

# DETEKSI TANDA KEHIDUPAN PADA KORBAN BENCANA ALAM DENGAN ALGORITMA FASTER R-CNN DAN OPEN POSE (LIFE SIGN DETECTOR USING FASTER REGION CONVOLUTIONAL NEURAL NETWORK (FASTER R-CNN) AND OPEN POSE)

Nesa Cristin Natalia Batubara<sup>1</sup>, Casi Setianingsih<sup>2</sup>, Meta Kallista<sup>3</sup>

<sup>1,2,3</sup> Universitas Telkom, Bandung

nesacristin@student.telkomuniversity.ac.id<sup>1</sup>, setiacasie@telkomuniversity.ac.id<sup>2</sup>,

metakallista@telkomuniversity.ac.id<sup>3</sup>

---

## Abstrak

Pengolahan citra dalam implementasinya digunakan untuk efisiensi pekerjaan yang berhubungan dengan pengolahan citra atau gambar. Dalam penelitian ini pengolahan citra diimplementasikan untuk mendeteksi korban bencana alam keadaan masih hidup atau tidak. Sistem pada penelitian ini dapat mendeteksi korban bencana alam dan melihat pergerakan dengan menampilkan grafik yang membedakan korban masih hidup atau tidak dengan keadaan terbaring dalam pencahayaan yang cukup dan bagian dada atau punggung yang terlihat karena sistem mendeteksi gerakan dada atau punggung dari rentang jarak 4 sampai 6 meter. Faster R-CNN sebagai metode deteksi ciri objek dalam hal ini untuk mengenali korban bencana alam menggunakan Bahasa Python beserta library-library pembelajaran mesin, seperti Tensorflow dan OpenCV. Dataset yang digunakan adalah dataset custom yang berjumlah 400 yaitu masing masing 347 kelas manusia dan 53 kelas bukan manusia. Model terbaik yang digunakan setelah melewati proses pengujian matrix adalah model yang mendapatkan nilai akurasi 93%, recall 99%, presisi 93%, step 150000, epochs 300 dan learning rate 0,002.

Sistem akan mendeteksi pergerakan dada atau punggung menggunakan motion detection dengan cara mengetahui titik bahu melalui open pose maka bisa membangun RoI untuk mendeteksi pergerakan. Setelah melakukan percobaan pada 1 manekin dan 3 manusia akurasi deteksi gerakan sebesar 100%.

Kata kunci : Object detection, Faster R-CNN, RoI, Motion Detection, Open pose

---

## Abstract

Image processing in its implementation is used for the efficiency of work related to image processing or images. In this study, image processing is implemented to detect victims of natural disasters, whether they are alive or not. The system in this study can detect victims of natural disasters and see their movements by displaying a graph that distinguishes the victim is alive or not from lying in a state of adequate lighting and the chest or back is visible because the system detects chest or back movements from a distance of 4 to 6 meters. . Faster R-CNN as an object feature detection method in this case to identify victims of natural disasters using Python and machine learning libraries, such as Tensorflow and OpenCV. The dataset used is a custom dataset, totaling 400, namely 347 human classes and 53 non-human classes. The best model used after passing the matrix testing process is the model that gets 93% accuracy, 99% recall, 93% precision, 150000 steps, 300 epochs and 0,002 learning rate.

The system will detect the movement of the chest or back using motion detection by knowing the point of the shoulder through an open pose so that it can build RoI to detect movement. After experimenting with 1 mannequin and 3 humans, the motion detection accuracy was 100%.

Keywords: Object detection, Faster R-CNN, RoI, Motion Detection, Open pose

---

## 1. Pendahuluan

Bencana alam adalah aktivitas alam yang menyebabkan kerugian pada aktivitas makhluk hidup dan pada umumnya terdapat korban. Korban meninggal bencana bisa bertambah terjadi karena beberapa faktor yaitu bencana alam tidak terprediksi dan proses penyelamatan yang lama dan manual. Lokasi bencana alam yang tidak terjangkau juga mempengaruhi komunikasi korban dan tim penyelamat. Oleh sebab itu, dalam mengoptimalkan waktu penyelamatan korban yang masih hidup dibutuhkan alat untuk *monitoring* wilayah bencana menemukan dan melihat keadaan korban masih hidup atau tidak. Semakin cepat korban ditemukan maka menekan jumlah korban jiwa yang meninggal saat bencana alam.

Berdasarkan persoalan tersebut, maka penelitian dirancang untuk membuat sebuah sistem yang dapat mendeteksi korban bencana alam menggunakan metode Faster R-CNN dan tanda kehidupan dapat mengenali manusia pada sebuah lokasi bencana dan mendeteksi korban bencana alam masih hidup atau tidak menggunakan Open Pose yang telah terlatih menggunakan data latih yang tersedia. Pada penelitian sebelumnya, dengan menerapkan algoritma CNN untuk menemukan korban bencana alam tanpa mengetahui kondisi korban masih hidup atau tidak.

Tugas akhir dilakukan dengan tujuan untuk mengembangkan penelitian sebelumnya namun menggunakan metode yang berbeda. Pada penelitian ini, digunakan library OpenCV sebagai *object detector* dan tensorflow sebagai library *machine learning*, yang nantinya akan digunakan juga algoritma klasifikasi Faster R-CNN untuk pendeteksian manusia dan Open pose yang dilengkapi dengan RoI untuk membaca pergerakan pada dada atau punggung. Pengujian yang dilakukan penulis menggunakan sampel video yang direkam menggunakan *drone*.

## 2. Dasar Teori

### 2.1 Object Detection

*Object detection* adalah merupakan proses mencari dan menemukan suatu objek dalam sebuah gambar maupun video dengan menentukan *bounding box*, sekaligus fungsi dari *object detection* untuk mengenali objek dan menentukan posisi koordinat pada objek. *Object detection* dapat melakukan prediksi pada kelas objek, kotak pembatas dan koordinat untuk setiap objek[1]. Konsep umum *Object detection* adalah melakukan *scanning* pada seluruh bagian gambar atau video lalu melakukan keputusan terhadap objek yang ingin di deteksi dan tidak dideteksi.

### 2.2 Pengolahan Citra Digital

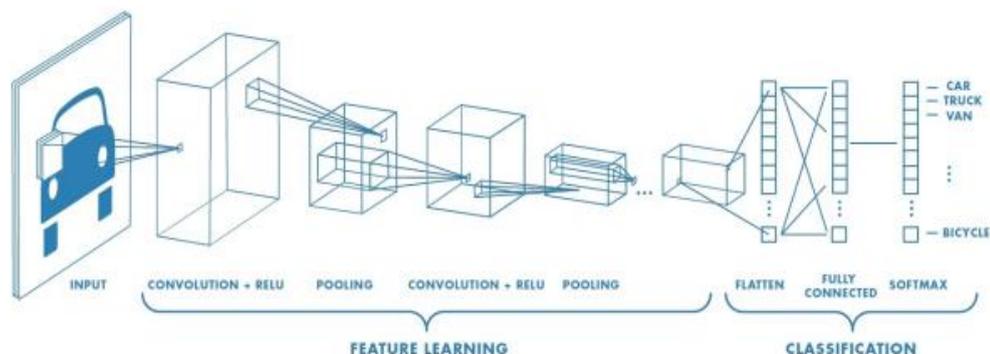
Pengolahan citra digital adalah ilmu yang mempelajari teknik dalam mengolah gambar dengan mengolah gambar yang diam atau hasil capture dari gambar bergerak yang diolah menggunakan komputer [2]. Citra merupakan fungsi kontinu dengan intensitas cahaya pada bidang dua dimensi. Agar citra dapat diolah menggunakan komputer maka citra harus direpresentasikan ke dalam bentuk numerik dengan nilai diskrit. Representasi dari fungsi kontinu menjadi nilai-nilai diskrit disebut digitalisasi citra.

Pengolahan citra merupakan aktivitas untuk mengubah, menganalisa atau memperbaiki suatu gambar. Berdasarkan tujuan transformasi operasi suatu citra dikategorikan dalam berbagai jenis [3]:

- Peningkatan Kualitas Citra (*Image Enhancement*) untuk meningkatkan fitur tertentu pada citra.
- Pemulihan Citra (*Image Restoration*) untuk mengembalikan kondisi citra pada kondisi yang diketahui sebelumnya akibat adanya noise menyebabkan penurunan kualitas citra.
- Pemampatan Citra (*Image Compression*) untuk menampilkan citra dalam bentuk yang lebih rapat menyebabkan penggunaan memori yang lebih sedikit.
- Segmentasi Citra (*Image Segmentation*) untuk membedakan suatu citra ke dalam suatu kriteria yang dapat dihitung seperti warna, intensitas, tekstur dan lain lain.
- Pengorakan Citra (*Image Analysis*) untuk menghitung sampel citra dalam besaran kuantitatif untuk menemukan hasil analisisnya.
- Rekonstruksi Citra (*Image Reconstruction*) untuk menampilkan objek baru hasil dari beberapa proyeksi citra sebelumnya.

### 2.3 Convolutional Neural Network

*Convolution Neural Network* (CNN) adalah salah satu algoritma dari *deep learning* yang merupakan pengembangan dari *Multilayer perception* (MLP) yang dirancang untuk mengolah data dalam bentuk dua dimensi, misalnya gambar atau suara [4]. CNN digunakan untuk membuat klasifikasi data yang telah diberi label. Lapisan CNN memiliki susunan neuron 3 dimensi (lebar, tinggi, kedalaman). Lebar dan tinggi merupakan ukuran lapisan sedangkan kedalaman mengacu pada jumlah lapisan. Sebuah CNN dapat memiliki puluhan hingga ratusan lapisan yang masing masing mempelajari mendeteksi berbagai fitur pada gambar. Penerapan pengolahan citra diterapkan pada setiap citra data latih pada resolusi yang berbeda dan *output* dari masing-masing gambar yang diolah dan dipakai pada lapisan berikutnya sebagai input.



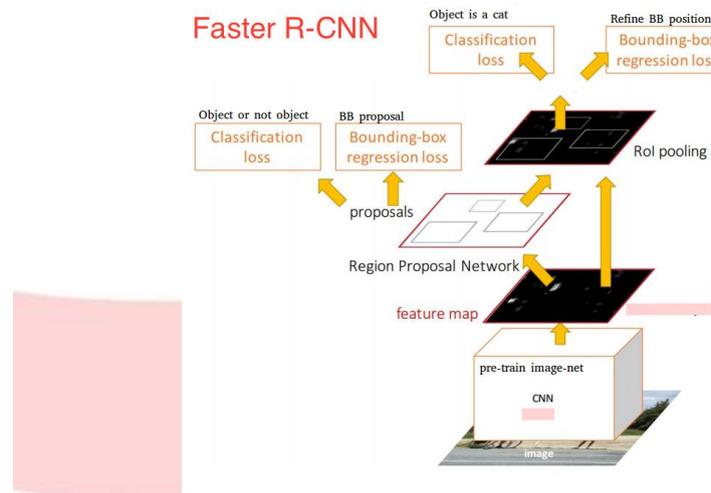
Gambar 1 Alur pada CNN dalam mempelajari citra [4].

### 2.4 Faster Region Convolutional Neural Network

Faster R-CNN digunakan untuk *object detection*, dalam penelitian ini, sebagai pengklasifikasi objek, yaitu manusia. Faster R-CNN memiliki 2 modul utama, yaitu *Deep Fully Convolutional Network* yang mengandung *Region Proposed Network* (RPN) dan modul sebagai detektor *Fast R-CNN* [5].

### 2.4.1 Region Proposed Network (RPN)

RPN adalah *convolutional neural network* yang disatukan pada suatu *region proposal* dan *featured map*. RPN berfungsi untuk menandai keberadaan objek dan mengidentifikasi suatu objek yang ditentukan dalam suatu kelas. RPN bekerja dengan memeriksa objek menggunakan *region proposal* dengan jumlah yang banyak kemudian menemukan kemunculan objek dan *background* pada suatu citra uji [5].



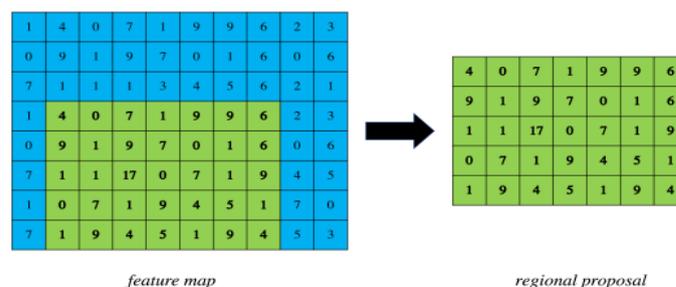
Gambar 2 Skema proses Faster R-CNN dalam deteksi objek [4].

Berdasarkan skema pada gambar di atas menunjukkan proses kerja dari *Faster R-CNN* bahwa lapisan input akan dibagikan dengan lapisan konvolusi lalu menghitung batasan wilayah yang dipilih. RPN akan menangkap semua gambar dari berbagai ukuran dan bentuk untuk memprediksi kumpulan objek dan bukan objek yang diusulkan oleh fitur citra. Kotak yang sudah berhasil dibentuk kemudian akan diekstraksi menggunakan menggunakan RoI untuk setiap temuan fitur. Selanjutnya, hasil dari vektor fitur dimasukkan ke dalam dua *layer* yang terhubung sepenuhnya untuk mendeteksi apakah objek tersebut dalam penelitian adalah korban dan membentuk daerah kotak fokus pendeteksian.

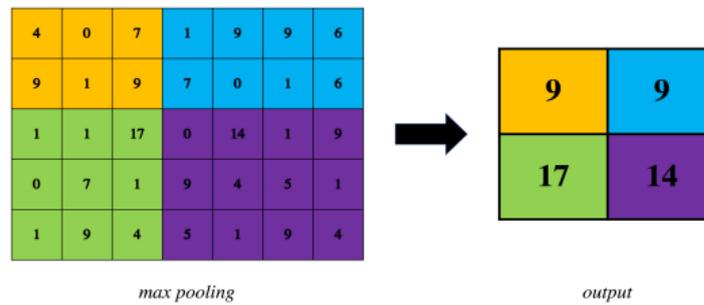
Tahap awal adalah deteksi *background* oleh algoritma RPN. Setelahnya pendeteksian akan dilakukan untuk mendapatkan objek. Hasilnya akan berupa nilai besaran objek yang menunjukkan bahwa pada area tersebut memiliki potensi terbesar terdapat objek dan mengeluarkan hasil regresi yang menunjukkan posisi objek tersebut. Posisi tersebut ditandai dengan *bounding box* yang memiliki *xmin,ymin,width* dan *height*.

### 2.4.2 Region of Interest (RoI) Pooling

RoI bekerja pada setiap *region of interest* dari input yang ada dengan menggunakan *featured map* yang memiliki skala yang sama besarnya dengan *region proposal* yang ditentukan. Penskalaan ini dilakukan dengan cara *max pooling* [6]. *Max Pooling* bertujuan untuk menghasilkan dimensi yang sama antar *output*. *Max Pooling* dapat meminimalisir nilai citra dengan menemukan nilai terbesar dalam *feature maps* dan memutuskan nilai terbesar sebagai *output section*. Fungsi utama *RoI pooling* adalah mempercepat *processing* apabila terdapat banyak objek dalam satu frame yang sama. Proses ini membuat efisiensi waktu lebih cepat seperti pada gambar 2.8.



Gambar 3 Ilustrasi pengambilan nilai citra berdasarkan feature map membentuk regional proposal [6].



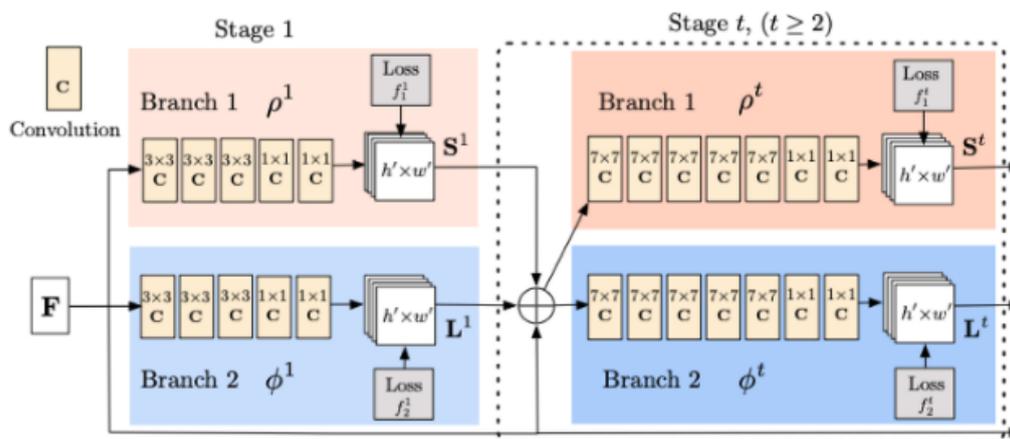
Gambar 4 Operasi pada max pooling untuk membentuk *output* citra baru [6].

Berdasarkan gambar 2.9 menunjukkan ketika daerah proposal telah dibuat dalam jumlah yang banyak, maka seluruh daerah proposal akan diubah ke dalam satu frame pada setiap *feature map*. *Region of Interest Pooling layer* menerima dua input, yaitu:

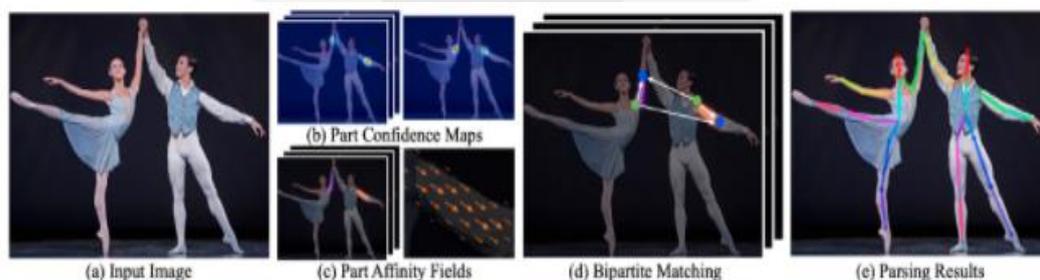
1. Sebuah *feature map* yang diproduksi dari *Convolutional Layer* setelah melewati berbagai operasi konvolusi dan *pooling*.
2. Jumlah proposal yang diproduksi oleh RPN.

### 2.5 Open Pose

OpenPose pertama kali diperkenalkan oleh Carnegie Mellon University, yang berdasar pada salah satu model CVPR 2017 *Estimation Pose multi-person realtime*. Openpose menggunakan jumlah komputasi yang sangat besar sehingga akan efektif apabila dijalankan pada GPU dengan kecepatan frame yang lebih rendah[7]. Gambar 2.10 di bawah ini adalah untuk merepresentasikan alur pada open pose dalam membentuk skeleton pada objek.



Gambar 5 Arsitektur Open Pose menggunakan “two-branch multi-stage” [8].



Gambar 6 Ilustrasi tahapan Open Pose berdasarkan arsitektur yang digunakan [8].

Gambar 5 dan 6 merupakan arsitektur dari *two-branch multi-stage* CNN Openpose yang berarti “Dua Cabang” maka menghasilkan dua *output* yang berbeda. ‘*Multi Stage*’ berarti *network* yang terkait akan ditumpuk satu sama lain di setiap tahap.

1. *Two Branch*

Pada tahap ini, *output* dari cabang atas yang warna jingga memprediksi “*confidence maps*” (gambar 6 bagian b) dari lokasi bagian tubuh yang berbeda seperti mata kanan, mata kiri, siku kanan dan lainnya.

## 2. Multi Stage

Pada tahap pertama, *output* dari network ini memberikan prediksi “confidence maps” disimbolkan dengan S dan juga “*affinity fields*” disimbolkan dengan L. Lalu, tahap berikutnya (sebelah kanan dari 5), prediksi dari 2 cabang di tahap sebelumnya dan fitur dari gambar F akan digabungkan dan hasil *output* akan dijadikan input di tahap berikutnya.

## 2.6 Confusion Matrix

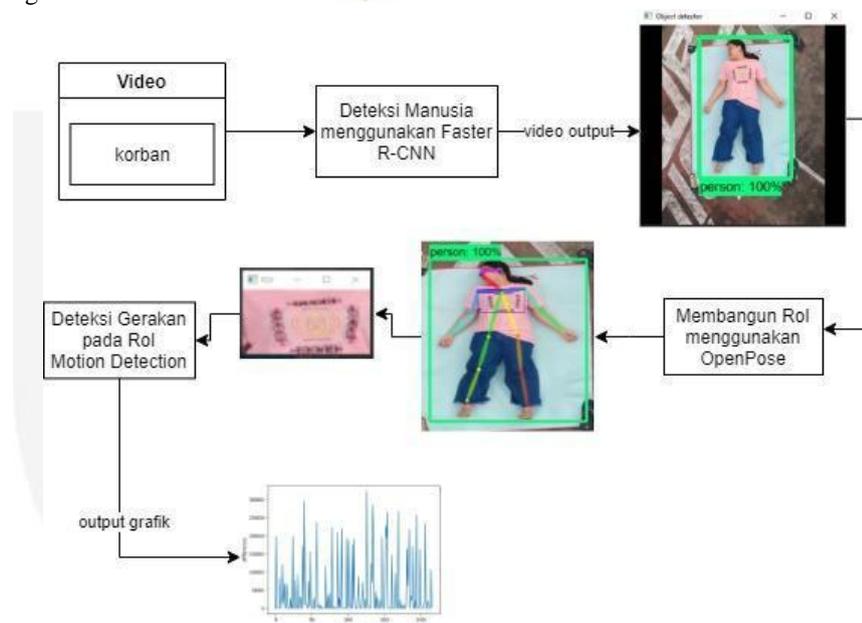
*Confusion matrix* adalah suatu metode penyampaian informasi mengenai prediksi dan kebenaran dari suatu sistem klasifikasi yang mempunyai label kebenaran [9]. *Confusion matrix* menampilkan data dalam bentuk matrik yang ditampilkan, akan ada data data yang diperoleh, yaitu:

- A) *True Positive* (TP): kondisi model memprediksi data benar dan prediksinya adalah benar.
- B) *True Negative* (TN): kondisi model memprediksi suatu data adalah salah dan prediksinya benar.
- C) *False Positive* (FP): kondisi model memprediksi suatu data adalah benar dan prediksinya salah.
- D) *False Negative* (FN): kondisi model memprediksi suatu data adalah salah tetapi prediksinya salah.

## 3. Perancangan Sistem

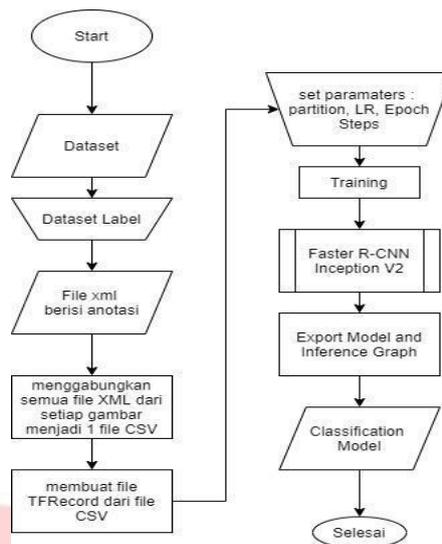
### 3.1 Desain Sistem

Desain sistem yang dirancang untuk mendeteksi tanda kehidupan pada korban bencana alam dilakukan dengan tahapan klasifikasi objek manusia sebagai korban melalui video. Jika ada manusia maka hasil deteksi manusia akan disimpan untuk membangun *Region of Interest* (RoI) yang berbentuk persegi pada bagian dada atau punggung pada objek menggunakan Open Pose dan membaca gerakan pada RoI menggunakan *motion detection* dengan cara membaca jumlah perubahan pixel. Berikut adalah gambaran sistem yang diusulkan:



Gambar 7 Gambaran umum sistem.

### 3.2 Diagram Alur Deteksi Manusia menggunakan Faster R-CNN



Gambar 8 Diagram alir proses deteksi manusia menggunakan Faster R-CNN.

Gambar 8 merupakan *flowchart* dari tahap tahap yang akan dilakukan pada *training*. Dataset yang akan melalui beberapa tahap untuk mengoptimalkan proses *training* pada citra. Proses *training* menghasilkan model berdasarkan dataset yang sudah dilatih yang akan digunakan pada klasifikasi deteksi objek.

### 3.3 Deteksi Gerakan Dada atau Punggung Objek



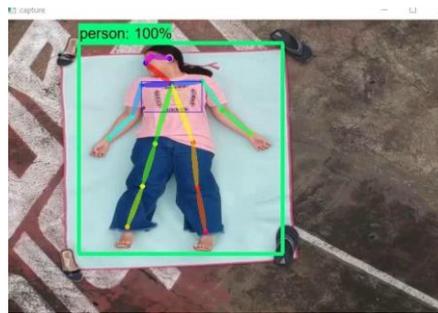
Gambar 9 Diagram alir deteksi gerakan pada korban.

Pada tahap ini proses pendeteksian dimulai dengan mengambil *output* dari hasil deteksi pada Faster R-CNN berupa video dengan format .avi yang akan diproses untuk tahap selanjutnya. Sistem ini akan mendeteksi adanya pergerakan pada dada atau punggung objek yang dibangun melalui RoI.

#### 1. Open pose

Open pose pada sistem ini dimanfaatkan untuk mendapatkan titik bahu pada objek penelitian untuk membangun RoI. Openpose yang digunakan memiliki 18 *body joint*. Titik yang digunakan untuk membangun RoI adalah keypoint 2,5,8,11. Titik pinggul dibutuhkan karena membangun RoI menggunakan jarak dari titik bahu dan pinggul dibagi 2.

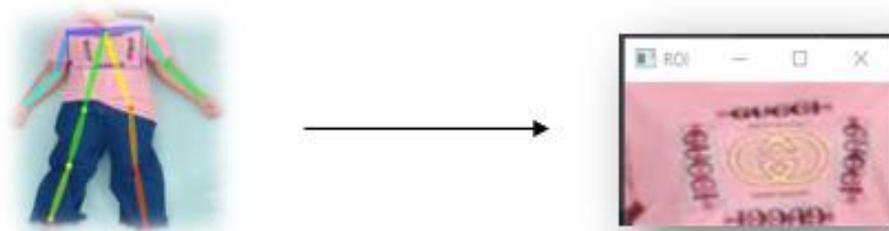
Model Open Pose yang digunakan penulis adalah model hasil pengembangan pengujian yang diberi nama model.h5 yang dilakukan oleh Michal Faber [10]. Contoh dari pendeteksian Open Pose pada sistem adalah sebagai berikut:



Gambar 10 Contoh deteksi skeleton open pose korban manusia pada sistem.

## 2. Region Of Interest (RoI)

Dalam tugas akhir ini dibutuhkan titik pada bahu dan titik paha atas untuk membangun RoI. Pada open pose nilai keypoint sudah diinisialisasikan setiap titik sesuai dengan rangka tubuh manusia. Titik yang digunakan adalah titik baru yaitu 2, 5 dan titik pinggul 8 dan 11.

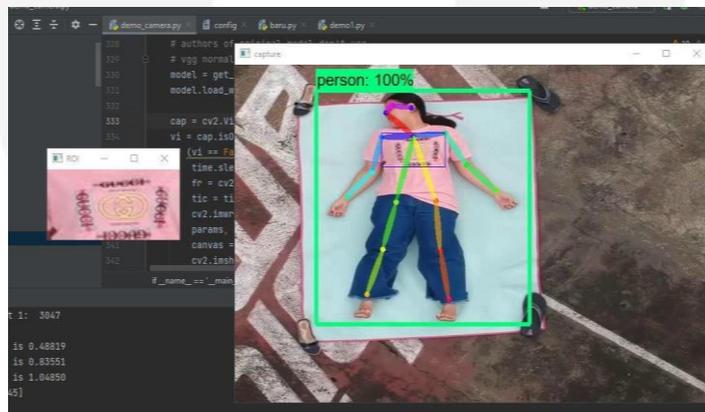


Gambar 11 Hasil rancangan daerah RoI pada objek manusia.

## 4. Hasil dan Pengujian

### 4.1 Implementasi Sistem

Sistem yang sudah dirancang diimplementasikan pada program untuk mendeteksi manusia sebagai objek korban bencana alam menggunakan model yang sudah dilatih. Berikut adalah hasil dari deteksi implementasi sistem pada gambar 12.



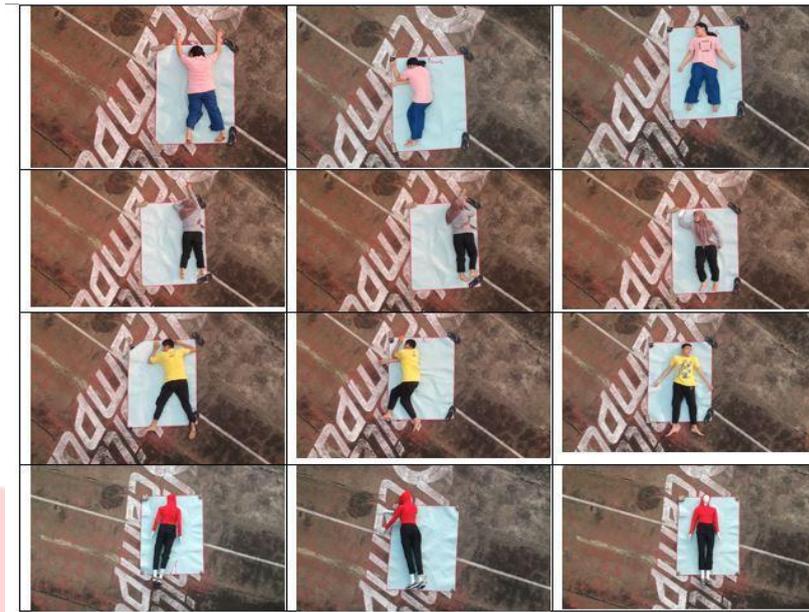
Gambar 12 Implementasi sistem gabungan deteksi tanda kehidupan pada objek manusia.

Hasil dari implementasi sistem berupa grafik yang menunjukkan adanya perubahan piksel pada manusia dan mannequin.

### 4.2 Pengujian Sistem

#### 4.2.1 Pengujian Faster R-CNN

Pengujian sistem dilakukan pada Faster R-CNN untuk mendeteksi manusia yaitu dengan menguji setiap parameter yang dapat memberi pengaruh terhadap performa sistem.



Gambar 13 Pose pada objek manusia dan manekin terdiri dari 3 pose tiap objek.

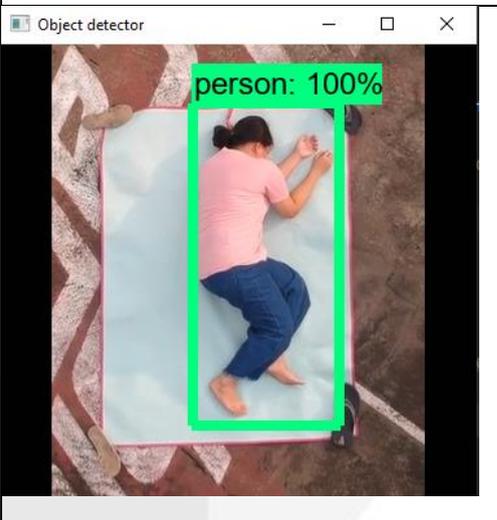
Setelah melakukan tahap pengujian pada Faster R-CNN mendapatkan hasil model training terbaik dengan parameter:

Tabel 1 Hasil proses pengujian menggunakan parameter model terbaik.

No	Pengujian	Bobot	Presisi (%)	Recall (%)	Akurasi (%)
1	Partisi Data	80:20	88	81	74
2	Learning rate	0,002	88	81	74
3	Epoch	300	88	90	80
4	Step	150000	93	99	99

Tabel 2 Hasil pengujian deteksi pada objek korban menggunakan Faster R-CNN.

No	Objek	Pose	Hasil
1	Manekin	Telentang	

2	Subjek1	Telentang	
3	Subjek2	Telungkup	
4	Subjek3	Miring	

4.2.2 Pengujian deteksi tanda kehidupan pada daerah RoI.

Tabel 3 Hasil pengujian deteksi tanda kehidupan pada objek penelitian.

No	Subjek	Pose	<i>Decease</i>	<i>Alive</i>	<i>Ket (benar/ salah).</i>
1	Mannequin	Telentang	✓		benar
		Tengkurap	✓		benar
		Miring	✓		benar
2	Subjek 1	Telentang		✓	benar

		Tengkurap		✓	benar
		Miring		✓	benar
3	Subjek 2	Telentang		✓	benar
		Tengkurap		✓	benar
		Miring		✓	benar
4	Subjek 3	Telentang		✓	benar
		Tengkurap		✓	benar
		Miring		✓	benar

Melalui hasil pengujian deteksi tanda kehidupan pada tabel 3 di atas dapat ditentukan akurasi dengan rumus:

$$akurasi = \frac{\text{jumlah berhasil}}{\text{total percobaan}} \times 100\% = \frac{12}{12} \times 100\% = 100\% \quad (1)$$

Pada persamaan 1 nilai akurasi pada deteksi tanda kehidupan pada semua objek adalah sebesar 100%.

## 5. Kesimpulan dan Saran

### 5.1 Kesimpulan

Berdasarkan implementasi sistem dan pengujian yang telah dilakukan, berikut adalah kesimpulan yang dapat ditarik penulis:

1. Pembuatan model pada Faster R-CNN dengan beragam parameter pengujian berjalan dengan baik. Model terbaik diperoleh dengan partisi dataset: 80 % 20%, *learning rate* sebesar 0,0002, epochs sebesar 300 dan step sebesar 15000 Model tersebut memberikan akurasi yang cukup baik untuk akurasi model maupun akurasi deteksi langsung.
2. Deteksi akurasi deteksi open pose pada 4 objek terdiri dari 3 manusia dan manekin dengan tiga pose yaitu telentang, tengkurap dan miring yang digunakan diperoleh akurasi sebesar 100% untuk melihat pergerakan.

### 5.2 Saran

Berdasarkan hasil perancangan dan pengujian sistem yang telah dilakukan pada tugas akhir ini, maka berikut adalah saran saran yang diusulkan untuk penelitian lebih lanjut, yaitu:

1. Menambah jumlah dataset, agar model yang dilatih lebih memiliki akurasi yang lebih tinggi.
2. Menggunakan spesifikasi PC yang lebih mumpuni untuk menjalankan Open Pose sehingga proses analisis berjalan lancar dan efektif.
3. Mengimplementasikan sistem dengan analisis secara *real-time*.

## REFERENSI

- [1] Z. Q. Zhao, P. Zheng, S. T. Xu, and X. Wu, "Object Detection with Deep Learning: A Review," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 30, no. 11, pp. 3212–3232, 2019
- [2] R. Kusumanto and A. N. Tompunu, "PENGOLAHAN CITRA DIGITAL UNTUK MENDETEKSI OBJEK," in *SEMANTIK*, Palembang, 2011.
- [3] S. Sausan, M. Syaryadhi and A. Rahman, "PERANCANGAN PROTOTIPE SISTEM PENDETEKSI POSISI KORBAN BENCANA BERBASIS MIKROKONTROLER ATMEGA328," *KITEKTRO*, vol. I, no. 8, pp. 35-42, 2016.
- [4] J. Ludwig, "Image Convolution," pp. 1–8, 2007.
- [5] A. Botalb, M. Moinuddin, U. M. Al-Saggaf, and S. S. A. Ali, "Contrasting Convolutional Neural Network (CNN) with Multi-Layer Perceptron (MLP)for Big Data Analysis," *Int. Conf. Intell. Adv. Syst. ICIAS 2018*, no. February 2019, pp. 1–5, 2018.
- [6] R. Girshick, "Fast R-CNN," *arxiv*, vol. 2, no. 27 September 2015, p. 9, 2015. Demirovic, E. Skejic, and A.

- Serifovic-Trbalic, "Performance of Some Image Processing Algorithms in Tensorflow," Int. Conf. Syst. Signals, Image Process., vol. 2018-June, pp. 2–5, 2018, doi: 11109/IWSSIP.2018.8439714.
- [7] @InProceedings {cao 2017 realtime, title = {Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields}, author = {Zhe Cao and Tomas Simon and Shih-En Wei and Yaser Sheikh}, book title = {The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)}, year = {2017}}
- [8] @inproceedings{cao2017realtime,author = {Zhe Cao and Tomas Simon and Shih-En Wei and Yaser Sheikh},book title = {CVPR},title = {Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields},year = {2017}}
- [9] J. Adiwibowo, K. Gunadi, and E. Setyati, "Deteksi Alat Pelindung Diri Menggunakan Metode YOLO dan Faster R-CNN," J. Infra, vol. 18, no. 2, pp. 106–112, 202
- [10] D. Demirovic, E. Skejic, and A. Serifovic-Trbalic, "Performance of Some Image Processing Algorithms in Tensorflow," Int. Conf. Syst. Signals, ImageProcess., vol. 2018-June, pp. 2–5, 2018, doi:11109/IWSSIP.2018.8439714.

