



## Algoritma YOLO sebagai deteksi korban akibat kerusakan geohazard menggunakan citra (computer vision)

**Azmi Khusnani<sup>1\*</sup>, Adi Jufriansah<sup>2</sup>, Sabarudin Saputra<sup>3</sup>**

<sup>1,2</sup> Pendidikan Fisika, IKIP Muhammadiyah Maumere, Indonesia

<sup>3</sup> Magister Teknik Informatika, Universitas Ahmad Dahlan, Indonesia

Email: husnaniazmi@gmail.com

\* Penulis korespondensi

### Informasi artikel

Sejarah artikel:  
 Dikirim 26/01/21  
 Revisi 28/01/21  
 Diterima 29/01/21

### Kata kunci:

YOLO  
 Bencana Alam  
 Deteksi Objek  
 Deep Learning  
 Convolutional Neural-  
 Network (CNN)

### Keywords:

Natural disasters  
 Object Detection  
 Deep Learning  
 Convolutional Neural-  
 Network (CNN)

### ABSTRAK

Penelitian ini bertujuan untuk melakukan identifikasi objek korban akibat kerusakan *geohazard* menggunakan algoritma YOLO. Alat yang digunakan pada penelitian adalah algoritma YOLO dengan bantuan *Google Colab. Dataset* yang digunakan berjumlah 80 objek anotasi yang terdiri dari 60 objek sebagai data latih dan 20 objek sebagai data uji dengan sumber gambar yang diperoleh dari internet. Hasil penelitian menunjukkan bahwa YOLO v4 telah mampu melakukan pendeteksian objek pada setiap objek pada gambar. Hasil ini ditunjukkan dengan munculnya *bounding box*, serta munculnya nilai akurasi. Nilai akurasi yang muncul menunjukkan hasil kerja mesin dalam identifikasi, semakin besar nilai akurasi maka menunjukkan bahwa hasil deteksi objek semakin baik.

This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license



### ABSTRACT

**YOLO algorithm as victim detection due to geohazard damage using image (computer vision).** This study aims to identify the object of the victim due to geohazard damage using the YOLO algorithm. The tool used in this research is the YOLO algorithm with the help of Google Colab. The dataset used is 80 annotation objects consisting of 60 objects as training data and 20 objects as test data with image sources obtained from the internet. The results show that YOLO v4 has been able to detect objects on every object in the image, this result is indicated by the appearance of bounding box, and the emergence of accuracy values. The accuracy value that appears shows the results of the machine's work in identification, the greater the accuracy value, the better the object detection results.

### How to Cite:

Khusnani, A., Jufriansah, A., & Saputra, S. (2021). Algoritma YOLO sebagai deteksi korban akibat kerusakan geohazard menggunakan citra (computer vision). *Berkala Fisika Indonesia: Jurnal Ilmiah Fisika, Pembelajaran Dan Aplikasinya*, 13(1), 14–21. <https://doi.org/10.12928/bfi-jifpa.v13i1.23191>.

## Pendahuluan

Ditinjau dari faktor geologis, geografis, demografis, dan klimatologis wilayah Indonesia merupakan kawasan rawan risiko bencana (Yariyan et al., 2020). Badan Nasional Penanggulangan Bencana (BNPB) mencatat setidaknya tiga ribu bencana alam terjadi sepanjang tahun 2021 (Yanuarto, 2021), yang mengakibatkan kerusakan rumah penduduk, fasilitas umum, dan korban jiwa. Pada waktu terjadinya bencana, hal terpenting yang dilakukan adalah melakukan evakuasi, lambatnya dalam mengevakuasi korban bencana menyebabkan naiknya jumlah kematian atau luka pada korban.

*Deep Learning* atau yang dikenal dengan pembelajaran mendalam merupakan bagian dari *Artificial Intelligence* dan merupakan pengembangan dari *Neural Network Multiple Layer* yang bermanfaat untuk pendeteksian pada objek, pengenalan suara, penerjemahan bahasa dan lain sebagainya (Garnelo & Shanahan, 2019; Wu et al., 2021). Pada fungsi sebagai deteksi objek, *Deep Learning* khususnya *Computer Vision* sudah banyak diterapkan pada berbagai bidang untuk melakukan analisis dan penyelesaian masalah (Arsal et al., 2020), sehingga pada penelitian ini berfungsi sebagai deteksi korban pada proses evakuasi.

*Computer Vision* adalah suatu ilmu yang populer dalam bidang *Deep Learning*. Hal tersebut didasari oleh cara pandang komputer terhadap objek yang diamati dan ditangkap menggunakan ilmu optika dalam bidang fisika dan sensor (Manajang et al., 2020). Fisika merupakan bidang lain yang terintegrasi dan erat kaitannya dengan *Computer Vision*. Sistem *Computer Vision* bergantung pada sensor gambar yang mendeteksi radiasi elektromagnetik yang disajikan dalam bentuk cahaya tampak atau inframerah (Iqbal & Trinugroho, 2008). Sensor dirancang dengan menerapkan hukum fisika yang berkaitan dengan fisika *solid state*. Di mana proses tersebut dijelaskan dengan perambatan cahaya. Sensor gambar yang digunakan bahkan menggunakan konsep mekanika kuantum untuk memberikan informasi yang lengkap terhadap proses pembentukan gambar, khususnya saat digunakan untuk keperluan proses evakuasi yang membutuhkan kualitas gambar yang kuat.

Evakuasi korban bencana di Indonesia sendiri melibatkan Badan Nasional Pencarian dan Pertolongan (BASARNAS) melalui tim *Search and Rescue* (SAR). Namun, teknologi yang digunakan belum dapat mempersingkat waktu pencarian. Sehingga dengan mampu menganalisa keadaan sekitar, *Computer Vision* dapat membantu proses evakuasi dalam menemukan korban. Deteksi objek korban bencana dapat dilakukan dengan menggunakan pengolahan citra (Aulia et al., 2021). Pada proses mendeteksi suatu objek maka akan berkaitan dengan identifikasi pada benda-benda yang ada di dunia nyata, seperti manusia, hewan, rumah, dan lainnya. Sehingga pada tahap pemrosesan dilakukan data latih objek pada sistem. Salah satu algoritma yang dapat digunakan pada deteksi objek adalah YOLO. YOLO merupakan salah satu metode deteksi objek tercepat dengan kinerja yang baik dan akurasi tinggi (Fang et al., 2020; Sarosa & Muna, 2021). Berdasarkan uraian di atas maka penelitian ini bertujuan untuk melakukan identifikasi objek korban akibat kerusakan *geohazard* menggunakan algoritma YOLO.

## Metode

Pada penelitian ini proses pengolahan citra menggunakan algoritma YOLO v4. Gambar diperoleh dari sumber internet <https://fokusjabar.id> (bencana banjir), <https://www.dw.com> (gempa bumi dan tsunami) dan [www.pexels.com](http://www.pexels.com). Dataset pada penelitian ini berjumlah 80 objek anotasi yang terdiri dari 60 objek sebagai data latih dan 20 objek sebagai data uji. Pada anotasi data latih disajikan sebagai berikut:

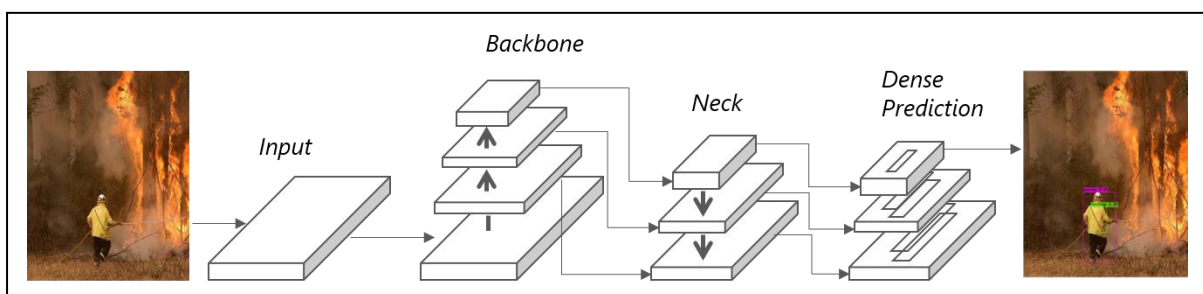
```
classes= 80
train = data/coco/trainvalno5k.txt
valid = data/coco_testdev
#valid = data/coco_val_5k.list
names = data/coco.names
backup = backup/
eval=coco
```

Langkah selanjutnya melakukan anotasi pada data masing-masing data latih. Anotasi berfungsi sebagai *bounding box* setiap objek citra berdasarkan kelas (AL-Alimi et al., 2020). Format anotasi yang digunakan disajikan pada Tabel 1.

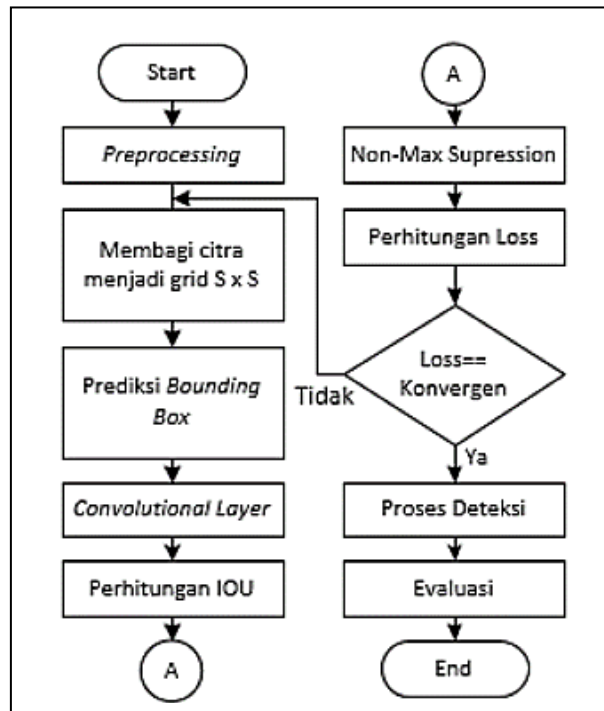
Tabel 1. Format anotasi

Anotasi	Keterangan
<object-class>	Integer number dari kelas objek (0, 1, 2, ...)
<x> <y>	Menentukan nilai titik tengah
<width>	Lebar <i>box</i>
<height>	Tinggi <i>box</i>

Arsitektur YOLO v4 memiliki satu tahap yang terdiri dari empat *block* meliputi *input*, *backbone*, *neck*, dan *dense prediction* yang ditampilkan pada Gambar 1, sedangkan diagram alir pada deteksi ditampilkan pada Gambar 2.



Gambar 1. Arsitektur YOLO v4



Gambar 2. Diagram alir algoritma YOLO v4 (Asshiddiqie et al., 2020)

Pada proses instalasi YOLO v4 menggunakan *tools* Google Colab dengan *Graphic Processing Unit* (GPU) yang diaktifkan sebagai berikut:

```

nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2020 NVIDIA Corporation
Built on Mon_Oct_12_20:09:46_PDT_2020
Cuda compilation tools, release 11.1, V11.1.105
Build cuda_11.1.TC455_06.29190527_0

```

```

-----
#define CUDNN_MAJOR 7
#define CUDNN_MINOR 6
#define CUDNN_PATCHLEVEL 5

```

```

-----
#define CUDNN_VERSION (CUDNN_MAJOR * 1000 + CUDNN_MINOR * 100 + CUDNN_PATCHLEVEL)
#include "driver_types.h"

```

## Hasil dan Pembahasan

YOLO merupakan salah satu jenis algoritma yang memiliki pendekatan yang berbeda dan memiliki akurasi tinggi (Fang et al., 2020). Perbedaan ini ditunjukkan dengan penerapan jaringan syaraf tunggal pada keseluruhan gambar yang nantinya jaringan ini akan membagi gambar menjadi wilayah-wilayah. Pembagian wilayah-wilayah ini akan memprediksi *bounding box* atau kotak pembatas dan probabilitas untuk setiap kotak wilayah pembatas. Selanjutnya akan ditimbang probabilitasnya untuk pengklasifikasian sebagai objek atau bukan (Huang et al., 2018; Jana et al., 2018). Pada bagian akhir akan dipilih kotak pembatas dengan nilai yang paling tinggi untuk di jadikan sebagai pemisah objek satu dengan yang lain seperti pada Gambar 3. Gambar 3 menampilkan hasil data uji, dimana *bounding box* berwarna ungu dengan identifikasi objek serta menampilkan nilai akurasi. Jumlah *bounding box* saat

deteksi objek dengan YOLO v4 akan beragam sesuai jumlah objek yang terbaca dan akan memiliki warna yang berbeda dengan jenis klasifikasi objek yang terdeteksi (Niu et al., 2021).



Gambar 3. Hasil data uji menggunakan YOLO v4

Deteksi merupakan usaha mengidentifikasi terkait keadaan suatu objek, Hal ini memiliki nilai kerumitan yang lebih kompleks dibandingkan dengan pengklasifikasian. Hal ini dikarenakan klasifikasi hanya mampu mengenali objek tanpa memberikan informasi letak objek secara tepat berada. Sebaliknya pada deteksi objek dengan YOLO hanya menggunakan lapisan konvolusi dan lapisan *pooling*, di mana untuk lapisan konvolusi terakhir akan disesuaikan dengan jumlah kelas dan jumlah kotak prediksi yang diinginkan (Fu et al., 2019; Yang & Jiachun, 2018).

Pembacaan anotasi citra kerusakan akibat *geohazard* telah didapatkan, seperti yang ditampilkan pada Gambar 4 dan Gambar 5. Gambar 4 menampilkan bencana *geohazard* akibat banjir bandang di mana Yolo v4 memberikan hasil pembacaan setiap citra sesuai dengan data anotasi. Sedangkan pada citra kerusakan gempa bumi dan tsunami pada Gambar 5 telah dapat membaca anotasi citra lebih kompleks sesuai jumlah kelas. Setiap pembacaan citra berupa *bounding box* di mana di dalamnya telah tertulis data yang telah terbaca seperti *person*, *car*, dan lainnya sesuai isi gambar (Tabel 2).



Gambar 4. Deteksi korban bencana banjir dengan YOLO v4



Gambar 5. Deteksi korban bencana gempa bumi-tsunami dengan YOLO v4

Hasil iterasi YOLO v4 berbantuan *Google Colab* didapatkan data di bawah ini, di mana pada contoh data ini sesuai dengan Gambar 5.

```

CUDA-version: 11010 (11020), cuDNN: 7.6.5, CUDNN_HALF=1, GPU count: 1
CUDNN_HALF=1
OpenCV version: 3.2.0
0 : compute_capability = 750, cudnn_half = 1, GPU: Tesla T4
net.optimized_memory = 0
mini_batch = 1, batch = 8, time_steps = 1, train = 0
  layer filters size/strd(dil)  input      output
  0 Create CUDA-stream - 0
  Create cudnn-handle 0
conv  32   3 x 3/ 1  608 x 608 x 3 -> 608 x 608 x 32 0.639 BF
  1 conv  64   3 x 3/ 2  608 x 608 x 32 -> 304 x 304 x 64 3.407 BF
  2 conv  64   1 x 1/ 1  304 x 304 x 64 -> 304 x 304 x 64 0.757 BF
  3 route 1
  4 conv  64   1 x 1/ 1  304 x 304 x 64 -> 304 x 304 x 64 0.757 BF
  5 conv  32   1 x 1/ 1  304 x 304 x 64 -> 304 x 304 x 32 0.379 BF
...
155 conv 1024  3 x 3/ 1  19 x 19 x 512 -> 19 x 19 x 1024 3.407 BF
156 conv  512  1 x 1/ 1  19 x 19 x 1024 -> 19 x 19 x 512 0.379 BF
157 conv 1024  3 x 3/ 1  19 x 19 x 512 -> 19 x 19 x 1024 3.407 BF
158 conv  512  1 x 1/ 1  19 x 19 x 1024 -> 19 x 19 x 512 0.379 BF
159 conv 1024  3 x 3/ 1  19 x 19 x 512 -> 19 x 19 x 1024 3.407 BF
160 conv  255  1 x 1/ 1  19 x 19 x 1024 -> 19 x 19 x 255 0.189 BF

```

Berdasarkan data di atas terlihat bahwa jumlah iterasi sebanyak 160 iterasi. Tabel 2 menjelaskan identifikasi data hasil anotasi citra dengan YOLO v4 berbantuan *Google Colab*, dimana presentasi hasil akurasi menjelaskan tingkat kebenaran objek yang terbaca. Nilai hasil akurasi akan semakin baik apabila tingkat kualitas gambar yang semakin baik (Jana et al., 2018). Seperti pada deteksi objek *boat* memiliki hasil 99%. Hasil presentasi akan berbeda jika pada data objek di dalam citra kecil dan tidak jelas maka nilai presentasi akan semakin menurun seperti pada data objek *person* 29% dan *motorbike* 29%. Jumlah

objek yang terbaca pada data citra dalam Tabel 1 terdapat pengulangan. Hal ini sesuai dengan hasil deteksi objek yang berbeda namun memiliki jenis yang sama.

Tabel 2. Identifikasi data anotasi citra

Data Citra (Flage)	Hasil Akurasi (%)
Boat	40
Boat	99
Car	67
Car	72
Person	42
Person	29
motorbike	27
Person	30
Person	32
Person	42
Motorbike	40
Person	29
Motorbike	46
Motorbike	36

## Simpulan

Berdasarkan hasil uraian pada penelitian ini, diketahui bahwa deteksi objek menggunakan YOLO v4 telah berhasil dilakukan. Hasil deteksi pada setiap ragam objek pada citra telah terlihat dengan pemetaan dengan *bounding box* setiap objek. Munculnya nilai akurasi menunjukkan bahwa deteksi objek dengan YOLO v4 telah bekerja secara optimal sesuai data anotasi citra yang diberikan. Pemanfaatan deteksi objek ini akan sangat bermanfaat pada deteksi korban bencana pada saat evakuasi serta dapat meminimalkan waktu pencarian dan jumlah korban akibat telah terpetakan pada sistem deteksi objek dengan YOLO v4. Pengoptimalan sebagai kelanjutan penelitian dapat dilakukan dengan mengintegrasikan pada bentuk teknologi seperti *drone robot*.

## Referensi

- AL-Alimi, D., Shao, Y., Alalimi, A., & Abdu, A. (2020). Mask R-CNN for geospatial object detection. *International Journal of Information Technology and Computer Science*, 12(5), 63–72. <https://doi.org/10.5815/ijitcs.2020.05.05>
- Arsal, M., Wardijono, B. A., & Anggraini, D. (2020). Face recognition untuk akses pegawai bank menggunakan deep learning dengan metode CNN. *Jurnal Nasional Teknologi Dan Sistem Informasi*, 6(1), 55–63. <https://doi.org/10.25077/teknosi.v6i1.2020.55-63>
- Asshiddiqie, M. A. J., Rahmat, B., & Anggraeny, F. T. (2020). Deteksi tanaman tebu pada lahan pertanian menggunakan metode convolutional neural network. *Jurnal Informatika Dan Sistem Informasi*, 1(1), 229–237.
- Aulia, D. S., Setianingsih, C., & Kallista, M. (2021). Deteksi tanda kehidupan pada korban bencana alam dengan algoritma YOLO dan open pose. *E-Proceeding of Engineering*, 8(5), 6162–6174.
- Fang, W., Wang, L., & Ren, P. (2020). Tinier-YOLO: A real-time object detection method for constrained environments. *IEEE Access*, 8, 1935–1944. <https://doi.org/10.1109/ACCESS.2019.2961959>
- Fu, K., Chen, Z., Zhang, Y., & Sun, X. (2019). Enhanced feature representation in detection for optical remote sensing images. *Remote Sensing*, 11(18), 1–17. <https://doi.org/10.3390/rs11182095>
- Garnelo, M., & Shanahan, M. (2019). Reconciling deep learning with symbolic artificial intelligence: Representing objects and relations. *Current Opinion in Behavioral Sciences*, 29, 17–23. <https://doi.org/10.1016/j.cobeha.2018.12.010>

- Huang, R., Pedoeem, J., & Chen, C. (2018). YOLO-LITE: A real-time object detection algorithm optimized for non-GPU computers. *Proceedings - 2018 IEEE International Conference on Big Data, Big Data 2018*, 2503–2510. <https://doi.org/10.1109/BigData.2018.8621865>
- Iqbal, M., & Trinugroho, I. A. (2008). Perkembangan riset aplikasi polarization imaging by reflection untuk objek transparan dalam bidang computer vision. *Seminar Nasional Aplikasi Teknologi Informasi, 2008(Snati)*, 1–6.
- Jana, A. P., Biswas, A., & Mohana. (2018). YOLO Based detection and classification of objects in video records. *2018 3rd IEEE International Conference on Recent Trends in Electronics, Information and Communication Technology, RTEICT 2018 - Proceedings*, 2448–2452. <https://doi.org/10.1109/RTEICT42901.2018.9012375>
- Manajang, D. J. P., Jacobus, A., Elektro, J. T., Sam, U., & Manado, R. (2020). Implementasi framework tensorflow object detection API dalam mengklasifikasi jenis kendaraan bermotor. *Jurnal Teknik Informatika*, 15(3), 171–178. <https://doi.org/10.35793/jti.15.3.2020.29775>
- Niu, Y., Xu, Z., Xu, E., Li, G., Huo, Y., & Sun, W. (2021). Monocular pedestrian 3D localization for social distance monitoring. *Sensors*, 21(17), 1–16. <https://doi.org/10.3390/s21175908>
- Sarosa, M., & Muna, N. (2021). Implementasi algoritma you only look once (YOLO) untuk deteksi korban bencana alam. *Jurnal Teknologi Informasi Dan Ilmu Komputer (JTIIK)*, 8(4), 787–792. <https://doi.org/10.25126/jtiik.202184407>
- Wu, X., Li, W., Hong, D., Tao, R., & Du, Q. (2021). Deep learning for unmanned aerial vehicle-based object detection and tracking: A survey. *IEEE Geoscience and Remote Sensing Magazine*, 1–24. <https://doi.org/10.1109/MGRS.2021.3115137>
- Yang, W., & Jiachun, Z. (2018). Real-time face detection based on YOLO. *1st IEEE International Conference on Knowledge Innovation and Invention, ICKII 2018, 2*, 221–224. <https://doi.org/10.1109/ICKII.2018.8569109>
- Yanuarto, T. (2021). *Catatan refleksi akhir tahun penanggulangan bencana 2021*.
- Yariyan, P., Zabihi, H., Wolf, I. D., Karami, M., & Amiriyan, S. (2020). Earthquake risk assessment using an integrated fuzzy analytic hierarchy process with artificial neural networks based on GIS: A case study of Sanandaj in Iran. *International Journal of Disaster Risk Reduction*, 50(101705), 1–62. <https://doi.org/10.1016/j.ijdrr.2020.101705>