

## Implementasi Metode Faster Region Convolutional Neural Network (Faster R-CNN) Untuk Pengenalan Jenis Burung Lovebird

Fino Charli<sup>1</sup>, Hadi Syaputra<sup>\*2</sup>, Muhammad Akbar<sup>3</sup>, Siti Sauda<sup>4</sup>, Febriyanti Panjaitan<sup>5</sup>

<sup>1,2,3,4,5</sup>Informatics Departement, Bina Darma University, Palembang, Indonesia

Email: <sup>1</sup>[finocharli77@gmail.com](mailto:finocharli77@gmail.com), <sup>2</sup>[hadisyputra@binadarma.ac.id](mailto:hadisyputra@binadarma.ac.id),

<sup>3</sup>[muhammadakbar@binadarma.ac.id](mailto:muhammadakbar@binadarma.ac.id), <sup>4</sup>[sitisauda@binadarma.ac.id](mailto:sitisauda@binadarma.ac.id),

<sup>5</sup>[febriyantipanjaitan@binadarma.ac.id](mailto:febriyantipanjaitan@binadarma.ac.id)

### Abstrak

Lovebird merupakan burung sosial yang hidup berkelompok di alam bebas. Burung dari genus *Agapornis* ini ukuran tubuhnya relative mungil bila dibanding burung berparuh bengkok lainnya. Panjangnya sekitar 13-17 cm dengan bobot 30-60 gram. Ragam dan jenis burung lovebird sangat banyak namun bila dilihat dengan pendekatan taksonomi, hanya terdapat 9 spesies burung lovebird. Penelitian metode Faster Region Convolutional Neural Network ini digunakan untuk mengenali jenis burung lovebird guna mencari tahu motif atau warna asli dari jenis burung lovebird tertentu berdasarkan tingkat akurasi dari hasil deteksi objek. Faster R-CNN merupakan pengembangan Convolutional Neural Network (CNN) yang ditemukan oleh Ross Girshick. Sampai saat ini metode ini masih terus dikembangkan dan digunakan untuk pengenalan objek atau objek recognition. Penelitian ini menggunakan data set burung lovebird dengan total 808 gambar dan 8 (kelas) yaitu *Agapornis Personata*, *Agapornis Fischeri*, *Agapornis Lilianae*, *Agapornis Nigrigenis*, *Agapornis Roseicollis*, *Agapornis Taranta*, *Agapornis Pullarius*, *Agapornis Canus*. Hasil dari proses learning didapatkan akurasi sebesar 96% dengan loss sebesar 0,2% . sehingga dapat disimpulkan bahwa metode Faster R-CNN dapat diterapkan pada pengenalan jenis burung lovebird.

Kata Kunci: Faster Regional Neural Network, object detection, convolutional neural network, image, deep learning, lovebird

### 1. PENDAHULUAN

*Deep learning* merupakan suatu kelas dari teknik pembelajaran mesin yang memanfaatkan banyak lapisan dalam pemrosesan informasi untuk ekstraksi dan informasi. *Deep learning* lahir pada tahun 2006 diperkenalkan oleh geoffrey hinton untuk menjelaskan algoritma baru yang dapat membuat komputer melihat (Purnama, 2019). *Deep learning* mulai dikenal pada kompetisi pengenalan citra *ImageNet Large Scale Visual Recognition Competition (ILSVRC)* tahun 2012 (Suyanto, Kurniawan Nur Ramadhani, Satria Mandala, 2019), teknik ini menjadi pemenang pertama di

(ILSVRC) dengan akurasi yang jauh lebih tinggi dibanding *shallow learning* (machine learning) konvensional.

Ada banyak Teknik pengenalan objek dengan *deep learning*, diantaranya Region based Convolutional Neural Network (R-CNN), Fast R-CNN, Faster R-CNN, dan You Only Look Once (YOLO) (Gandhi, 2018). Faster R-CNN merupakan Teknik pengenalan objek oleh mesin yang dikembangkan oleh Ross Girshick dkk pada tahun 2015 (Gandhi, 2018). Faster R-CNN memiliki 3 modul utama yang pertama *deep fully convolutional network* yang mengusulkan region, modul kedua adalah detector Fast R-CNN yang menggunakan region yang diusulkan dan RPN yang berfungsi mengarahkan Fast R-CNN untuk melihat (Ren et al., 2016).

Untuk melakukan pengenalan objek terlebih dahulu kita mengumpulkan dataset, dataset merupakan kumpulan citra yang kita kumpulkan dan diberi label oleh manusia yang kemudian di proses oleh mesin untuk sebagai pembelajaran sebelum melakukan klasifikasi (Pumama, 2019). burung lovebird merupakan hewan asli daratan Afrika dan memiliki 9 (sembilan) jenis diantaranya : *Agapornis personata*, *Agapornis fischeri*, *Agapornis lilianae*, *Agapornis nigrigenis*, *Agapornis roseicollis*, *Agapornis taranta*, *Agapornis pullarius*, *Agapornis canus*, dan *Agapornis canus*. Lovebird merupakan burung social berparuh bengkok asli daratan Afrika. Dalam bebas burung lovebird dapat terdiri dari 5 hingga 20 ekor per kelompok. Sedangkan burung dewasa akan hidup berpasangan dengan sangkar tak jauh dari koloninya, burung dari genus *Agapornis* ini ukuran tubuhnya relative mungil dengan bobot 30- 60 gram. Mutasi warna yang banyak terjadi pada burung Lovebird yang ada di Indonesia adalah sebuah hasil rekayasa genetik dengan mengawin silangkan burung Lovebird yang tidak sejenis sehingga banyak dijumpai warna yang berbeda dalam satu jenis burung. Pada penelitian ini penulis akan menggunakan Algoritma Faster Region Convolutional Neural Network (Faster R-CNN) untuk mengenali jenis burung Lovebird pada sebuah citra input.

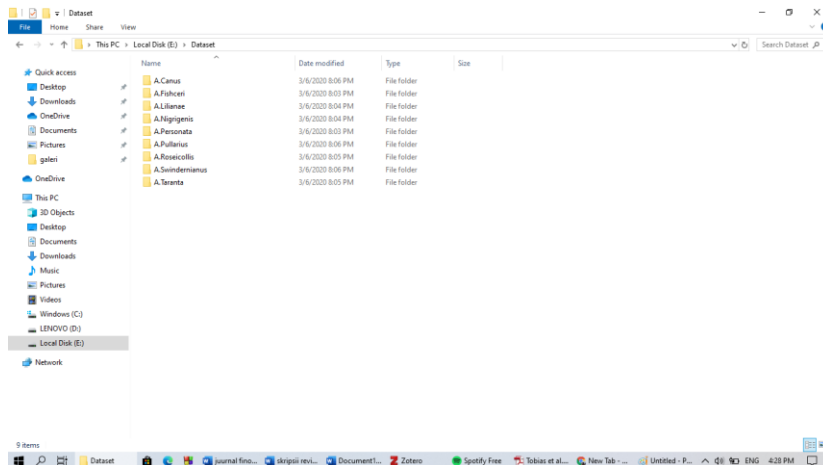
## 2. METODOLOGI PENELITIAN

### 2.1. Tahapan Penelitian

#### 1) Pengumpulan Gambar

Data yang diambil dan dikumpulkan hanyalah 8 (delapan) jenis dari 9 (Sembilan) jenis lovebird yang ada. Masing- masing kelas berjumlah 100 gambar berformat .jpg. hal ini dilakukan agar data uji dan data latih mendapat bobot training yang sama sehingga menghasilkan bias rendah pada data latih dan data uji. Mengingat jenis Lovebird jenis *Agapornis swindernianus* susah

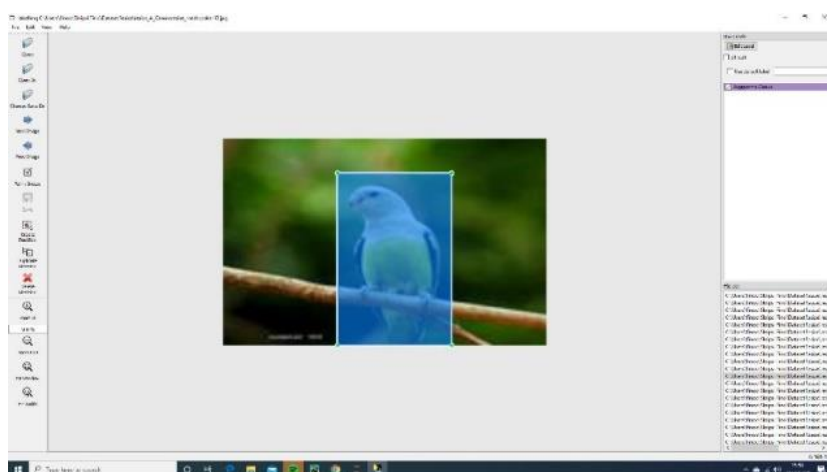
untuk ditangkarkan dan sangat Langkah sehingga sulit untuk mendapatkan 100 buah gambar jenis burung ini.



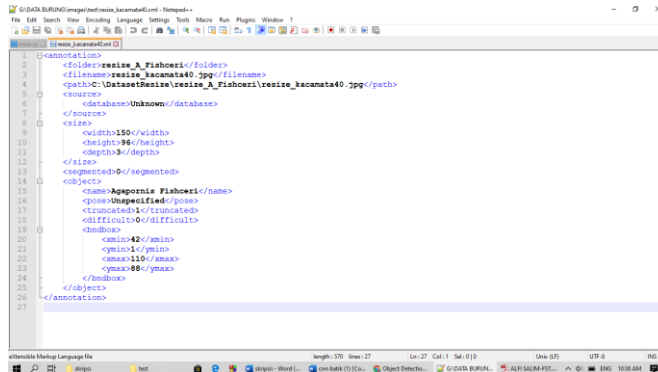
Gambar 2. folder per serratus gambar

## 2) Pelabelan data

Pemberian label pada gambar atau anotasi ini bertujuan untuk mendapatkan ciri khusus yang merepresentasikan masing-masing objek yang terdapat dalam gambar. Ciri khusus tersebut nantinya akan dijadikan learning dalam proses training.



Gambar 3. Pelabelan data



Gambar 4. Hasil Anotasi

Hasil anotasi akan berupa koordinat objek dalam bentuk file xml seperti gambar 4.

### 3) Membagi Dataset

Pada tahap ini dataset dibagi menjadi dua bagian yaitu data train dan data tes. Sesuai dengan konsep split dataset, data train harus lebih banyak dari data test. Pada penelitian ini pembagian dataset merujuk pada penelitian yang pernah dilakukan sebelumnya (Susanto et al., 2019). Dataset dibagi menjadi 80% untuk data train dan 20% untuk data test.

### 4) Konversi file xml ke .csv

Konversi file xml menjadi file .csv bertujuan untuk membuat himpunan data yang semula tidak rapi menjadi bentuk tabel yang berisi fitur yang akan dipelajari dalam proses konvolusi. Berikut hasil konversi file xml hasil anotasi yang sudah dikonversi kedalam bentuk .csv

filename	width	height	class	xmin	ymin	xmax	ymax
resize_13	150	96	Agapornis	57	2	105	63
resize_35	150	96	Agapornis	50	11	120	96
resize_41	150	96	Agapornis	44	3	129	92
resize_46	150	96	Agapornis	64	27	99	74
resize_48	150	96	Agapornis	40	13	137	73
resize_49	150	96	Agapornis	59	12	122	62
resize_49	150	96	Agapornis	1	26	89	64
resize_49	150	96	Agapornis	11	13	120	81
resize_49	150	96	Agapornis	54	26	64	81
resize_49	150	96	Agapornis	66	34	95	83
resize_49	150	96	Agapornis	24	11	148	95
resize_49	150	96	Agapornis	20	3	124	93
resize_49	150	96	Agapornis	17	34	78	84
resize_49	150	96	Agapornis	79	17	114	85
resize_49	150	96	Agapornis	35	9	143	92
resize_49	150	96	Agapornis	47	9	98	93
resize_49	150	96	Agapornis	36	6	113	92
resize_49	150	96	Agapornis	32	6	129	88
resize_49	150	96	Agapornis	23	2	118	96
resize_49	150	96	Agapornis	64	36	87	70
resize_49	150	96	Agapornis	54	15	98	82
resize_49	150	96	Agapornis	51	3	135	95

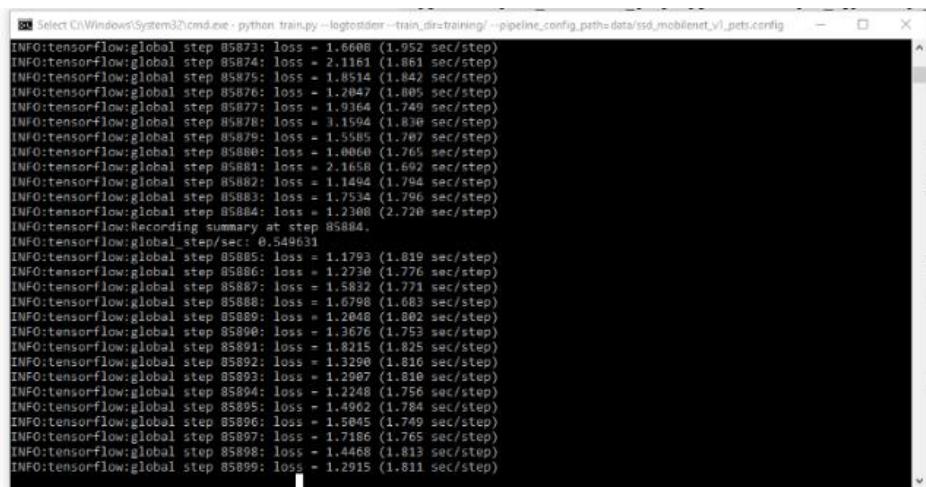
Gambar 5. Hasil konversi file xml

a. Mengenerate file ke bentuk TFRecord

Setelah mengumpulkan semua fitur dalam bentuk csv maka Langkah selanjutnya ialah mengubah file .csv yang berisi fitur2 tadi kedalam bentuk biner agar dapat diolah menggunakan tensorflow

b. Training Data

Training data adalah proses yang memakan waktu lama. Training data dilakukan dengan cara menjalankan sebuah script yang berisi source code dan konfigurasi yang telah kita setting pada komputer kita. Training dilakukan sebanyak 2000000 step yang mana setiap step ditampilkan hasil dari pembelajaran (learning()) sistem. Setiap step akan menampilkan sebuah variabel yang disebut loss. Loss menunjukkan apakah model berlatih dengan baik atau tidak. Untuk lebih jelasnya kita dapat melihatnya dalam Inference graph yang merupakan catatan atau rekaman selama kita melakukan training.



```
Select C:\Windows\System32\cmd.exe - python train.py --logtostderr --train_dir=training/ --pipeline_config_path=data/ssd_mobilenet_v1_pets.config
INFO:tensorflow:global step 85873: loss = 1.6608 (1.952 sec/step)
INFO:tensorflow:global step 85874: loss = 2.1161 (1.861 sec/step)
INFO:tensorflow:global step 85875: loss = 1.8514 (1.842 sec/step)
INFO:tensorflow:global step 85876: loss = 1.2047 (1.805 sec/step)
INFO:tensorflow:global step 85877: loss = 1.9164 (1.749 sec/step)
INFO:tensorflow:global step 85878: loss = 3.1594 (1.830 sec/step)
INFO:tensorflow:global step 85879: loss = 1.5585 (1.707 sec/step)
INFO:tensorflow:global step 85880: loss = 1.0060 (1.765 sec/step)
INFO:tensorflow:global step 85881: loss = 2.1658 (1.692 sec/step)
INFO:tensorflow:global step 85882: loss = 1.1494 (1.794 sec/step)
INFO:tensorflow:global step 85883: loss = 1.7534 (1.796 sec/step)
INFO:tensorflow:global step 85884: loss = 1.2308 (2.720 sec/step)
INFO:tensorflow:Recording summary at step 85884.
INFO:tensorflow:global step/sec: 0.549631
INFO:tensorflow:global step 85886: loss = 1.1793 (1.819 sec/step)
INFO:tensorflow:global step 85886: loss = 1.2730 (1.776 sec/step)
INFO:tensorflow:global step 85887: loss = 1.5832 (1.771 sec/step)
INFO:tensorflow:global step 85888: loss = 1.6798 (1.683 sec/step)
INFO:tensorflow:global step 85889: loss = 1.2048 (1.802 sec/step)
INFO:tensorflow:global step 85890: loss = 1.3676 (1.753 sec/step)
INFO:tensorflow:global step 85891: loss = 1.8215 (1.825 sec/step)
INFO:tensorflow:global step 85892: loss = 1.3290 (1.810 sec/step)
INFO:tensorflow:global step 85893: loss = 1.2907 (1.810 sec/step)
INFO:tensorflow:global step 85894: loss = 1.2248 (1.756 sec/step)
INFO:tensorflow:global step 85895: loss = 1.4962 (1.784 sec/step)
INFO:tensorflow:global step 85896: loss = 1.5045 (1.749 sec/step)
INFO:tensorflow:global step 85897: loss = 1.7186 (1.765 sec/step)
INFO:tensorflow:global step 85898: loss = 1.4468 (1.813 sec/step)
INFO:tensorflow:global step 85899: loss = 1.2915 (1.811 sec/step)
```

Gambar 6. Proses training

c. Menyimpan Hasil frozen graph

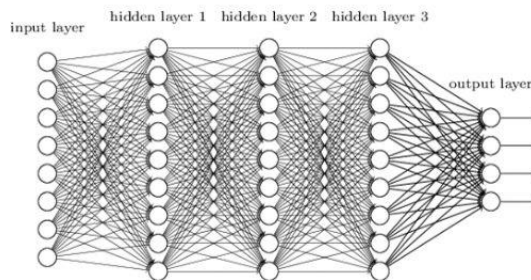
selama proses training, tensorboard akan merekam semua proses yang terjadi pada saat proses berlangsung, file terekam dalam bentuk .ckpt (checkpoint) dalam folder training. File yang tersimpan dan merekam step terakhir akan dikonversi menjadi sebuah model hasil pelatihan dengan format protobuf dengan ekstensi file ".pb"

Gambar 7. model training tensorflow dan checkpoint

### 2.3 Deep Learning

*Deep learning* adalah suatu teknik pada machine learning yang memanfaatkan banyak layer pada pengolahan informasi non-linier yang digunakan untuk melakukan ekstraksi fitur pengenalan pola dan klasifikasi (Deng dan Yu, 2014).

#### Deep neural network

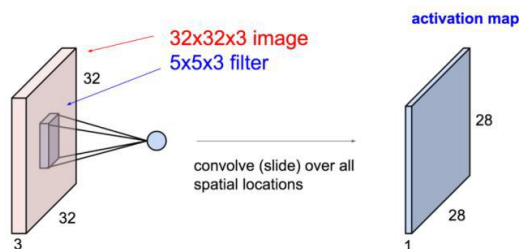


Gambar 8. Deep Neural Network

### 2.4 Convolutional Neural Network

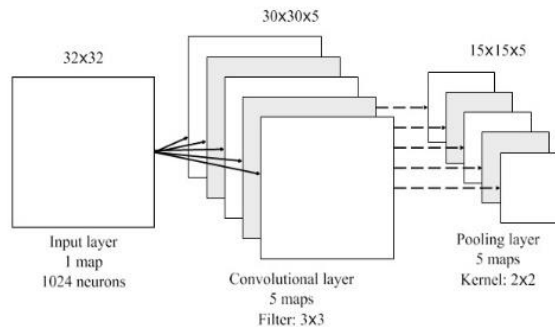
Convolutional Neural Network merupakan salah satu jenis neural network yang biasanya digunakan dalam pengolahan data image (Santoso & Ariyanto, 2018). Arsitektur CNN terdiri atas satu lapisan (input layer), satu lapis keluaran (output layer), dan lapis tersembunyi (hidden layer). Lapis tersembunyi umumnya berisi *convolutional layer*, *pooling layer*, *normalization layer*, *ReLU layer*, *fully connected layer*, dan *loss layer* (Suyanto, Kurniawan Nur Ramadhani, Satria Mandala, 2019). Pada proses training terdapat 3 tahapan yaitu convolutional layer, pooling layer, dan fully connected layer (Santoso & Ariyanto, 2018). Berikut ulasan 3 layer tersebut:

- 1) Convolutional layer, terdiri dari neuron yang tersusun sedemikian rupa sehingga membentuk sebuah filter dengan tinggi dan panjang (pixels). Tiap filter memiliki nilai dan nilai dari filter inilah yang menjadi parameter yang akan di-update dalam proses learning.



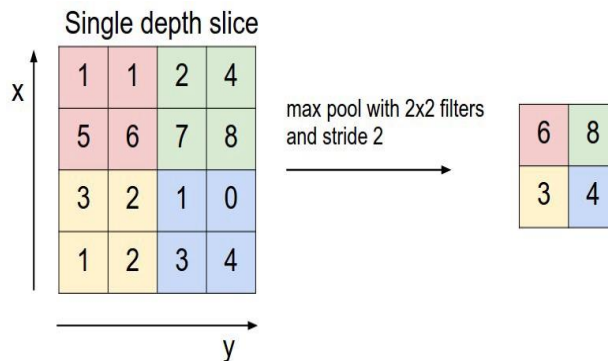
Gambar 9. Convolutional Layer

- 2) Pooling layer, pooling layer berfungsi menjaga ukuran data Ketika convolution dilakukan, yaitu dengan melakukan reduksi sampel (down sampling) contoh gambar 2.2 dengan pooling, kita dapat merepresentasikan data menjadi lebih kecil, mudah dikelola, dan mudah mengontrol overfitting.



Gambar 10. proses pooling untuk mereduksi data

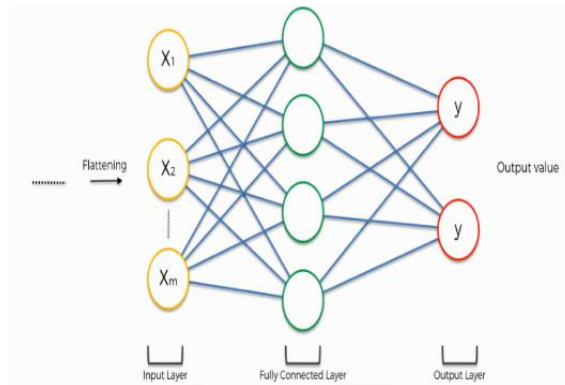
Pada umumnya proses pooling dilakukan menggunakan max pooling atau average pooling. Caranya adalah dengan memilih nilai maksimum atau nilai rata-rata dalam suatu area tertentu contoh gambar 10.



Gambar 2.4 Teknik Max pooling

- 3) Fully Connected layer, pada lapisan yang terhubung secara penuh (fully connected layer), setiap neurons memiliki koneksi penuh ke semua aktivasi dalam lapisan sebelumnya. Hal ini sama persis dengan MLP, yaitu komputasi menggunakan suatu perkalian matriks yang diikuti dengan bias offset

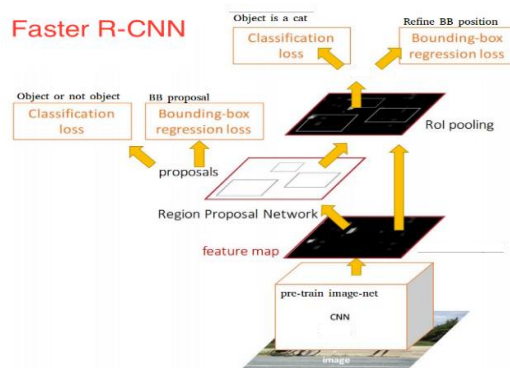




Gambar 11. Fully Connected Layer

#### 2.4 Faster Region Convolutional Neural Network

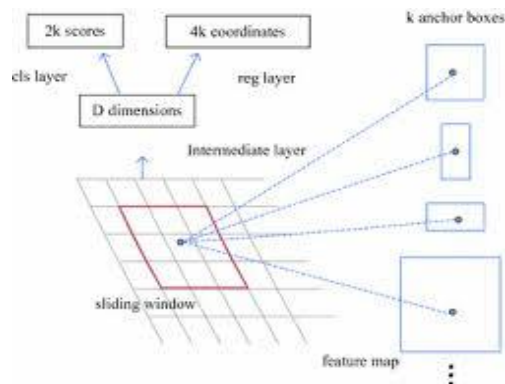
Faster R-CNN merupakan pengembangan dari Fast R-CNN. Sebagai generasi terbaru dari metode objek selection menunjukan hasil yang mengesankan pada objek deteksi



Gambar 12. Arsitektur dari Faster R-CNN (Le, 2020)

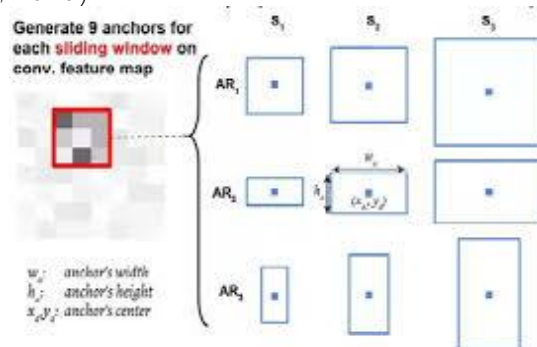
Gambar 12. menunjukkan arsitektur dari faster R-CNN. Faster R-CNN memiliki Langkah-langkah sebagai berikut :

1. Region Proposal Network (RPN) yaitu tugas cepat yang berguna untuk mencari pada input gambar kemungkinan lokasi dari objek. Posisi dari objek yang ada pada gambar memiliki kemungkinan objek dibatasi dari wilayah yang diketahui sebagai region of interest (ROI). RPN mengambil gambar dari berbagai ukuran sebagai input dan output sekumpulan proposal objek persegi panjang, masing-masing dengan skor objektivitas.



Gambar 13 Region Proposal Network (RPN)

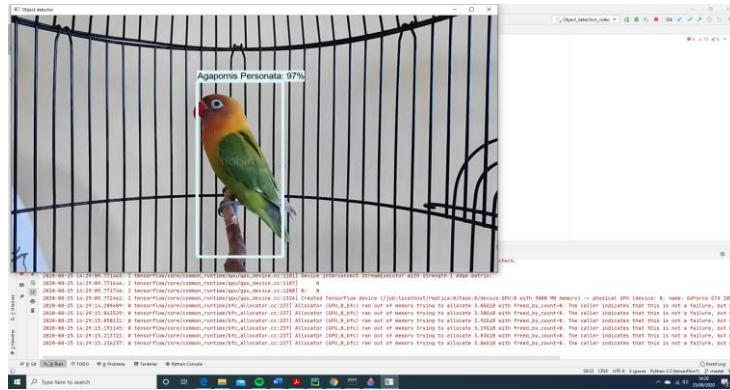
Pada Region Proposal Network awalnya gambar dimasukkan ke dalam jaringan Convolutional Neural Network. Gambar input diteruskan ke jaringan convolutional layer terakhir yang menampilkan feature map (Gilbert, 2019). Sliding window ditempatkan pada setiap bagian dari feature map. Sliding window mask biasanya diambil dari ukuran mask  $n \times n$ . Sesuai dengan setiap sliding window, secara bersamaan memprediksi beberapa region proposal, dimana jumlah proposal maksimum yang mungkin untuk setiap lokasi dilambangkan sebagai  $k$ . Layer reg memiliki output  $4k$  yang mengkodekan koordinat kotak  $k$ , dan layer cls menghasilkan  $2k$  yang memperkirakan probabilitas objek atau tidak objek untuk setiap proposal. Setiap anchor diposisikan di tengah dari sliding windows. Secara standar menggunakan 3 skala dan 3 rasio aspek yang menghasilkan  $k=9$  anchor pada setiap sliding windows (Gilbert, 2019). Klasifikasi menunjukkan probabilitas 0 atau 1 yang menunjukkan apakah wilayah tersebut berisi objek atau tidak dengan  $p^* = 1$  jika  $IoU > 0,7$ ,  $p^* = .1$  jika  $IoU < 0,3$ ,  $p^* = 0$  jika selain itu (Gilbert, 2019).



Gambar 14 Tiga Anchor dengan Aspect ratio dan Scaling

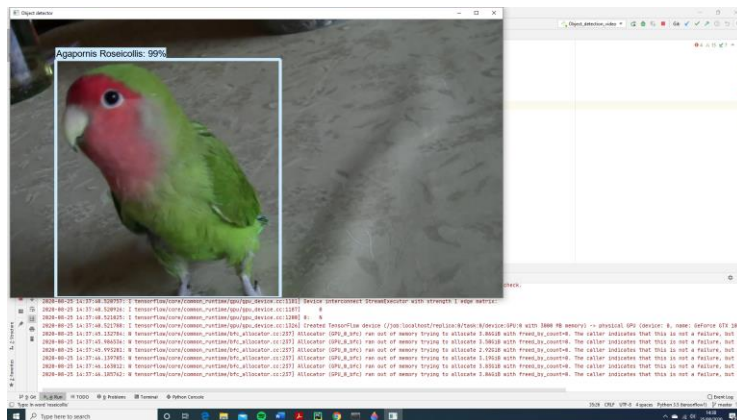
### 3. HASIL DAN PEMBAHASAN

Setelah melakukan proses training menggunakan algoritma deep learning dengan model Faster R-CNN dapat kita ketahui bahwa training dilakukan hingga nilai loss konsisten dibawah 0,05. Sehingga mampu menghasilkan akurasi yang cukup tinggi, hal tersebut dapat dilihat dari gambar berikut



Gambar 15 Agapornis Personata

Gambar diatas merupakan hasil capture dari video yang dijalankan dengan test\_video.py menggunakan pycharm. Hasil prediksi bertujuan untuk menentukan dua hal, apakah ada burung Lovebird dalam suatu citra yang diberikan ? jika ada jenis burung Lovebird apa yang ada pada citra ?.



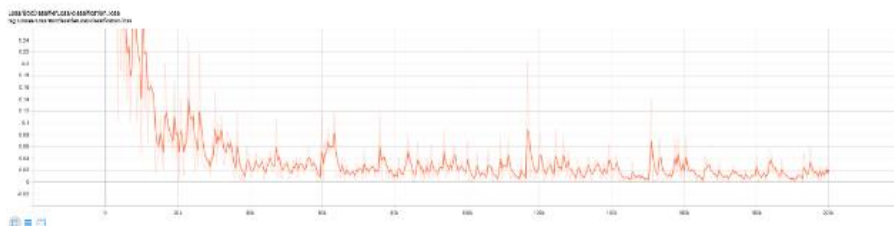
Gambar 16. Agapornis Roseicollis

Dari gambar 15 dan 16, dapat dilihat bounding box mengenali jenis burung lovebird hingga mampu memberikan persentase dari deteksi objek dari video

yang diberikan. Terlihat hasil akurasi sebesar 97% pada Agapornis personata dan 99% untuk video yang bergambar lovebird jenis Agapornis Roseicollis.

### 3.1 Inference graph

Seperti yang telah kita bahas sebelumnya proses training dapat diamati jika kita menggunakan framework tensorflow. Hal ini dikarenakan tools tensorboard yang memuat inference graph. Adapun beberapa grafik utama yang menjadi acuan penulis dalam mengamati proses training diantaranya grafic box classifier loss dan grafik RPN loss.



Gambar 17. Grafic box classifier loss/classification

Grafik ini menunjukkan kesalahan yang dilakukan model dalam mengklasifikasi objek yang terdeteksi menjadi berbagai macam kelas lain. Dengan demikian model mendapatkan nilai kesalahan rata-rata dibawah 0,03 dalam mendeteksi objek menjadi kelas yang seharusnya selama proses training



Gambar 18 Grafik RPN loss/objectness loss

Pada grafiki ini tensorboard menginformasikan kesalahan classfier dalam mengklasifikasikan sebuah objek atau sebuah latar belakang. Selama proses trainig tensorboard akan merekam semua proses yang terjadi pada saat proses berlangsung. File terekam dalam dile berekstensi .ckpt(checkpoint) dalam folder training. File yang tersimpan dan merekam step step terakhir akan dikonversi menjadi sebuah model hasil pelatihan dengan format protobuf dengan ekstensi file .pb

#### 4. KESIMPULAN

Berdasarkan hasil analisis yang telah dilakukan, maka dapat diperoleh beberapa kesimpulan sebagai berikut :

- 1) Ragam warna yang terdapat pada jenis burung Lovebird sudah dapat menjadi bahan pemembeda antara satu jenis ke jenis lainnya.
- 2) Metode Faster Region Convolutional Neural Network dapat memprediksi jenis burung lovebird.
- 3) Tingkat akurasi model yang didapatkan dari hasil pendeteksian jenis burung lovebird pada suatu citra digital menggunakan Faster region convolutional Neural Network berkisar 78% hingga 99%.

#### DAFTAR PUSTAKA

- Gandhi, R. (2018, July 9). *R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms*. Medium. <https://towardsdatascience.com/r-cnn-fast-r-cnn-faster-r-cnn-yolo-object-detection-algorithms-36d53571365e>
- Gilbert, N. (2019). *Implementasi Faster Region-Based Convolutional Neural Network pada Klasifikasi Bentuk Diagram iStar 2.0 untuk Requirements Modeling* [Bachelor\_thesis, Universitas Multimedia Nusantara]. <http://kc.umn.ac.id/10245/>
- Le, J. (2020, January 29). *The 5 Computer Vision Techniques That Will Change How You See The World*. Medium. <https://heartbeat.fritz.ai/the-5-computer-vision-techniques-that-will-change-how-you-see-the-world-1ee19334354b>
- Purnama, B. (2019). *Pengantar Machine Learning Konsep dan Praktikum dengan Contoh Latihan Berbasis R dan Python*. Informatika.
- Ren, S., He, K., Girshick, R., & Sun, J. (2016). Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *ArXiv:1506.01497 [Cs]*. <http://arxiv.org/abs/1506.01497>
- Santoso, A., & Ariyanto, G. (2018). IMPLEMENTASI DEEP LEARNING BERBASIS KERAS UNTUK PENGENALAN WAJAH. *Emitor: Jurnal Teknik Elektro*, 18(01), 15–21. <https://doi.org/10.23917/emitor.v18i01.6235>
- Susanto, K. N., Gunadi, K., & Setyati, E. (2019). Pengenalan Karakter pada Plat Nomor Indonesia dengan Tilt Correction dan Metode Faster R-CNN. *Jurnal Infra*, 7(1), 1–7.
- Suyanto, Kurniawan Nur Ramadhani, Satria Mandala. (2019). *Deep Learning Modernisasi Machine Learning Untuk Big Data*. Informatika.
- Tim Karya Tani Mandiri. (2018). *Rahasia Sukses Budidaya Burung Berkicau*. Nuansa Aulia.