

Sistem Pendeteksi Sapi pada Peternakan dengan Citra Video UAV Menggunakan Metode Regions Based Convolution Neural Network

Muhamad Rizky Arief Wicaksono
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
ariefrizky@student.telkomuniversity.ac.id

Casi Setianingsih
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
setiacasi@telkomuniversity.ac.id

Randy Erfa Saputra
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
resaputra@telkomuniversity.ac.id

Abstrak - Kemajuan teknologi industri yang mengalami perkembangan dan kemajuan yang pesat seiring dengan semakin majunya teknologi modern. Salah satu contoh yang sangat berdampak pada industri adalah industri peternakan yang dapat diterapkan sistem identifikasi dan lokalisasi menggunakan deteksi objek untuk meningkatkan efisiensi dan produktivitas. *R-CNN* adalah sebuah teknik yang terinspirasi dari cara manusia menghasilkan persepsi visual atau yang bisa dikatakan kemampuan melihat manusia dari mata diolah oleh otak manusia untuk menjadikannya sebuah visual. Hasil yang didapatkan setelah melakukan penelitian di lokasi peternakan dan memprosesnya menjadi sebuah citra berupa video, dan kemudian diolah kembali dengan mendapatkan hasil video yang sudah berhasil mendeteksi keberadaan hewan ternak dengan baik dengan sistem deteksi objek menggunakan algoritma *Regions based Convolutional Neural Network*. Didapatkan tingkat akurasi yang berada di angka yang paling optimal pada ketinggian 10 meter dalam keadaan diam dengan rata – rata akurasi berada pada angka 55.5% untuk kondisi drone diam, sedangkan untuk kondisi drone bergerak didapatkan pada 5 meter diatas objek dengan kecepatan 0.1 m/s dengan rata – rata akurasi yang berada di angka 50%.

Kata kunci - UAV, Deteksi, Drone, R-CNN.

I. PENDAHULUAN

Deteksi objek merupakan suatu proses identifikasi objek tertentu kedalam suatu citra atau video. Hal ini sering kali digunakan dalam berbagai macam bidang seperti keamanan, transportasi, dan pengawasan. Dengan berkembangnya sektor industri peternakan dari yang konvensional ke modern, deteksi objek akan sangat membantu dan diterapkan untuk membantu dan dapat diterapkan untuk membantu dalam melacak dan mengawasi ternak [1].

Dengan teknologi objek, peternakan dapat melacak posisi serta perilaku hewan ternaknya, dan memantau tingkah laku mereka sekaligus memonitoring kondisi kesehatan mereka dengan lebih mudah. Selain itu deteksi objek juga dapat membantu peternak dalam mengelola produksi, termasuk pemantauan ternak, dengan keputusan

tepat dan efisien dalam memaksimalkan dan meningkatkan kualitas prosuk ternak.

Di Indonesia sendiri belum banyak peternakan modern, dan masih asing dengan teknologi deteksi objek. Dengan permasalahan diatas maka diperlukannya suatu sistem yang dapat mengetahui keberadaan hewan yang berada di peternakan dengan menggunakan deteksi objek yang menggunakan algoritma R-CNN atau Regions based Convolution Neural Network yang menggabungkan teknik deep learning dengan pemrosesan citra. Dengan adanya metode ini dapat mengatasi masalah dalam deteksi objek konvensional yang sering kali tidak efisien dan tidak akurat.

Sebelum melakukan kegiatan dilakukan pengumpulan data teori dan beberapa literatur review berupa paper dan prosiding yang berkaitan dengan algoritma R-CNN (*Regions based Convolution Neural Network*) untuk mempelajari cara kerja dan juga arsitektur dari R-CNN. Setelah itu dilakukan pengambilan data dilokasi penelitian yaitu peternakan sapi.

II. KAJIAN TEORI

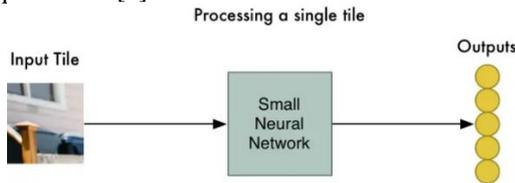
A. Computer Vision

Merupakan bidang ilmu yang mempelajari, memahami, merekonstruksi, dan menginterpretasikan gambar pada suatu komputer agar dapat mengenali suatu objek dan mampu menampilkan objek beserta dengan informasi dalam bentuk digital. Penggunaannya didesain untuk membuat komputer dapat melihat dan mengenali suatu objek yang terdeteksi [2].

Saat ini implementasi dari *Computer Vision* dalam hal mendeteksi sebuah objek di kehidupan sehari-hari sangatlah luas. *Self-driving car*, *face recognition*, dan juga *moving sensor* merupakan pengimplementasian *computer vision* yang ada di kehidupan sehari-hari. Objek *Computer Vision* biasanya mengacu pada suatu elemen tunggal, hal ini membuat pendeteksian suatu objek sulit membedakan kelas dan juga menlokalisasi dan mendeteksi objek dengan cara merepresentasikan dengan bantuan kotak-kotak pembatas [3].

B. Region Based C-NN

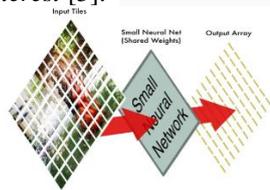
Dalam penelitian ini R-CNN digunakan sebagai Algoritma untuk mendeteksi objek yaitu sapi. R-CNN yang merupakan turunan dari *Convolution Neural Network* yang dimana bisa digunakan pada data image. *CNN* bisa digunakan untuk mendeteksi dan mengenali suatu objek yang ada pada sebuah image atau gambar. *CNN* adalah sebuah teknik yang terinspirasi dari cara manusia menghasilkan persepsi visual atau yang bisa dikatakan kemampuan melihat manusia dari mata diolah oleh otak manusia untuk menjadikannya sebuah visual atau gambar yang dapat dilihat [4].



GAMBAR 1
Frame-by-Frame berdasarkan artikel oleh Qolbiyatul Lina

Tiap gambar yang akan di proses *frame-by-frame* menjadi kecil-kecil, dan akan menghasilkan gambar yang telah dipotong dengan konvolusi yang sama setiap gambar hasil konvolusi tersebut kemudian dijadikanlah sebuah input untuk menghasilkan sebuah representasi fitur. Pada hal ini memberikan *CNN* kemampuan mengenali sebuah objek, dimanapun posisi objek tersebut muncul [5][1].

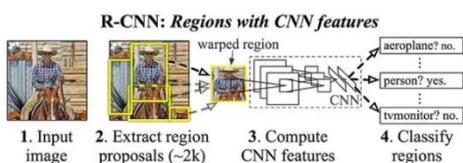
Proses ini dilakukan pada semua bagian dari masing-masing gambar kecilnya, dengan menggunakan filter yang sama. Dengan kata lain, tiap-tiap bagian gambar akan memiliki *weight sharing* yang sama, dan jika ada sesuatu yang terlihat menarik maka akan ditandai dengan bagian itu sebagai *object of interest* [5].



GAMBAR 2
Weights Sharing Biases berdasarkan artikel oleh Qolbiyatul Lina

R-CNN muncul karena banyaknya potongan gambar yang harus diproses oleh *CNN*. *Regions based CNN* terdiri dari 3 tahap:

1. Mencari region atau bagian gambar yang mungkin sebuah objek.
2. Tiap region kemudian dijadikan input untuk *CNN* sebagai ekstraktor dari tiap region.
3. Setiap fitur yang dihasilkan, kemudian menjadi input untuk *SVM*.



GAMBAR 3
Arsitektur R-CNN berdasarkan artikel oleh Richard Dharmadi

C. Library Tensorflow

Tensorflow dalam penelitian ini digunakan melatih dan menjalankan *Neural Network* yang bertujuan untuk mengklasifikasikan suatu objek yang aka di deteksi. Tensorflow merupakan *Library* yang dikembangkan oleh Google dan merupakan salah satu library yang paling populer dan banyak digunakan untuk mengembangkan serta menerapkan *Machine Learning* dan algoritma lain yang memiliki banyak operasi matematika yang perlu dilakukan [6].

Machine Learning memiliki manfaat besar bagi banyak perusahaan, sehingga banyak perusahaan dari berbagai bidang industri membutuhkan alat serta sumber daya yang memadai untuk menerapkan ML dan memecahkan masalah mereka. Dari mulai bidang kesehatan sampai ke *social network* dan *e-commerce*. *Tenserflow* digunakan oleh banyak pihak untuk berbagai hal [6][7].

D. Pyhton

Python adalah bahasa pemrograman yang dipakai untuk membuat program dan untuk melakukan suatu analisis data, dalam penelitian ini Pyhton sebagai bahasa pemrograman digunakan untuk membuat program dan menganalisis data yang akan dipakai untuk mendeteksi objek.

Selain digunakan untuk membuat aplikasi, perintah, dan juga menganalisis data. *Pyhton* bisa digunakan untuk membuat program apa saja dan menyelesaikan berbagai permasalahan, selain itu *Pyhton* juga dinilai mudah untuk dipelajari namun pyhton juga termasuk bahasa pemrograman tingkat tinggi.

Pyhton bersifat *open-source*, yang dimana ada banyak sekali fitue dan kode buatan creator yang semakin memperluas kapabilitas *Pyhton* sendiri [8][9].

E. OpenCV

OpenCV adalah sebuah library pada bahasa pemrograman yang ditujukan untuk pengolahan gambar secara *real-time*, *library* ini digunakan sebagai besar untuk pengolahan citra atau gambar secara *real-time* yang secara teori *OpenCV* digunakan untuk meniru cara kerja sistem visual manusia yaitu melihat suatu objek melalui indera penglihatan yang akan diteruskan ke otak untuk diproses. Dalam penelitian ini *OpenCV* digunakan sebagai *Library* untuk mengolah gambar dan data yang sudah direkam sebelumnya [10].

Penggunaan *OpenCV* juga banyak di industri, khususnya bagian *inspection* atau pengecekan kualitas produk, mulai dari makanan, minuman, *sparepart* kendaraan atau mesin, dan masih banyak lagi, banyak aspek yang dapat diperiksa oleh *OpenCV*, misalnya pengecekan warna, label dan lainnya yang ada pada *part-part* mesin atau *sparepart* [11].

F. Numpy

Numpy adalah library *Pyhton* yang berfokus pada *Scientific Computing*. *Numpy* menyediakan fungsi-fungsi yang siap dipakai untuk memudahkan perhitungan-perhitungan *scientific* seperti, matriks, aljabar, statistik, dan lainnya. *Numpy* juga berfungsi untuk mengolah data berbentuk *array* dengan metode komputasi ilmiah [12][13].

G. Scipy

Scipy adalah suatu modul untuk melakukan beberapa hitungan *scientific* dengan bahasa pemrograman *Pyhton*. *Scipy* dapat menyelesaikan persamaan differensial, integral

numerik, pemrosesan sinyal, dan lainnya, untuk menggunakan *Scipy* membutuhkan *Numpy* yang merupakan array yang berperan sebagai matriks[14].

H. Confusion Matrix

Confusion Matrix adalah salah satu tools analitik prediktif yang menampilkan serta membandingkan nilai yang sebenarnya dengan hasil yang telah di prediksi oleh model yang dapat digunakan untuk menghasilkan evaluasi seperti akurasi, presisi, *recall* [15].

Terdapat empat nilai yang dihasilkan didalam tabel confusion matrix yaitu, True Positive (TP), False Positive (FP), False Negative (FN), True Negative (TN). Dibawah ini merupakan tabel confusion matrix [16].

		Nilai Aktual	
		Positive	Negative
Nilai Prediksi	Positive	TP	FP
	Negative	FN	TN

GAMBAR 4

Ilustrasi Confusion Matrix berdasarkan artikel oleh Lutfia Afifah

True Positive (TP) adalah jumlah data bernilai negatif dan diprediksi benar positif, dan *False Positive* (FP) adalah jumlah data bernilai negatif dan diprediksi sebagai positif. Sedangkan *False Negative* (FN) jumlah data bernilai negatif dan diprediksi sebagai negative dan *True Negative* (TN) jumlah data bernilai negatif dan diprediksi benar sebagai negatif [15][16].

Untuk mengetahui performa model yang telah dibuat, maka ada beberapa rumus yang digunakan untuk menghitung Presisi, Recall, dan Akurasi, dibawah ini merupakan rumus – rumus yang digunakan juga dalam penelitian ini.

1. Akurasi

Untuk nilai akurasi dari data bernilai positif yang diprediksi positif dan data bernilai negatif yang diprediksi negatif dibagi dengan jumlah keseluruhan data yang ada didalam dataset.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \quad (2.1)$$

2. Presisi

Untuk mengetahui tingkat keakuratan data yang asli dengan data hasil prediksi. Presisi merupakan rasio prediksi yang benar positif dibandingkan dengan keseluruhan data hasil prediksi.

$$Precision = \frac{TP}{TP + FP} \quad (2.2)$$

3. Recall

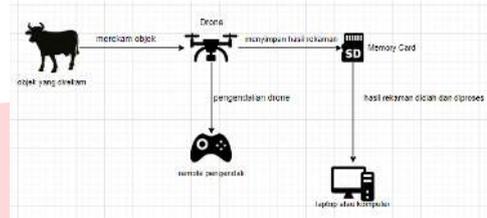
Untuk mengetahui keberhasilan model dalam menemukan kembali sebuah informasi. Recall merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif.

$$Recall = \frac{TP}{TP + FN} \quad (2.3)$$

III. METODE

A. Desain Sistem

Dalam penelitian Tugas Akhir ini, dibuat sebuah sistem deteksi objek hewan sapi. Sapi berperan sebagai objek yang dideteksi sistem ini menggunakan algoritma R-CNN (Region based Convolution Neural Network), yang dimana sistem ini akan mendeteksi keberadaan hewan sapi dengan kamera drone yang akan merekam dan akan disimpan didalam kartu memori yang kemudian kartu memori tersebut akan diambil dan diproses di perangkat komputer atau laptop untuk dilakukan proses pendeteksi hewan menggunakan algoritma R-CNN (regions based convolution neural network).

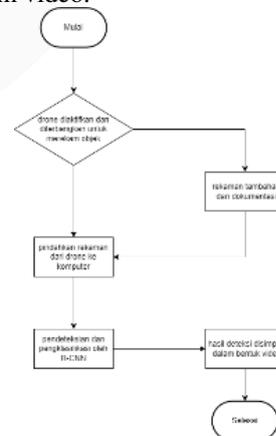


GAMBAR 5
Desain Sistem

Perancangan sistem menggunakan arsitektur algoritma R-CNN yang sudah ada namun belum dilakukan training, pada model R-CNN diberikan Custom Training yang berdasarkan dataset yang diperlukan yaitu hewan sapi. Dan untuk memperoleh dataset yang diperlukan ratusan hingga ribuan foto atau gambar sapi didalamnya, gambar-gambar yang sudah dikumpulkan akan diberikan label yang mana label tersebut digunakan untuk menandakan objek yang dideteksi.

B. Diagram Alir Pengambilan Data

Penelitian dimulai dari pengambilan data guna mendapatkan rekaman objek sapi di peternakan, drone di terbangkan dan dikendalikan oleh operator sampai mendapatkan hasil rekaman. Setelah rekaman pertama didapatkan, maka drone diterbangkan kembali guna mendapatkan rekaman berikutnya dengan ketentuan yang sudah ditetapkan, jika sudah selesai merekam data video yang ada di drone dipindahkan ke perangkat komputer atau laptop yang sudah ada algoritma R-CNN dengan dataset yang sudah di tarin untuk dilakukan deteksi sehingga mendapatkan outline bounding box muncul mendeteksi sapi yang ada didalam video.

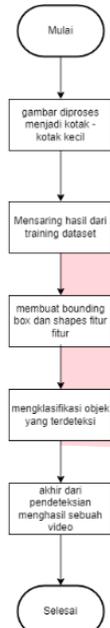


GAMBAR 6

Flowchart Pengambilan data

C. Diagram Alir Pendeteksi Objek

Untuk pendeteksian objek dimulai dengan penginputan data kedalam algoritma R-CNN yang akan di filter frame-by-frame menjadi kotak-kotak kecil dan kemudian akan melakukan penentuan bounding box pada objek. Kotak-kotak kecil tersebut akan menentukan kelas untuk objek didalam bounding box dan akan menampilkan hasil objek yang telah terdeteksi.



GAMBAR 7
Flowchart Pendeteksi objek

D. Diagram Alir Perancangan Sistem

Membuat sistem Deteksi Objek sapi menggunakan algoritma R-CNN (*Regions based Convolution Neural Network*) dengan menginput hasil rekaman yang sebelumnya sudah diambil menggunakan kamera Drone, kemudian video rekaman akan diproses didalam program yang sudah memuat algoritma yang digunakan.



GAMBAR 8
Flowchart Perancangan

E. Metode Training R-CNN

Training untuk *Regions based Convolution Neural Network* dilakukan di platform *Jupyter Notebook*, menggunakan dataset sintesis terdiri dari kotak, segitiga, dan lingkaran. Yang memungkinkan untuk men-training dataset dengan cepat, namun memerlukan GPU dan *Network Backbone* menggunakan bantuan dari *Resnet101* yang berfungsi untuk menghaluskan dan merapihkan dari data yang sudah di training. Yang dimana hasil filtering pertama berupa gambar seperti diarsir dan disinilah *Resnet101* digunakan untuk memperhalus gambar.

IV. HASIL DAN PEMBAHASAN

Dalam Tugas Akhir ini ada beberapa parameter pengujian yaitu keakurasian deteksi, jarak deteksi, dan kecepatan sistem bergerak.

A. Pengujian Klasifikasi

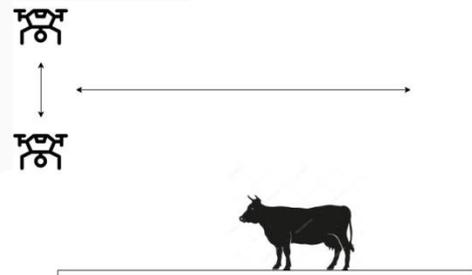
Pengujian klasifikasi dilakukan menggunakan data beberapa foto hewan selain sapi yang diambil di Kebun Binatang Bandung yang dimana akurasi model akan diuji dengan membedakan sapi dengan hewan lainnya, hasil yang didapat adalah model akan dapat membedakan dengan beruang, unta, singa, rusa, tapir, harimau, buaya, dan gajah, dan tidak akan muncul bounding box kepada hewan-hewan lain selain sapi. Sementara bounding box akan muncul kepada hewan yang memiliki fitur seperti sapi layaknya benteng atau kerbau.

B. Pengujian Lapangan

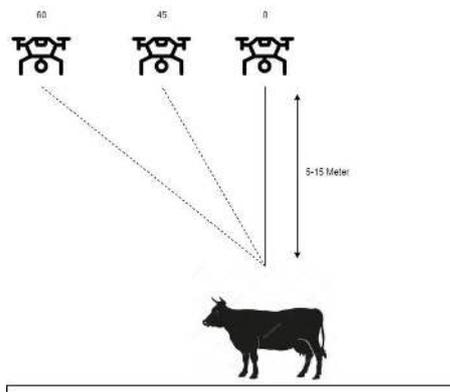
Percobaan dan pengambilan data dilaksanakan di peternakan sapi yang berada di Bogor dan video diambil menggunakan dua buah Drone, drone 1 dan drone 2. Pengambilan data dilakukan sebanyak 2 kali pada ketinggian 5, 10, dan 15 meter dari permukaan tanah. Dan video diambil dari sudut 0, 45, dan 60 derajat dari objek. Video diambil menggunakan drone saat diam dan drone dalam keadaan bergerak dengan kecepatan 0.1 m/s dengan intensitas cahaya yang dicatat selama pengujian.

C. Skenario Pengujian

Pada tahap penelitian ini, *drone* yang dikendalikan oleh seorang operator/pilot akan merekam objek yang ada berada dibawah, dalam hal ini objek tersebut adalah sapi.



GAMBAR 9
Skenario Pengujian Drone dalam Kondisi Diam
Pada gambar diatas kamera drone diam ketinggian yang paling rendah dan ketinggian paling tinggi secara bertahap.



GAMBAR 10

Skenario Pengujian Drone dengan angle

Sedangkan pada gambar diatas ini drone akan merekam objek yang berada dibawahnya sesuai dengan angle yang sudah ditentukan sebelumnya, dan sama hal pada pengujian sebelumnya drone akan bergerak dari ketinggian yang paling rendah dan ketinggian paling tinggi secara bertahap dengan ketinggian yang sudah ditetapkan sebelumnya.

D. Pengujian dan Hasil

Dalam pengujian 1 drone bergerak maju dengan variasi ketinggian 5 meter, 10 meter dan 15 meter, dikarenakan kondisi baterai drone yang terbatas maka untuk mempersingkat waktu pengambilan rekaman drone diterbangkan kurang lebih 5 menit di udara untuk merekam semua pengujian ketinggian. Dari pengujian dan pengumpulan data maka didapatkan data pada tabel dibawah ini.

TABEL 1
Tabel Percobaan 1 Akurasi Kondisi Drone Diam

Percobaan 1 Kondisi Drone Diam								
Ketinggian 5 Meter			Ketinggian 10 Meter			Ketinggian 15 Meter		
Su dut	Lux	Akur asi	Su dut	Lux	Akur asi	Su dut	Lux	Akur asi
0°	98556	25%	0°	98556	66.6%	0°	98556	25%
45°	77470	50%	45°	77470	50%	45°	77470	0%
60°	12892	40%	60°	12892	33.3%	60°	12892	0%

Pada tabel diatas menunjukkan hasil perhitungan akurasi yang didapat dari hasil percobaan 1 dengan ketinggian 5 meter, 10 meter, dan 15 meter pada sudut 0° dengan intensitas cahaya 98556 lx dan, pada sudut 45° dengan intensitas cahaya 77470 lx, dan pada sudut 60° intensitas cahaya 12892 lx. Maka didapatkan hasil seperti tabel diatas.

TABEL 2
Tabel Percobaan 2 Akurasi Kondisi Drone Diam

Percobaan 2 Kondisi Drone Diam								
Ketinggian 5 Meter			Ketinggian 10 Meter			Ketinggian 15 Meter		
Su dut	Lux	Akur asi	Su dut	Lux	Akur asi	Su dut	Lux	Akur asi
0°	24515	66.6%	0°	24515	100%	0°	24515	33.3%
45°	18761	75%	45°	18761	50%	45°	18761	0%

60°	31475	66.6%	60°	31475	33.3%	60°	31475	0%
-----	-------	-------	-----	-------	-------	-----	-------	----

Sedangkan pada tabel diatas menunjukkan hasil perhitungan akurasi yang didapat dari hasil percobaan 2 dengan ketinggian 5 meter, 10 meter, dan 15 meter pada sudut 0° dengan intensitas cahaya 24515 lx dan, pada sudut 45° dengan intensitas cahaya 18761 lx, dan pada sudut 60° intensitas cahaya 31475 lx. Maka didapatkan hasil seperti tabel diatas.

TABEL 3

Tabel Rata - Rata Akurasi dari 2 Percobaan Kondisi Drone Diam

Rata – Rata Akurasi Kondisi Drone Diam	
Ketinggian 5 Meter	53.90%
Ketinggian 10 Meter	55.50%
Ketinggian 10 Meter	9.70%

Tabel diatas menunjukkan hasil perhitungan dari rata – rata nilai akurasi yang didapat dari percobaan 1 dan percobaan 2, dimana nilai yang paling optimal berada pada ketinggian 10 meter dengan kondisi drone diam dengan nilai rata- rata akurasi sebesar 55.50%.

TABEL 4

Tabel Percobaan 1 Akurasi Kondisi Drone Bergerak

Percobaan 1 Kondisi Drone Bergerak								
Ketinggian 5 Meter			Ketinggian 10 Meter			Ketinggian 15 Meter		
Kec epat an	Lu x	Akur asi	Kec epat an	Lu x	Akur asi	Kec epat an	Lu x	Akur asi
0.1 m/s	46966	50%	0.1 m/s	46966	25%	0.1 m/s	46966	0%

Pada tabel diatas menunjukkan hasil perhitungan akurasi yang didapat dari hasil percobaan 1 dengan ketinggian 5 meter, 10 meter, dan 15 meter dengan kecepatan sebesar 0.1 m/s dengan intensitas cahaya sebesar 46966. Maka didapatkan hasil seperti tabel diatas.

TABEL 5

Tabel Percobaan 2 Akurasi Kondisi Drone Bergerak

Percobaan 2 Kondisi Drone Bergerak								
Ketinggian 5 Meter			Ketinggian 10 Meter			Ketinggian 15 Meter		
Kece pata n	L u x	Akur asi	Kece pata n	L u x	Akur asi	Kece pata n	L u x	Akur asi
0.1 m/s	561	40%	0.1 m/s	561	20%	0.1 m/s	561	0%

Pada tabel diatas menunjukkan hasil perhitungan akurasi yang didapat dari hasil percobaan 2 dengan ketinggian 5 meter, 10 meter, dan 15 meter dengan kecepatannya sebesar 0.1 m/s dengan intensitas cahaya sebesar 46966. Maka didapatkan hasil seperti tabel diatas.

TABEL 6

Tabel Rata - Rata Akurasi dari 2 Percobaan Kondisi Drone Bergerak

Rata – Rata Akurasi Kondisi Drone Bergerak	
Ketinggian 5 Meter	45%
Ketinggian 10 Meter	22.5%
Ketinggian 10 Meter	0%

Tabel diatas menunjukkan hasil perhitungan dari rata – rata nilai akurasi yang didapat dari percobaan 1 dan percobaan 2, dimana nilai yang paling optimal berada pada ketinggian 5 meter dengan kondisi drone bergerak dengan kecepatan sebesar 0.1 m/s dengan nilai rata- rata akurasi sebesar 45%.

Dari percobaan – percobaan yang dilakukan setiap per-sekian menit dinaikan *value* yang sudah ditentukan dari ketinggian 5 meter naik ke 10 meter dan dinaikan lagi ke 15 meter dengan *angle* atau posisi kamera berada di tiap sudut 0°, 45°, dan 60° pada setiap tingkat ketinggian untuk kondisi drone dalam keadaan diam. Sedangkan untuk kondisi drone dalam keadaan bergerak dengan kecepatan 0.1 m/s dengan ketinggian yang sama dengan ketinggian yang digunakan untuk drone dengan keadaan diam.

V. KESIMPULAN

Kesimpulan yang didapat pada penelitian dan pengujian serta analisis pada Tugas Akhir ini adalah sebagai berikut:

1. Algoritma R-CNN dapat melakukan deteksi dan mengklasifikasi sapi yang ada di peternakan dengan menggunakan *Drone* pada ketinggian serta posisi yang sudah ditentukan, dan dapat memudahkan untuk mengetahui keadaan hewan ternak dalam hal ini adalah sapi dan algoritma R-CNN tidak mendeteksi dan mengklasifikasi Non-Sapi setelah melakukan *training* pada algoritma.
2. Algoritma R-CNN pada saat pengujian didapatkan bahwasannya algoritma dapat mendeteksi suatu objek dengan tingkat akurasi berada 5 meter sampai 10 meter dalam kondisi diam maupun bergerak. Dan untuk tingkat akurasi yang paling optimal berada pada ketinggian 10 meter dengan tingkat akurasi 55,50% dengan kondisi drone diam, sedangkan untuk kondisi drone bergerak dengan kecepatan sekitar 0,1 m/s tingkat akurasi paling optimal berada pada jarak 5 meter dengan rata-rata akurasinya 50%.

REFERENSI

- [1] R. Dharmadi, “Convolutional Neural Net untuk Deteksi Objek | by Richard Dharmadi | Medium,” 2018. https://medium.com/@richad_/convolutional-neural-net-untuk-deteksi-objek-f14d72f11ba6 (accessed Feb. 04, 2023).
- [2] G. D. Nursyafitri, “Kenali Penggunaan Computer Vision dalam Data Science,” 2022. <https://dqlab.id/kenali-penggunaan-computer-vision-dalam-data-science> (accessed Feb. 04, 2023).
- [3] W. Piao, “Contoh Penerapan Computer Vision Dalam Kehidupan Sehari-Hari.” <https://labkom99.com/2022/05/contoh-penerapan-computer-vision.html> (accessed Feb. 17, 2023).
- [4] A. Primawati, I. Mutia, and D. Marlina, “Analisis Klasifikasi Populasi Ternak Kambing Dan Domba Dengan Model Convolutional Neural Network,” *Fakt. Exacta*, vol. 14, no. 1, p. 22, 2021, doi: 10.30998/faktorexacta.v14i1.8734.
- [5] Q. Lina, “Apa itu Convolutional Neural Network? | by QOLBIYATUL LINA | Medium.” <https://medium.com/@16611110/apa-itu-convolutional-neural-network-836f70b193a4> (accessed Feb. 17, 2023).
- [6] R. Maulid Pradistya, “Mengenal Tensorflow, Library untuk Keperluan Machine Learnin...,” 2021. <https://www.dqlab.id/mengenal-tensorflow-library-untuk-keperluan-machine-learning-python> (accessed Feb. 04, 2023).
- [7] D. J. P. Manajang, S. R. U. A. Sompie, and A. Jacobus, “Implementasi Framework Tensorflow Object Detection API Dalam Mengklasifikasi Jenis Kendaraan Bermotor,” *J. Tek. Inform.*, vol. 15, no. 3, pp. 171–178, 2020.
- [8] B. Tandika, “Bahasa Pemrograman Phyton: Arti, Fungsi, dan Keunggulan - Glints Blog,” 2022. <https://glints.com/id/lowongan/apa-itu-bahasa-pemrograman-python/#.Y94o6HZBzIU> (accessed Feb. 04, 2023).
- [9] G. Nurvinda K, “Yuk Kenali Kegunaan dan Fungsi Python di Dunia Industri,” May 27, 2022. <https://www.dqlab.id/yuk-kenali-kegunaan-dan-fungsi-python-di-dunia-industri> (accessed Feb. 17, 2023).
- [10] A. Budi K, “Implementasi OpenCV Pada Industri dan Kehidupan Sehari-hari,” Aug. 17, 2022. <https://crocodic.com/implementasi-opencv-pada-industri-dan-kehidupan-sehari-hari/> (accessed Feb. 17, 2023).
- [11] Srimulia, “Mengenal OpenCV Dalam Python: Pengertian , Sejarah, Dukungan pada OS, Fitur-fitur - idmetafora,” 2022. <https://idmetafora.com/news/read/1177/Mengenal-OpenCV-Dalam-Python-Pengertian-Sejarah-Dukungan-pada-OS-Fitur-fitur.html> (accessed Feb. 04, 2023).
- [12] A. Muhardian, “Memahami: Apa itu

- Numpy pada Python?”
<https://www.petanikode.com/python-numpy/> (accessed Feb. 04, 2023).
- [13] Gubuk Pintar, “Mengenal Numpy Python dan Contoh Penggunaannya - Gubuk Pintar,” Nov. 29, 2020. <https://www.gubukpintar.com/2020/11/numpy-adalah.html> (accessed Feb. 17, 2023).
- [14] A. Nabila, “Scipy dan KMeans pada Python. SciPy adalah suatu modul untuk... | by ANISA NABILA | Medium,” 2019. <https://medium.com/@16611111/scipy-dan-kmeans-pada-python-ed8f9849ceab> (accessed Feb. 04, 2023).
- [15] L. Afifah, “Apa itu Confusion Matrix di Machine Learning? - IlmudataPy.” <https://ilmudatapy.com/apa-itu-confusion-matrix/> (accessed Feb. 18, 2023).
- [16] K. Setyo Nugroho, “Confusion Matrix untuk Evaluasi Model pada Supervised Learning | by Kunchahyo Setyo Nugroho | Medium,” Nov. 13, 2019. <https://ksnugroho.medium.com/confusion-matrix-untuk-evaluasi-model-pada-unsupervised-machine-learning-bc4b1ae9ae3f> (accessed Feb. 18, 2023).