

Deteksi Pemakaian Helm Proyek Dengan Metode Convolutional Neural Network

Bambang Widodo, *Teknologi Informasi Institut Sains dan Teknologi Terpadu Surabaya(ISTTS)*,
Hendrawan Armanto, *Informatika Institut Sains dan Teknologi Terpadu Surabaya(ISTTS)*,
Endang Setyati, *Teknologi Informasi Institut Sains dan Teknologi Terpadu Surabaya(ISTTS)*.

Abstrak— Penggunaan helm proyek pada pekerjaan dengan resiko kecelakaan tinggi seperti pekerjaan konstruksi seringkali diabaikan oleh pekerja, sehingga apabila terjadi kecelakaan kerja yang tidak diinginkan seperti kejatuhan benda keras dari atas atau terjadi benturan keras pada kepala akan berakibat fatal bagi pekerja tersebut. Penelitian ini bertujuan untuk mendeteksi pemakaian helm proyek oleh pekerja konstruksi pada citra. Metode yang digunakan pada penelitian ini adalah *Convolutional Neural Network YOLO*. Sistem terdiri dari tiga proses utama yaitu proses *pre processing*, proses *training* dan proses deteksi. Proses *pre processing* adalah melakukan *resize* dan anotasi *labeling* pada citra dataset. Selanjutnya adalah proses *training* pada dataset dengan menggunakan *transfer learning YOLOv2*. Pada proses deteksi digunakan 4 buah *anchor box* pada setiap *grid* pada citra, hasil dari mencari nilai ukuran *bounding box* yang memiliki *IOU* terbaik dengan melakukan proses *clustering* pada dataset training dan validasi yang terdapat *bounding box* menggunakan algoritma *K-Mean clustering*. Algoritma *Intersection Over Union (IOU)* dan *Non Max Suppression (NMS)* digunakan agar *bounding box* prediksi yang dibuat presisi dengan objek yang berhasil dideteksi dan untuk menghilangkan multideteksi pada objek yang sama. Proses deteksi pada sistem ini melakukan lokalisasi dan klasifikasi dengan sekali langkah proses sehingga hasil dari proses deteksi ini adalah orang menggunakan helm proyek dan orang tidak menggunakan helm proyek. Pengujian sistem deteksi dilakukan secara individu maupun kelompok maksimal 5 orang dengan *F1-score* yang diperoleh sebesar 0,79.

Kata Kunci—*Bounding Box, CNN, Helm Proyek, YOLO*

I. PENDAHULUAN

Helm adalah alat pelindung kepala yang berfungsi melindungi kepala dari benturan sehingga kepala tidak mengalami cedera. Beberapa kegiatan manusia yang menggunakan helm untuk melindungi kepala diantaranya

Bambang Widodo, Departemen Teknologi Informasi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: bamb.inside@gmail.com)

Hendrawan Armanto, Departemen Informatika, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: hendrawan@stts.edu)

Endang Setyati, Departemen Teknologi Informasi, Institut Sains dan Teknologi Terpadu Surabaya, Surabaya, Jawa Timur, Indonesia (e-mail: endang@stts.edu)

adalah mengendarai kendaraan bermotor roda dua, melakukan pekerjaan konstruksi, beberapa kegiatan olah raga dan lain sebagainya.

Untuk pekerjaan konstruksi para pekerja diwajibkan menggunakan alat keselamatan kerja, helm proyek adalah salah satu alat keselamatan kerja yang wajib digunakan, hal tersebut sebagai salah satu tujuan penting dan harus dipatuhi oleh pekerja konstruksi dalam K3 (Keselamatan dan Kesehatan Kerja). Akan tetapi masih banyak pekerja konstruksi yang tidak menggunakan helm proyek pada saat bekerja. Mereka menggunakan helm yang sudah disediakan hanya apabila ada pengawasan dari supervisor atau penyelia, akan tetapi bila supervisor tersebut tidak berada di lokasi, sering diantara para pekerja melepas helmnya, padahal hal tersebut sangat berbahaya dan bisa berakibat fatal bila terjadi suatu insiden yang tidak diinginkan seperti kejatuhan benda keras dari atas atau terjadi benturan keras yang mengakibatkan cedera pada kepala.

Pada penelitian ini menggunakan metode *Convolutional Neural Network (CNN) YOLO*. *Convolutional Neural Network (CNN) YOLO* adalah salah satu metode *Deep learning (DL)* dan dapat digunakan untuk mendeteksi dan mengenali suatu objek pada sebuah citra sehingga diharapkan dapat mendeteksi pekerja yang menggunakan helm proyek dan yang tidak menggunakan helm proyek.

II. TINJAUAN PUSTAKA

Pada tahun 2017 Jimit Mistry, Aashish K. Misraa, Meenu Agarwal, Ayushi Vyas, Vishal M. Chudasama dan Kishor P. Upla telah melakukan penelitian untuk melakukan deteksi orang menggunakan helm dan orang tidak menggunakan helm pada pengendara sepeda motor beserta ekstraksi plat nomor menggunakan *Convolutional Neural Network (CNN)*[1]. Penelitian ini diterbitkan di Jurnal IEEE 2017. Peneliti menggunakan dua model *YOLOv2*[2], salah satunya untuk mendeteksi pengendara sepeda motor yang memakai helm dan yang tidak memakai helm. Model *YOLOv2* pertama dilatih dengan dataset *COCO*[3] yang mendeteksi jumlah kelas dalam suatu gambar. Gambar kelas orang yang telah dipotong dari *YOLOv2* pertama digunakan sebagai input untuk model *YOLOv2* kedua yang selanjutnya dilatih dengan dataset helm. Dataset helm ini terdiri dari gambar helm saja.

Untuk deteksi orang menggunakan dataset *COCO*, untuk tahap kedua deteksi helm menggunakan dataset 3054 gambar helm. Untuk tahap *testing* dataset gambar menggunakan helm dan tidak menggunakan helm diunduh dari *ImageNet*[4]. Dataset testing adalah 409 gambar memakai helm dan 403 gambar tidak memakai helm. Hasil akurasi dari penelitian ini adalah 94,7% pada 2500 iterasi.

Pada tahun 2015 Xiaogang Chen, Pengxu Wei, Wei Ke, Qixiang Ye dan Jianbin Jiao melakukan penelitian deteksi pejalan kaki dengan Deep Convolutional Neural Network (Deep-CNN)[5] yang diterbitkan di Springer International Publishing Switzerland. Dalam penelitian ini, *Aggregated Channel Features (ACF)* dan *Deep Convolutional Neural Networks (DCNN)* digunakan untuk mendeteksi pejalan kaki dengan fitur yang sangat kompleks secara efisien dan efektif. ACF digunakan untuk menghasilkan kandidat windows pejalan kaki dan DCNN digunakan untuk klasifikasi. Eksperimen yang dilakukan memiliki peforma yang baik untuk dataset INRIA[6].

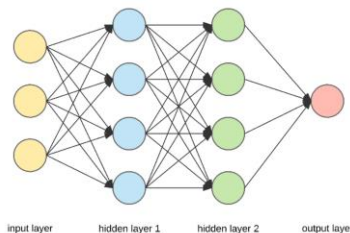
Pada tahun 2016 Sugianto dari Institut Sains dan Teknologi Terpadu Surabaya melakukan penelitian deteksi alat pelindung kepala (helm) menggunakan metode *Haar Cascade Classifier*[7]. Sistem terdiri dari dua proses utama yaitu proses *training* data dan proses deteksi. Pada proses *training* terdapat empat proses, yaitu *haar-like feature*, citra integral, *ada-boost* dan *cascade classifier*. *Haar-like feature* merupakan kumpulan fitur khusus yang mempresentasikan kepala, wajah dan helm. Citra integral adalah cara cepat menghitung *haar feature*, sedangkan *ada-boost* adalah pembobotan secara statistik nilai-nilai fitur yang didapat dan di-filter menggunakan *cascade classifier*.

Dataset yang digunakan adalah 1026 dataset positif dan 1627 dataaset negatif, dataset positif berupa citra manusia dengan menggunakan helm dan dataset negatif adalah citra yang bukan manusia menggunakan helm (citra *background* dan pemandangan). Hasil uji coba pada 20 frame video dengan 7 skenario adalah akurasi sebesar 92% untuk uji coba individu dan akurasi sebesar 71% untuk uji coba kelompok.

Selanjutnya adalah beberapa teori penunjang yang digunakan dalam penelitian ini.

A. Artificial Neural Network (ANN)

Artificial Neural Network (ANN) merupakan suatu model komputasi paralel yang meniru fungsi dari sistem jaringan syaraf biologi otak manusia. Dalam otak manusia terdiri dari milyaran neuron yang saling berhubungan. Hubungan ini disebut dengan *Synapses*. Komponen neuron terdiri dari satu inti sel yang akan melakukan pemrosesan informasi, satu akson (*axon*) dan minimal satu *dendrit*. Informasi yang masuk akan diterima oleh *dendrit*. Selain itu, *dendrit* juga menyertasi akson sebagai keluaran dari suatu pemrosesan informasi.. Struktur dasar dari Jaringan Syaraf Tiruan terdiri dari 3 jenis layer utama, yaitu input, hidden, dan output layer [8]

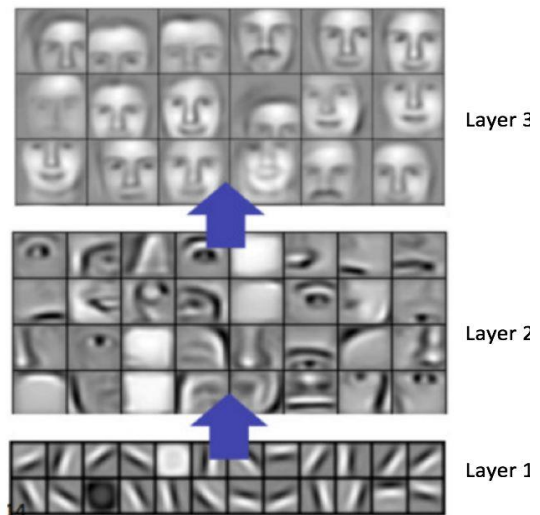


Gambar.1. Struktur Artificial Neural Network (ANN)

B. Convolutional Neural Network (CNN)

Convolutional Neural Network (CNN)[9] telah memiliki terobosan hasil yang baik selama dekade terakhir di berbagai bidang yang terkait dengan pengenalan pola; dari pemrosesan gambar hingga ke pengenalan suara. Aspek yang paling bermanfaat dari CNN adalah mengurangi

jumlah parameter dalam *Artificial Neural Network (ANN)*. Prestasi ini telah mendorong peneliti dan pengembang untuk melakukan pendekatan model yang lebih besar untuk menyelesaikan tugas-tugas kompleks, yang tidak mungkin dilakukan dengan ANN klasik. Aspek penting lain dari CNN, adalah untuk memperoleh fitur abstrak saat input merambat ke lapisan yang lebih dalam. Misalnya, dalam klasifikasi gambar, tepi citra mungkin terdeteksi di lapisan pertama, dan kemudian bentuk lebih sederhana di lapisan kedua, dan kemudian fitur dengan tingkat yang lebih tinggi seperti wajah di lapisan selanjutnya.

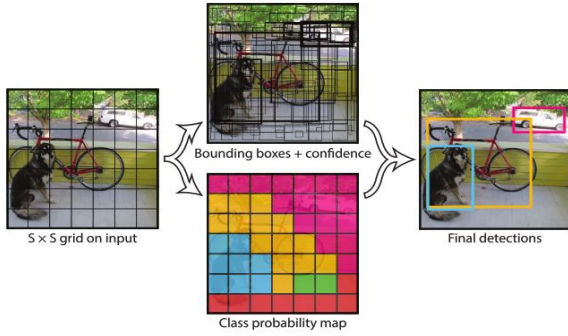


Gambar.2. Learned Fiture Dari Sebuah CNN

C. You Only Look Once (YOLO)

You Only Look Once atau YOLO[10], akan membagi inputan citra menjadi grid berukuran $S \times S$, dimana nilai S adalah 7 dengan input gambar berukuran 448×448 . Jika titik tengah koordinat pada GT (*Ground Truth*) suatu objek berada di dalam *grid*, maka *grid* tersebut bertanggung jawab untuk mendeteksi suatu objek. Setiap grid sel memprediksi B *bounding box* dan *confidence score* untuk kotak tersebut[11]. *Confidence score* ini mencerminkan seberapa yakin model tersebut bahwa di dalam *bounding box* berisi objek dan juga nilai akurasi prediksinya. Untuk mendapatkan *bounding box* prediksi, akan dilakukan konvolusi dari inputan citra, sehingga hasil akhirnya akan mendapat ukuran *bounding box* sebesar $S \times S \times (B \times 5 + C)$ dimana B adalah banyaknya *bounding box* (umumnya 2) dalam 1 *grid* dan C adalah banyaknya kelas yang dapat diklasifikasi. Nilai B dikalikan dengan 5 karena sebuah *bounding box* memiliki 5 nilai yang perlu disimpan, koordinat x , koordinat y , lebar (*width*), tinggi (*height*), dan *confidence score* (nilai probabilitas *bounding box* yang bersangkutan memiliki sebuah objek).

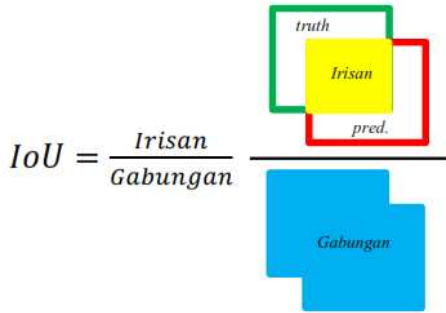
Untuk semua atribut pada *bounding box* akan dilakukan normalisasi sehingga nilainya menjadi antara 0 hingga 1. Koordinat x dan y akan dinormalisasi menyesuaikan titik kiri atas dari *grid* yang bersangkutan. Dan tinggi dan lebar akan dinormalisasi sesuai dengan ukuran gambar (lebar dan tinggi). Nilai koordinat x dan y pada sebuah *bounding box* pada setiap *grid* merupakan titik tengah *grid* yang bersangkutan. Ilustrasi YOLO dapat dilihat pada Gambar 3.



Gambar.3. Model YOLO

D. Intersection Over Union (IOU)

Intersection over Union (IOU) merupakan metode evaluasi untuk mengukur akurasi deteksi objek terhadap suatu dataset[12]. IOU membutuhkan 2 area yang akan diiris dan digabung, 2 area tersebut adalah area *ground-truth bounding box* yang merupakan *bounding box* aktual dan area yang dideteksi dari model yang dibuat[13]. Setiap *bounding box* terdiri dari 5 prediksi: x, y, w, h, dan *confidence*. Koordinat (x; y) adalah pusat *grid* sel pada *bounding box*. Lebar dan tinggi (w dan h) adalah ukuran dari citra. Prediksi *confidence* adalah IOU (*Intersection Over Union*) antara *predicted bounding box* dan *ground truth bounding box*. Setiap *grid* sel juga memprediksi probabilitas kelas bersyarat C , $\Pr(\text{Class}_i|\text{Object})$. Probabilitas ini dikondisikan pada sel *grid* yang berisi objek. Sistem hanya memprediksi satu set probabilitas kelas pada setiap *grid* sel, terlepas dari jumlah B *box*. Pada saat pengujian sistem mengalikan probabilitas kelas bersyarat dan *box confidence prediction*, yang memberi nilai *confidence* pada kelas untuk setiap *bounding box*.



Gambar.4. Intersection Over Union (IOU)

E. Non Max Suppression (NMS)

Non Max Suppression (NMS)[14] digunakan untuk menyeleksi *bounding box* yang muncul berlebihan pada objek yang sama dengan membandingkan nilai *confidence* masing-masing *bounding box*, hanya nilai *confidence* yang paling tinggi (maksimal) yang akan dipertahankan.

Algoritma *Non Max Suppression* :

1. Pilih proposal dengan skor *confidence* tertinggi, hapus dari B dan tambahkan ke dalam daftar proposal final D . (Awalnya D kosong).
2. Sekarang bandingkan proposal ini dengan semua proposal - hitung IOU (*Intersection over Union*) proposal ini dengan setiap proposal lainnya. Jika IOU lebih besar dari ambang N , hapus proposal tersebut dari B .

3. Sekali lagi ambil proposal dengan *confidence* tertinggi dari proposal yang tersisa di B dan hapus dari B dan tambahkan ke D .
4. Sekali lagi hitung IOU proposal ini dengan semua proposal di B dan hilangkan kotak yang memiliki IOU tinggi dari ambang batas.
5. Proses ini diulangi sampai tidak ada lagi proposal yang tersisa di B .

Dibawah ini adalah *pseudo-code Non Max Suppression* :

Algorithm 1 Non-Max Suppression

```

1: procedure NMS( $B, c$ )
2:    $B_{nms} \leftarrow \emptyset$ 
3:   for  $b_i \in B$  do
4:      $discard \leftarrow \text{False}$ 
5:     for  $b_j \in B$  do
6:       if  $\text{same}(b_i, b_j) > \lambda_{nms}$  then
7:         if  $\text{score}(c, b_j) > \text{score}(c, b_i)$  then
8:            $discard \leftarrow \text{True}$ 
9:       if not  $discard$  then
10:         $B_{nms} \leftarrow B_{nms} \cup b_i$ 
11:   return  $B_{nms}$ 

```

Gambar.5. Pseudo-code Non Max Suppression

F. Akurasi

Perhitungan tingkat akurasi menggunakan *precision*, *recall* dan *F1-score*[15]. *Precision* adalah tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. *Recall* adalah tingkat keberhasilan sistem dalam menemukan sebuah informasi. *F1-score* adalah ukuran akurasi sebuah uji coba.

$$Precision = \frac{\text{true positive}}{\text{true positive} + \text{false positive}} \quad (1)$$

$$Recall = \frac{\text{true positive}}{\text{true positive} + \text{false negative}} \quad (2)$$

$$F1\text{-score} = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (3)$$

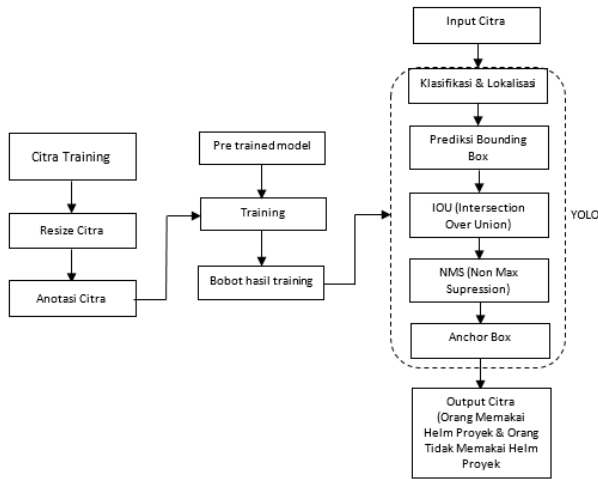
- *True Positive* adalah jumlah objek yang berhasil terdeteksi
- *False positive* adalah jumlah objek yang terdeteksi namun bukan objek yang benar
- *False negative* adalah jumlah objek yang tidak terdeteksi tetapi objek yang benar

III. PERANCANGAN

Pada Bab Perancangan ini akan dibahas mengenai desain alur kerja dari sistem dan algoritma yang akan digunakan.

A. Alur Sistem

Alur sistem yang digunakan pada penelitian ini ditunjukkan pada gambar 6. Proses awal adalah *resize* dan *labeling* citra lalu melakukan *training* dengan *transfer learning* menggunakan *pre trained model YOLOv2* menghasilkan bobot baru. Selanjutnya bobot baru digunakan untuk melakukan proses deteksi orang memakai helm proyek dan orang tidak memakai helm proyek pada citra uji coba.



Gambar.6. Alur Sistem

B. Pre Processing

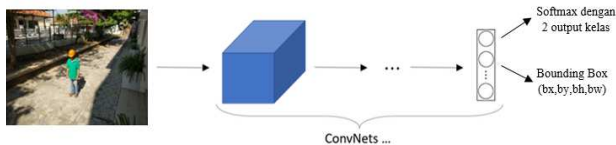
Pada tahap pre processing semua citra dataset dirubah ukurannya menjadi 448 x 448 piksel serta dilakukan proses *labeling* atau anotasi dengan menggunakan program *labelImg*.

C. Training

Dataset hasil anotasi selanjutnya digunakan untuk proses *training* dengan YOLOv2 menggunakan *transfer learning* untuk mendapatkan bobot baru, yaitu dengan menggunakan bobot *pre trained model* yang sudah dilatih untuk mengenali objek baru. *Pre trained model* yang digunakan adalah bobot YOLOv2 dijalankan pada Google Colaboratory, *epoch* dan *batch size* pada proses training 100 dan 64, sedangkan bobot baru yang dihasilkan akan digunakan untuk mendeteksi orang memakai helm proyek dan orang tidak memakai helm proyek.

D. Klasifikasi dan Lokalisasi

Klasifikasi citra adalah pengkategorian suatu citra ke dalam suatu kategori atau kelas tertentu sedangkan lokalisasi citra adalah menentukan letak atau posisi dari suatu objek didalam citra tersebut dengan *bounding box*. Deteksi objek pada citra adalah penggabungan dari klasifikasi dan lokalisasi beberapa objek dalam satu citra. Untuk klasifikasi dan lokalisasi citra hanya bisa untuk satu objek sedangkan pada deteksi objek pada citra bisa digunakan untuk lebih dari satu objek. Pada algoritma YOLO, dilakukan proses klasifikasi terlebih dahulu baru proses lokalisasi dengan menambahkan *bounding box* pada objek kelas yang terdeteksi dalam sekali proses deteksi.

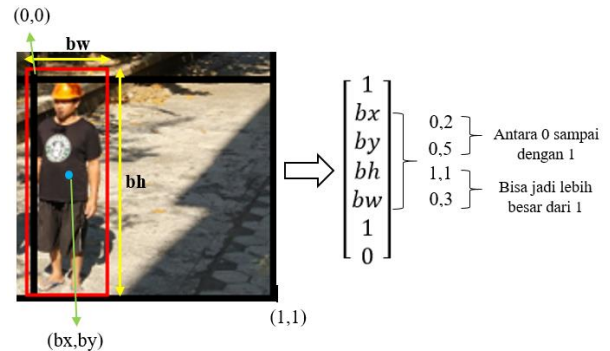


Gambar.7. Proses Deteksi Objek Dengan 2 Kelas

E. Prediksi Bounding Box

Untuk menentukan *bounding box* dari sebuah *grid* pada citra, maka didefinisikan koordinat sebuah *grid* pada citra adalah (0,0) terletak di atas kiri dan (1,1) terletak di bawah kanan. Pada Gambar 7 titik tengah *bounding box* berupa titik berwarna biru, disini mewakili *bx* dan *by*. Nilai *bx* dan *by* antara 0 sampai dengan 1 dari sebuah *grid*, untuk nilai *bx*

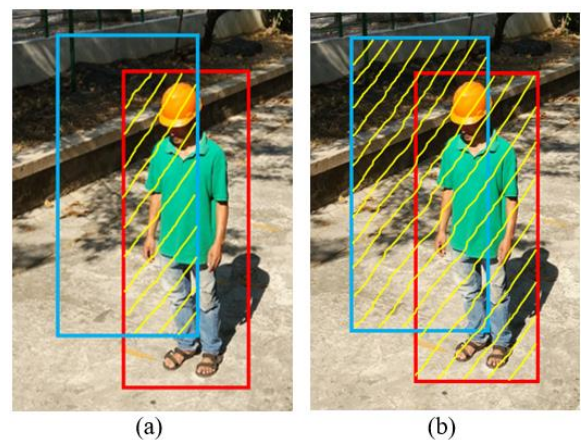
dicari dengan cara mengukur dari titik (0,0) kearah kanan sampai dengan tepi kanan *grid* pada titik (1,1). Disini nilai *bx* = 0,2. Sedangkan nilai *by* dicari dengan cara mengukur dari titik (0,0) kearah bawah sampai dengan tepi bawah *grid* pada titik (1,1). Nilai *by* = 0,5. Untuk nilai *bw* adalah lebar *bounding box* dibandingkan dengan lebar *grid*, tinggi dan lebar *grid* didefinisikan 1 jadi nilai *bw* = 0,3. Nilai *bh* adalah tinggi *bounding box* dibandingkan dengan tinggi *grid*, nilai *bh* = 1,1 karena tinggi dari *bounding box* melebihi tinggi dari *grid*. Untuk nilai *bx* dan *by* adalah antara 0 sampai dengan 1, sedangkan untuk nilai *bh* dan *bw* bisa jadi lebih besar dari 1 atau melebihi ukuran tinggi dan lebar dari *grid*.



Gambar.8. Menentukan *Bounding Box* Pada Citra

F. Intersection Over Union (IOU)

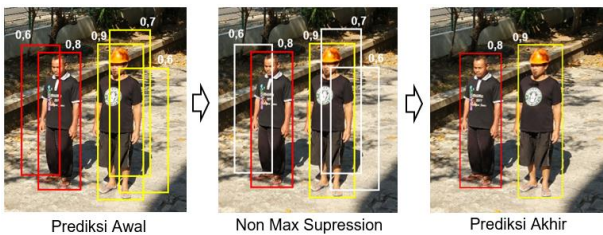
IOU dihitung dari luas area irisan dibagi luas gabungan dari *bounding box ground truth* (warna merah) dan *bounding box* prediksi (warna biru). Bila perbandingan rasio atau IOU antara *bounding box* prediksi dan *bounding box ground truth* adalah 1, maka dikatakan IOU sangat sempurna karena tujuan dari *bounding box* prediksi adalah sama atau mendekati *bounding box ground truth*. Untuk nilai hasil IOU sama atau lebih besar dari 0,5 bisa dikatakan *bounding box* prediksi sudah cukup tepat dalam membuat *bounding box* pada objek. Penentuan nilai *threshold* akan berpengaruh pada hasil lokalisasi objek pada citra. Bila *threshold* terlalu tinggi maka bisa berakibat jumlah objek yang terdeteksi lebih sedikit dari jumlah objek yang sebenarnya, sedangkan bila *threshold* terlalu rendah bisa berakibat satu objek dideteksi oleh beberapa *bounding box*.



Gambar.9. Area Irisan *Bounding Box Ground Truth* dan *Bounding Box* Prediksi (a), Area Gabungan *Bounding Box Ground Truth* dan *Bounding Box* Prediksi (b)

G. Non Max Suppression (NMS)

Salah satu kendala dalam deteksi objek adalah ada kemungkinan *multiple* deteksi pada satu objek. Satu objek tidak dideteksi sebagai satu objek tapi dideteksi berkali-kali dan dikenali sebagai lebih dari satu objek sehingga akurasi dari deteksi objek akan rendah. Untuk mengatasi permasalahan tersebut maka digunakan *Non Max Suppression* (NMS). *Non Max Suppression* (NMS) bekerja dengan cara membersihkan *bounding box* prediksi yang mempunyai nilai probabilitas terendah dan mengambil satu *bounding box* prediksi yang mempunyai nilai probabilitas tertinggi. Bila *threshold Non Max Suppression* ditentukan 0,6 maka setelah melalui proses *Non Max Suppression* akan diambil *bounding box* dengan nilai probabilitas tertinggi yaitu 0,8 untuk objek orang tidak memakai helm proyek (*bounding box* warna merah) dan 0,9 untuk objek orang memakai helm proyek (*bounding box* warna kuning).



Gambar.10. Proses *Non Max Suppression* Pada *Bounding Box*

H. Anchor Box

Sebuah *grid* pada citra, hanya dapat mendeteksi satu objek. Bila dalam satu *grid* terdapat dua buah objek atau lebih, maka dapat digunakan *anchor box*. Pada penelitian ini menggunakan 4 buah *anchor box*, hasil dari mencari nilai ukuran *bounding box* yang memiliki IOU terbaik dengan melakukan proses *clustering* pada dataset training dan validasi yang terdapat *bounding box* menggunakan algoritma *K-Mean clustering*[16].

TABEL I
UKURAN 4 BUAH ANCHOR BOX

No	Lebar	Tinggi	Ukuran
1	0,06505013	0,24317773	Sangat Kecil
2	0,10062081	0,45632851	Kecil
3	0,29744874	0,75714696	Besar
4	0,16295355	0,6445702	Sedang

IV. PENGUJIAN DAN ANALISIS

A. Dataset

Dataset yang digunakan dalam penelitian ini adalah citra orang memakai helm proyek (diambil dari dataset VOC2028) sebanyak 1500 citra yang dibagi menjadi 1200 citra untuk data pelatihan dan 300 citra untuk data validasi serta dataset orang tidak memakai helm proyek (diambil dari dataset INRIA) sebanyak 1500 citra yang dibagi menjadi 1200 citra untuk data pelatihan dan 300 citra untuk data validasi. Posisi tubuh pada dataset orang memakai helm proyek dan dataset orang tidak memakai helm proyek bervariasi.

TABEL II
DATASET

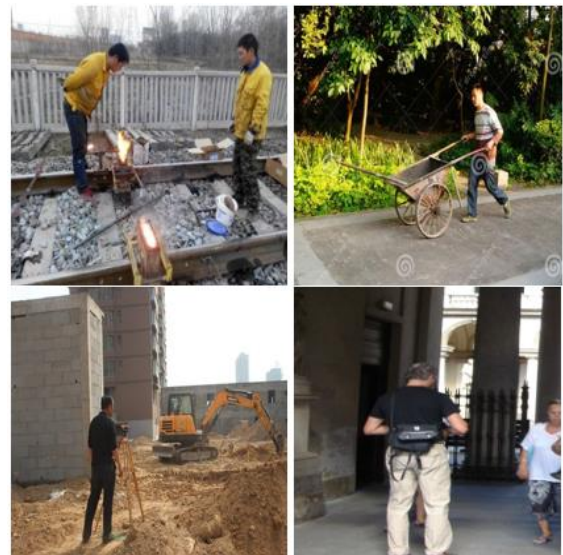
	Orang Memakai Helm Proyek	Orang Tidak memakai Helm Proyek	Total
Dataset Training	1200	1200	2400
Dataset Validasi	300	300	600
			3000

- Citra Orang Memakai Helm Proyek



Gambar.11. Citra Orang Memakai Helm Proyek

- Citra Orang Tidak Memakai Helm Proyek



Gambar.12. Citra Orang Tidak Memakai Helm Proyek

B. Pengujian

Uji coba dilakukan pada data uji sejumlah 90 citra yang dibagi menjadi 9 skenario yaitu 1 orang memakai helm proyek, 1 orang tidak memakai helm proyek, 1 orang memakai helm proyek 1 orang memakai topi baseball, 1 orang memakai helm proyek 1 orang tidak memakai helm proyek, 1 orang memakai helm proyek 1 orang memakai helm halfface, 2 orang memakai helm proyek 1 orang memakai topi baseball 1 orang memakai helm halfface 1 orang tidak memakai helm proyek, 2 orang memakai helm proyek ditentang, 3 orang memakai helm proyek dan 3 orang memakai helm proyek 2 orang tidak memakai helm proyek.

Jarak objek dengan kamera adalah 1 – 10 meter. Citra diambil dengan kamera yang diletakkan didepan pekerja yang berjalan kedepan, posisi kamera dipojok dengan ketinggian 2,5 meter diatas tanah. *Threshold* objek adalah 0,3 dan *threshold IOU* adalah 0,3.

Pada Gambar 13 untuk uji coba dengan kondisi 1 orang memakai helm proyek pada jarak objek dengan kamera 4

meter, sistem berhasil mendeteksi dengan benar, nilai prediksi yang dihasilkan adalah 0,789. Sedangkan pada Gambar 14 terdapat 2 objek dengan 2 kelas yang berbeda, yaitu 1 orang memakai helm proyek dan 1 orang tidak memakai helm proyek pada jarak objek dengan kamera 1 meter. Pada kondisi ini objek lebih sulit untuk dideteksi dibandingkan dengan kondisi pada Gambar 13 yang hanya memiliki 1 objek pada sebuah citra. Pada hasil uji coba ini sistem berhasil mendeteksi dengan benar, nilai prediksi yang dihasilkan adalah 0,741 (orang memakai helm proyek) dan 0,376 (orang tidak memakai helm proyek).



Gambar.13. Hasil Uji Coba 1 Orang Memakai Helm Proyek



Gambar.14. Hasil Uji Coba 1 Orang Memakai Helm Proyek 1 Orang

Untuk keseluruhan hasil uji coba ditunjukkan pada Tabel III. Dalam 9 skenario yang telah dijelaskan sebelumnya, pekerja dikondisikan dengan berbagai macam situasi, seperti jumlah pekerja paling sedikit adalah 1 orang dan paling banyak adalah 5 orang. Selain itu penutup kepala tidak hanya helm proyek, tapi digunakan juga topi baseball dan helm *half*face, hal ini bertujuan untuk menguji sistem dalam mendeteksi penutup kepala selain helm proyek.

TABEL III
DATA HASIL PENGUJIAN

Jarak Objek Dengan Kamera (m)	Kondisi	Precision	Recall	F1-Score
1-10	1 orang memakai helm proyek	1	1	1
1-10	1 orang tidak memakai helm proyek	0,90	0,90	0,90
1-10	1 orang memakai helm proyek 1 orang memakai topi baseball	0,50	0,90	0,63
1-10	1 orang memakai helm proyek 1 tidak memakai helm proyek	0,65	0,90	0,70
1-10	1 orang memakai helm proyek 1 tidak memakai helm halfface	0,90	0,87	0,88
1-10	2 orang memakai helm proyek 1 memakai topi baseball 1 memakai helm halfface 1 tidak memakai helm proyek	0,82	0,57	0,65
1-10	2 orang helm ditengah	0,80	0,80	0,80
1-10	3 orang pakai helm proyek	1	0,87	0,92
1-10	3 orang memakai helm proyek 2 tidak memakai helm proyek	0,80	0,55	0,63
	Rata-rata	0,82	0,82	0,79

V. KESIMPULAN

Dari hasil pengujian pada Tabel III terhadap 90 citra yang dibagi menjadi 9 skenario yang dilakukan diperoleh *F1-score* sebesar 0,79. Hasil *F1-score* untuk jumlah pekerja yang berkelompok (lebih dari 1 orang pekerja) lebih rendah dibandingkan dengan jumlah pekerja yang individu karena seringkali tubuh pekerja berdempetan, tertutup satu dengan yang lain, sehingga sistem kesulitan untuk mendeteksi dengan benar. Dengan hasil ini sistem dapat membantu pengawas untuk memastikan para pekerja menggunakan helm proyek di lokasi pekerjaan dengan hasil yang optimal pada pekerja individu (tidak berkelompok).

DAFTAR PUSTAKA

- [1] J. Mistry, A. K. Misraa, M. Agarwal, A. Vyas, V. M. Chudasama, and K. P. Upla, "An automatic detection of helmeted and non-helmeted motorcyclist with license plate extraction using convolutional neural network," in *2017 seventh international conference on image processing theory, tools and applications (IPTA)*, 2017, pp. 1–6.
- [2] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 7263–7271.
- [3] T.-Y. Lin *et al.*, "Microsoft coco: Common objects in context," in

- European conference on computer vision*, 2014, pp. 740–755.
- [4] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” *Commun. ACM*, vol. 60, no. 6, pp. 84–90, 2017.
- [5] X. Chen, P. Wei, W. Ke, Q. Ye, and J. Jiao, “Pedestrian detection with deep convolutional neural network,” in *Asian Conference on Computer Vision*, 2014, pp. 354–365.
- [6] N. Dalal and B. Triggs, “Histograms of oriented gradients for human detection,” in *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR ’05)*, 2005, vol. 1, pp. 886–893.
- [7] S. Sugianto, E. Setyati, and H. Armanto, “Deteksi Alat Pelindung Kepala (Helm) Menggunakan Metode Haar Cascade Classifier,” *Joutica*, vol. 4, no. 1, pp. 232–236, 2019.
- [8] A. Hermawan, “Jaringan Saraf Tiruan: Teori dan Aplikasi,” 2006.
- [9] S. Albawi, T. A. Mohammed, and S. Al-Zawi, “Understanding of a convolutional neural network,” in *2017 International Conference on Engineering and Technology (ICET)*, 2017, pp. 1–6.
- [10] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 779–788.
- [11] M. Menegaz, “Understanding YOLO,” 2017. <https://hackernoon.com/understanding-yolo-f5a74bbc7967>.
- [12] A. Rosebrock, “Intersection over Union (IoU) for object detection,” *l[\setminus]nea*. [Consultado 20 enero 2020]. Dispon. en <https://www.pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>, 2016.
- [13] A. Kamal, “Yolo, yolov2 and yolov3: All you want to know,” *Mediu. Seach date*, 2019.
- [14] K. Sambasivarao, “Non-maximum suppression (nms),” *Online unter https://towardsdatascience.com/non-maximum-suppression-nms-93ce178e177c [Stand 16.10. 2019]*, 2019.
- [15] M. Septian and K. Kunci, “DETEKSI WAJAH MENGGUNAKAN METODE VIOLA JONES PADA GRAPHICS PROCESSING UNIT FACE DETECTION USING VIOLA JONES METHOD ON GRAPHICS,” *Ilmu Komputer, Tek. Inf.*, 2014.
- [16] V. Yadav, “Generating Anchor boxes for Yolo-like network for vehicle detection using KITTI dataset,” 2017. <https://vivek-yadav.medium.com/part-1-generating-anchor-boxes-for-yolo-like-network-for-vehicle-detection-using-kitti-dataset-b2fe033e5807>.