto, 2020). Masukan referensi untuk PBVS adalah posisi 3D dan orientasi antara objek dan robot end-effector (tangan robot) dalam ruang lingkup kendali berdasarkan koordinat cartesian. Sedangkan untuk kontrol servo visual berbasis gambar adalah posisi objek 2D yang diperoleh dari kamera. Pendekatan menggunakan IBVS kurang berhasil

dimana estimasi jarak pada objek bergerak kurang akurat dibandingkan dengan pengukuran jarak pada objek tidak bergerak terutama pada lingkungan yang dinamis sehingga mempengaruhi akurasi servoing (Firouzi & Najjaran, 2010), (Herdianto et al., 2021). Untuk menghitung jarak objek bergerak terhadap kamera maka objek harus terlebih dahulu terdeteksi oleh kamera. Ada empat metode yang dapat dilakukan dalam pendeteksian objek yaitu: berbasis model, region-based, berbasis fitur dan kontur (Alizadeh, 2015), (Nababan et al., 2017), (Herdianto & Nasution, 2023). Metode deteksi objek berbasis model membutuhkan pengetahuan tentang bentuk – bentuk objek untuk proses pencocokan ketika menemukan objek di lingkungan sekitar sebelum digunakan metode ini juga menerapkan model geometris dari objek. Tetapi metode ini memiliki kekurangan yaitu :

objek yang tidak ada dalam database tidak dapat dikenali oleh model ini dan metode ini sulit untuk diterapkan (Shaaban & Omar, 2012).

Sedangkan metode region-based dalam melakukan pendeteksian objek menggunakan bentuk 2D dari daerah objek. Pendekatan dengan metode ini sangat bergantung pada informasi yang diberikan oleh piksel-piksel dari daerah objek seperti warna dan tekstur. Untuk metode berbasis fitur cara mendeteksi objek dengan mencari fitur spesifik dari suatu objek seperti titik-titik tertentu dan deteksi tepi objek. Kelebihan dari metode berbasis fitur ini adalah algoritma sederhana dan stabil dalam mendeteksi objek tetapi metode ini tidak efisien dalam aplikasi pendeteksian objek secara real time. Metode berbasis kontur mendeteksi kontur batas objek daripada mendeteksi semua piksel yang membentuk objek. Dalam metode berbasis kontur objek dalam bingkai berikutnya yang ditangkap oleh kamera ditentukan menggunakan gerakan informasi objek tersebut. Dengan demikian bentuk dan posisi kontur objek pada bingkai selanjutnya dapat diperbaiki sesuai bentuk objek sesungguhnya. Selanjutnya informasi gerak objek diperbarui pada setiap perubahan lokasi kontur (Shaaban & Omar, 2012), (Hariyanto et al., 2019). Namun metode ini memiliki kekurangan yaitu tidak dapat mendeteksi objek sebagian. Namun demikian metode berbasis kontur memiliki kelebihan seperti komputasi lebih ringan dibandingkan dengan metode lain dan memiliki akurasi yang baik dalam mendeteksi benda diam dan bergerak. Beberapa penelitian sebelumnya terkait pengukuran jarak objek terhadap kamera telah dilakukan peneliti sebelumnya seperti yang telah dilakukan oleh (Yamaguti et al., 1997) dimana untuk menghitung jarak objek menggunakan rasio dua gambar yang diambil dari dua kamera dari lokasi yang berbeda selanjutnya dilakukan pemetaan dengan kompleks log mapping pada dua gambar tersebut. Namun model yang dihasilkan (Yamaguti et al., 1997) membutuhkan waktu komputasi yang lama dan hanya berlaku untuk objek yang sejajar dengan bidang kamera. Selanjutnya ada juga penelitian pengukuran jarak yang diusulkan oleh (M & R, 2010). Pada penelitian ini pengukuran jarak objek diawali dengan menetapkan titik awal geometri objek yang berbentuk persegi. Lalu ada juga penelitian yang dilakukan oleh (Zhang et al., 2013) yang mengembangkan algoritma 3 langkah untuk menghitung posisi 3D dari objek terhadap kamera. Dalam algoritma yang diusulkan oleh (Zhang et al., 2013) parameter kamera dikalibrasi terlebih dahulu

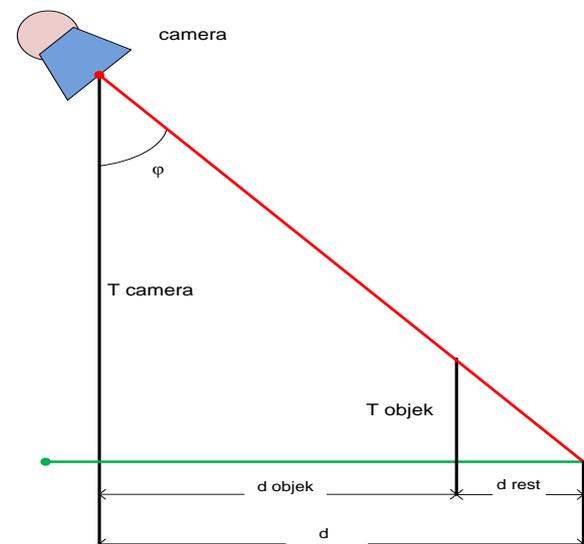
kemudian sebuah model diatur untuk mengukur jarak objek terhadap kamera yang bergerak.

Dari ke tiga penelitian yang telah diuraikan di atas peneliti menganalisis masih adanya kekurangan pada hasil ke penelitian di atas yaitu objek yang diukur bersifat tertentu dan hasil pengukuran masih tidak akurat. Oleh karena itu pada penelitian yang akan dilakukan peneliti mencoba menerapkan metode deep learning YOLO untuk mendeteksi dan melakukan pengukuran jarak objek berbasis kamera yang memiliki jangkauan jarak pengukuran lebih dari 7000 mm.

## KAJIAN LITERATUR

### Model Dasar Pengukuran Jarak

Pengukuran jarak objek terhadap kamera dapat dilakukan bila tinggi, sudut kamera diketahui seperti Gambar 1.



Gambar 1. Model Pengukuran Jarak Objek (Jüngel et al., 2008)

Dari Gambar 1 maka untuk menghitung jarak  $d$  dan  $d_{rest}$  dapat dilakukan dengan menggunakan persamaan (1) dan (2).

$$d = \tan(\varphi) * T_{camera} \quad (1)$$

dan

$$d_{rest} = \tan(\varphi) * T_{objek} \quad (2)$$

sedangkan untuk menghitung jarak  $d_{objek}$  dilakukan dengan persamaan (3).

$$d_{objek} = d - d_{rest} \quad (3)$$

atau

$$d_{objek} = \tan(\varphi) * (T_{camera} - T_{objek}) \quad (4)$$



kemudian digabungkan menjadi satu set akhir dari bounding box dan label kelas.

### METODE PENELITIAN

Untuk menyelesaikan penelitian ini maka peneliti menyusun kerangka penelitian seperti Gambar berikut:



Gambar 4. Tahapan Penelitian Yang Dilakukan

Keterangan :

- Studi Literatur  
Pada tahap ini peneliti mengumpulkan data terkait perkembangan hasil penelitian tentang pengukuran jarak objek menggunakan kamera yang telah dilakukan peneliti sebelumnya. Data yang dikumpulkan bersumber dari proseding, jurnal nasional dan internasional serta buku.
- Pengumpulan Data Citra Objek  
Tahap ini peneliti mengumpulkan sampel beberapa objek dengan jumlah tertentu dan berbagai jarak untuk dipakai pada saat proses pelatihan.
- Proses Pelatihan  
Agar deep learning dapat mendeteksi objek dan membentuk bounding box dari image citra objek maka jaringan deep learning YOLO mesti dilakukan pelatihan.
- Mengukur Tinggi Image Citra Objek

Bounding box yang terbentuk pada saat proses pelatihan selanjutnya dilakukan pengukuran tingginya.

- Membuat Formula  
Menentukan formula yang tepat sesuai variabel yang digunakan dalam pengukuran jarak.
- Pengujian  
Melakukan pengujian formula dan proses pelatihan yang telah dilakukan dengan kesesuaian hasil prediksi jarak yang diperoleh.
- Kesimpulan  
Menarik kesimpulan dari hasil pengujian yang telah dilakukan untuk mengetahui ketercapaian tujuan penelitian.

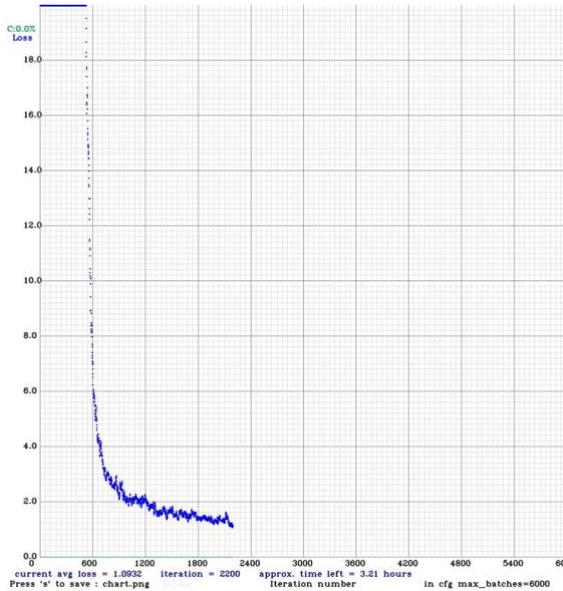
### HASIL DAN PEMBAHASAN

Agar jaringan YOLO dapat menjadi pintar dalam mengenali objek untuk selanjutnya melakukan pengukuran jarak maka langkah awal yang dilakukan adalah melakukan pelatihan terhadap citra objek yang telah dikumpulkan. Dan tujuan dari pelatihan ini adalah untuk memperoleh bobot jaringan yang ideal yang nantinya akan dipakai pada proses testing.

```
Used FMA & AVX2
OpenCV version: 4.5.2
yolov4-obj
mini_batch = 4, batch = 64, time_steps = 1, train = 1
layer  filters  size/strd(dil)  input  output
0 conv  21  3 x 3/ 1  608 x 608 x 3 -> 608 x 608 x 21 0.419 BF
1 conv  21  3 x 3/ 2  608 x 608 x 21 -> 304 x 304 x 21 0.734 BF
2 conv  21  1 x 1/ 1  304 x 304 x 21 -> 304 x 304 x 21 0.082 BF
3 route  1  -> 304 x 304 x 21
4 conv  21  1 x 1/ 1  304 x 304 x 21 -> 304 x 304 x 21 0.082 BF
5 conv  21  1 x 1/ 1  304 x 304 x 21 -> 304 x 304 x 21 0.082 BF
6 conv  21  3 x 3/ 1  304 x 304 x 21 -> 304 x 304 x 21 0.734 BF
7 Shortcut Layer: 4, wt = 0, wn = 0, outputs: 304 x 304 x 21 0.002 BF
8 conv  21  1 x 1/ 1  304 x 304 x 21 -> 304 x 304 x 21 0.082 BF
9 route  8 2  -> 304 x 304 x 42
10 conv 21  1 x 1/ 1  304 x 304 x 42 -> 304 x 304 x 21 0.163 BF
11 conv 21  3 x 3/ 2  304 x 304 x 21 -> 152 x 152 x 21 0.183 BF
12 conv 21  1 x 1/ 1  152 x 152 x 21 -> 152 x 152 x 21 0.020 BF
13 route 11  -> 152 x 152 x 21
14 conv 21  1 x 1/ 1  152 x 152 x 21 -> 152 x 152 x 21 0.020 BF
15 conv 21  1 x 1/ 1  152 x 152 x 21 -> 152 x 152 x 21 0.020 BF
16 conv 21  3 x 3/ 1  152 x 152 x 21 -> 152 x 152 x 21 0.183 BF
17 Shortcut Layer: 14, wt = 0, wn = 0, outputs: 152 x 152 x 21 0.000 BF
18 conv 21  1 x 1/ 1  152 x 152 x 21 -> 152 x 152 x 21 0.020 BF
```

Gambar 5. Bentuk Jaringan YOLO Yang Dilatih

Gambar 5 memperlihatkan jaringan YOLO yang digunakan dalam proses pelatihan dimana terlihat jaringan ini terdiri dari beberapa layer yang berisi convolusi dengan berbagai filter.



Gambar 6. Proses Pelatihan YOLO

Proses pelatihan pada gambar 6 menunjukkan perubahan nilai average loss (AVG) loss, iterasi, estimasi waktu pelatihan dan maksimum batches pelatihan. Saat pertama kali dilakukan pelatihan nilai AVG di atas 18, nilai ini masih tinggi jika untuk diterapkan. Selanjutnya nilai AVG ini terus diperbaiki hingga 0,98 dan nilai dianggap sudah ideal artinya jaringan sudah baik.

Untuk membuktikan bahwasannya jaringan hasil pelatihan sudah baik maka dilakukan proses pengujian dengan data citra yang pernah dilatih dan belum dengan persentase 70% data latih dan 30% belum pernah dilatihkan.



Gambar 7. YOLO Mendeteksi Objek Jarak 2400 mm

Pada gambar 7 YOLO dapat mendeteksi objek manusia dengan jarak objek 2400 mm dan ditandai

dengan berhasilnya YOLO membentuk bounding box warna merah pada objek manusia.



Gambar 8. YOLO Mendeteksi Objek Jarak 3400 mm

Pada gambar 8 YOLO dapat mendeteksi objek manusia dengan jarak objek 4300 mm dan ditandai dengan berhasilnya YOLO membentuk bounding box warna merah pada objek manusia.



Gambar 9. YOLO Mendeteksi Objek Jarak 9400 mm

Pada gambar 9 YOLO dapat mendeteksi objek manusia dengan jarak objek 9400 mm dan ditandai dengan berhasilnya YOLO membentuk bounding box warna merah pada objek manusia. Lalu masing – masing bounding box yang dibentuk diukur tinggi pikselnya selanjutnya dengan menggunakan persamaan 1 dihitung jaraknya.

$$Jarak = f - \left( \left( \frac{r}{1-h} \right) * (k * (r - h)) \right) \quad (11)$$

Dimana:

- r : tinggi image referensi (piksel)
- h : tinggi citra objek (piksel)
- f : focal length kamera
- k : konstanta

Tinggi bounding box yang terbentuk merupakan tinggi citra objek (h).

Tabel 1. Hasil Pengukuran Jarak Objek Manusia

No	Jarak Objek (Mm)	Pengukuran Kamera (Mm)	Error
1	18400	18820,787	-420,787
2	19400	19850,396	-450,396
3	20400	20980,769	-580,769
4	21400	21750,873	-350,873
5	22400	23100,736	-700,736
6	23400	23740,728	-340,728
7	25400	25230,591	169,409
8	26400	25990,173	409,827
9	27400	27510,387	-110,387
10	28400	27880,915	519,085
11	29400	29210,268	189,732
12	30400	30060,508	339,492
13	31400	31270,739	129,261

## KESIMPULAN

Dari hasil pengujian yang telah dilakukan diketahui deep learning YOLO dapat mendeteksi objek dan membentuk bounding box untuk selanjutnya dilakukan pengukuran jarak objek. Untuk objek manusia yang diukur dengan tinggi badan 1750 mm maka jarak yang dapat diukur hingga 31400 mm.

## DAFTAR PUSTAKA

- Alizadeh, P. (2015). *Object Distance Measurement Using a Single Camera for Robotic Applications*. Laurentian.
- Firouzi, H., & Najjaran, H. (2010). Real-time monocular vision-based object tracking with object distance and motion estimation. *IEEE/ASME International Conference on Advanced Intelligent Mechatronics, AIM*, 987–992. <https://doi.org/10.1109/AIM.2010.5695936>
- Hariyanto, E., Iqbal, M., Siahaan, A. P. U., Saragih, K. S., & Batubara, S. (2019). Comparative study of tiger identification using template matching approach based on edge patterns. *Journal of Physics: Conference Series*, 1196(1). <https://doi.org/10.1088/1742-6596/1196/1/012025>
- Herdianto. (2008). *Desain Robot Line Follower Menggunakan Mikrokontroler AT89C51*.
- Herdianto, H. (2020). *Pengaruh Penggunaan Driver Motor DC Pada Robot Line Tracer*. 4, 22–28. <https://doi.org/10.30865/komik.v4i1.2572>
- Herdianto, H., Mursyidah, M., & Rusli, R. (2021). Perancangan Washtafel Otomatis Menggunakan Mikrokontroler ATMEGA16. *Jurnal Infomedia*, 6(1), 1–6.
- Herdianto, H., & Nasution, D. (2023). Implementasi Metode CNN Untuk Klasifikasi Objek. *METHOMIKA Jurnal Manajemen Informatika Dan Komputerisasi Akuntansi*, 7(1), 54–60. <https://doi.org/10.46880/jmika.vol7no1.pp54-60>
- Jamzad, M Foroughnassiraei, A Chiniforooshan, E Ghorbani, R., Kazemi, M., Chitsaz, H. R Mobasser, F., & Sadjad, S. B. (2000). Middle sized Soccer Robots: ARVAND. *Proceeding of RoboCup-99: Robot Soccer World Cup III*, 61–73.
- Jünger, M., Mellmann, H., & Spranger, M. (2008). Improving vision-based distance measurements using reference objects. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 5001 LNAI, 89–100. [https://doi.org/10.1007/978-3-540-68847-1\\_8](https://doi.org/10.1007/978-3-540-68847-1_8)
- M, C., & R, B. (2010). *Video Camera Measuring Application Using Matlab*, "Solid State Phenomena. 166–167, 139–144.
- Nababan, E. B., Iqbal, M., & Rahmat, R. F. (2017). Breast cancer identification on digital mammogram using Evolving Connectionist Systems. *2016 International Conference on Informatics and Computing, ICIC 2016, Icic*, 132–136. <https://doi.org/10.1109/IAC.2016.7905703>
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2016-Decem*, 779–788. <https://doi.org/10.1109/CVPR.2016.91>
- Shaaban, K. M., & Omar, N. M. (2012). 3D information extraction using Region-based Deformable Net for monocular robot navigation. *Journal of Visual Communication and Image Representation*, 23(2), 397–408. <https://doi.org/10.1016/j.jvcir.2011.12.001>
- wikipedia. (2023). *Visual servoing*. [https://en.wikipedia.org/wiki/Visual\\_servoing](https://en.wikipedia.org/wiki/Visual_servoing)
- Yamaguti, N., Oe, S., & Terada, K. (1997). Method of distance measurement by using monocular camera. *Proceedings of the SICE Annual Conference*, 1255–1260. <https://doi.org/10.1109/sice.1997.624999>
- Zhang, Z., Han, Y., Zhou, Y., & Dai, M. (2013). A novel absolute localization estimation of a target with monocular vision. *Optik*, 124(12), 1218–1223. <https://doi.org/10.1016/j.ijleo.2012.03.032>