



# Klasifikasi Pneumonia dengan *Deep Learning* *Faster Region Convolutional Neural Network* Arsitektur VGG16 dan ResNet50

Hafidz Daffa Hekmatyar<sup>1</sup>, Wahyu Andi Saputra<sup>2\*</sup>, Cipi Ramdani<sup>3</sup>

<sup>1,2</sup>*Teknik Informatika, Institut Teknologi Telkom Purwokerto,  
Jl. DI. Pandjaitan 128, Purwokerto, Jawa Tengah, Indonesia*

<sup>3</sup>*Sistem Informasi, Institut Teknologi Telkom Purwokerto,  
Jl. DI. Pandjaitan 128, Purwokerto, Jawa Tengah, Indonesia*

\*Email Penulis Koresponden: [andi@ittelkom-pwt.ac.id](mailto:andi@ittelkom-pwt.ac.id)

## **Abstrak:**

COVID-19 merupakan virus baru yang menyerang paru-paru dan diindikasikan dengan adanya Pneumonia. Penting bagi pakar kesehatan untuk memberikan perawatan tepat jika ditemukan Pneumonia pada kasus COVID-19. Namun demikian, kendala yang dihadapi dari citra toraks adalah mendeteksi adanya indikasi Pneumonia dalam mengklasifikasikan toraks dengan Pneumonia. Tujuan dari penelitian ini adalah mengklasifikasikan objek yang terdeteksi sebagai Pneumonia dengan menggunakan *Faster R-CNN*, yakni teknik yang mengkombinasikan algoritma *Region Proposal Network* (RPN) dan *Convolutional Neural Network* (CNN). Penelitian ini menggunakan metode *Faster R-CNN* untuk mendeteksi adanya Pneumonia pada pasien COVID-19 dengan menggunakan dua arsitektur CNN yang berbeda yaitu arsitektur VGG16 dan ResNet50. Dari pengujian yang diterapkan pada 500 data *X-ray* paru-paru Pneumonia COVID-19, arsitektur VGG16 mempunyai nilai akurasi tertinggi yaitu sebesar 85,8% sedangkan ResNet50 mempunyai nilai akurasi sebesar 84%. Dengan dikembangkannya penelitian ini diharapkan dapat membantu tenaga medis dalam mendeteksi Pneumonia secara dini pada pasien yang terkena virus COVID-19 dengan tepat.

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



## **Kata Kunci:**

*Klasifikasi Pneumonia;  
COVID-19;  
Faster R-CNN;  
Convolutional Neural Network*

## **Riwayat Artikel:**

Diserahkan 12 Maret 2022  
Direvisi 8 November 2022  
Diterima 9 November 2022  
Dipublikasi 26 Desember 2022

## **DOI:**

10.22441/incomtech.v12i3.15112

## 1. PENDAHULUAN

Pneumonia adalah infeksi paru-paru yang melibatkan *alveoli* paru-paru (kantong udara) dan dapat disebabkan oleh mikroba, termasuk bakteri, virus, atau jamur.

Pneumonia terjadi oleh peradangan paru-paru pada *alveolus* yang kemudian terjadi infeksi yang menutup kapiler sehingga terjadi gejala sesak napas [1]. Pneumonia biasanya disebabkan oleh bakteri dan paling sering oleh bakteri yang disebut *pneumococci*, terkadang bakteri *mycoplasma* Pneumonia juga terlibat. Virus juga dapat menyebabkan Pneumonia seperti virus flu babi, *influenza*, *coronavirus*, dan lain-lain [2].

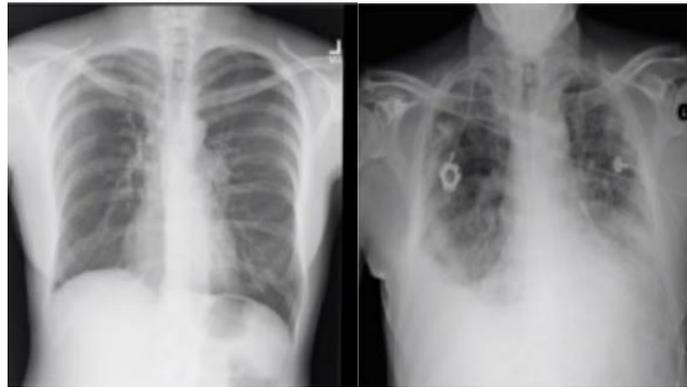
Pneumonia pada seseorang disebabkan oleh mikroba seperti virus, bakteri, dan jamur yang masuk ke dalam tubuh melalui hidung, mulut terkadang juga mata melalui udara, air dan juga dari sentuhan. Mikroba tersebut masuk ke dalam tubuh dan berkembang pada saluran pernafasan, kemudian menyerang paru-paru sehingga terjadi peradangan yang indikasikan dengan menumpuknya cairan yang disebabkan karena imun seseorang tidak dapat melawan mikroba/virus tersebut [3].

*Coronavirus* adalah suatu kelompok virus yang dapat menyebabkan penyakit pada hewan atau manusia. Beberapa jenis *coronavirus* diketahui menyebabkan infeksi saluran nafas pada manusia mulai dari batuk pilek hingga yang lebih serius seperti *Middle East Respiratory Syndrome* (MERS) dan *Severe Acute Respiratory Syndrome* (SARS) [4]. Pada saat ini virus COVID-19 menjadi sebuah pandemi yang menyerang berbagai negara dan sangat berbahaya.

Pada 31 Desember 2019 China melaporkan adanya penyakit baru. Pada penghujung tahun 2019, *World Health Organization* (WHO) di China mendapat pemberitahuan tentang adanya virus yang penyebabnya tidak diketahui. Infeksi pernapasan akut yang menyerang paru-paru, terdeteksi di kota Wuhan, Provinsi Hubei. Pada tanggal 30 Januari 2020, WHO mengumumkan virus tersebut sebagai darurat kesehatan masyarakat global. Beberapa waktu kemudian, tepatnya pada 11 Februari 2020, WHO mengumumkan adanya virus baru yang diberi nama *Coronavirus Disease 2019* (COVID-19) [5].

*World Health Organization* (WHO) resmi mengumumkan wabah COVID-19 sebagai pandemi global pada Rabu 11 Maret 2020 [6]. Per tanggal 8 September 2021 Virus COVID-19 telah menyerang lebih dari 222 juta orang di dunia dan di Indonesia lebih dari 4,1 juta orang terserang virus COVID-19, dan masih terus bertambah sampai saat ini di seluruh dunia [7].

Tanda-tanda seseorang yang terkena Pneumonia adalah demam tinggi, batuk berdahak, sesak napas dan kelemahan fisik [8]. Pneumonia juga bisa berakibat fatal dan menyebabkan kematian [9]. Deteksi Pneumonia merupakan hal yang sangat penting dalam mendiagnosa Pneumonia yang ada pada seseorang. Dengan menggunakan citra *X-ray*, tenaga medis dapat mempelajari dan mengetahui apakah seseorang mempunyai Pneumonia atau tidak. **Gambar 1** merupakan citra *X-ray* paru paru *normal* dan paru-paru Pneumonia.



Gambar 1. Citra Paru-Paru Normal (kiri) dan Pneumonia (kanan) [10]

Dengan semakin bertambahnya pasien positif COVID-19, deteksi Pneumonia menjadi sangat penting dalam upaya menekan gejala yang ditimbulkan oleh Pneumonia. Dalam mendiagnosis Pneumonia, diperlukan keahlian oleh seorang tenaga medis berdasarkan kepakarannya, yang artinya kemampuan menerjemahkan citra Pneumonia bersifat subjektif. Dengan dibuatnya penelitian ini diharapkan dapat membantu tenaga medis dalam mendiagnosis pasien Pneumonia yang ditimbulkan oleh virus COVID-19 secara dini, sehingga dapat mengurangi tingkat risiko kematian yang ditimbulkan oleh Pneumonia. Dari permasalahan tersebut, penelitian ini mengembangkan sebuah model untuk mendiagnosis lokasi Pneumonia dengan menggunakan citra X-ray untuk mempercepat penanganan pasien COVID-19 dengan menggunakan algoritma *Faster Region Convolutional Neural Network (Faster R-CNN)*.

Teknik *deep learning* adalah metode klasifikasi citra yang banyak diterapkan pada berbagai metode karena tingkat akurasi yang tinggi dan juga dapat mendeteksi dengan cepat [11]. Pada penelitian ini dikembangkan sebuah model pendeteksi Pneumonia berbasis *image classification* dengan mengambil data X-ray sebagai metode klasifikasi. Model yang akan dibuat pada penelitian ini menggunakan algoritma *Faster R-CNN* sebagai metode pendeteksi objek pada citra.

*Faster Region-based Convolutional Neural Network* atau biasa disingkat *Faster R-CNN* adalah sebuah metode yang digunakan untuk mendeteksi objek pada sebuah gambar. *Faster R-CNN* adalah metode *deep learning* yang menggunakan gabungan antara *Fast R-CNN* dan *RPN (Region Proposal Network)* sebagai arsitektur utamanya [12]. Tujuan dari penelitian ini adalah membuat sebuah model yang dapat mendeteksi lokasi Pneumonia COVID-19 dengan cepat dan mudah dengan algoritma *Faster R-CNN*.

Penelitian [13] meneliti tentang deteksi nodul kanker paru-paru dengan menggunakan citra 3D rongga toraks. Metode yang digunakan adalah *Faster R-CNN* dengan arsitektur *CmixNet*. Dari hasil penelitian diperoleh nilai *sensitivity* sebesar 98%. Penelitian lainnya dari [10] yang menerapkan arsitektur *ShuffleNet* dan *ResNet* untuk mendeteksi COVID-19. Hasil dari penelitian ini adalah nilai akurasi sebesar 86,93% dengan hanya 1125 citra X-ray toraks. Studi mengenai COVID-19 dan Pneumonia juga dilakukan oleh [14] dalam mendeteksi COVID-19 dengan menggunakan *VGG-16* dengan akurasi yang diperoleh adalah sebesar 92,86%. Berdasarkan studi pustaka penelitian yang berkaitan dengan klasifikasi

objek, menunjukkan bahwa *Faster R-CNN* dapat digunakan dalam mengklasifikasi objek berupa citra dengan pendekatan pembelajaran mesin. Oleh sebab itu, penelitian ini menerapkan *Faster R-CNN* sebagai *tool* atau piranti dalam mengklasifikasi objek.

Sementara itu, dalam melakukan evaluasi model pembelajaran mesin, *Mean Average Precision* (mAP) merupakan salah satu metode pengukuran yang populer dalam studi yang menerapkan *Faster R-CNN*. Penelitian [15] menerapkan mAP sebagai *metrics* dalam mengukur kinerja metode yang diajukan. Penelitian [15] juga membandingkan berbagai metode dalam mendeteksi objek dan menunjukkan hasil deteksinya dengan menggunakan *metrics* mAP. mAP dijelaskan dalam [16] dapat diterapkan pada riset yang melibatkan deteksi objek dan segmentasi seperti *Mobile SSD* dan *YOLO* dengan berbagai benchmark umum seperti Pascal, VOC dan COCO. Dalam studi *Faster R-CNN*, penelitian yang dilakukan oleh [17] dalam menunjukkan novelti pada studi *Faster R-CNN* juga menerapkan mAP dalam menunjukkan performa sistem yang dikembangkan. mAP hingga saat ini masih digunakan dan menjadi *metrics* yang populer dalam menunjukkan hasil kinerja sistem yang bertujuan untuk mendeteksi objek. Persamaan 1 menunjukkan formula dalam penerapan mAP [16].

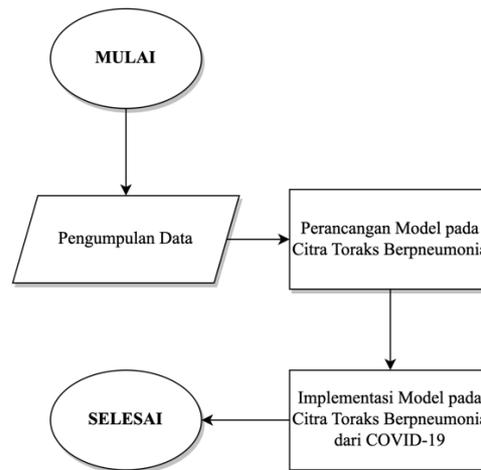
$$mAP = \frac{1}{n} \sum_{k=1}^{k=n} AP_k \quad (1)$$

dimana  $n$  adalah banyaknya kelas deteksi dan  $AP_k$  adalah *average precision* atau rerata nilai presisi dari kelas  $k$  (kelas yang dideteksi).

Dari berbagai studi yang telah dikaji, penelitian ini mengimplementasikan dua arsitektur CNN untuk membandingkan performa keduanya, yaitu arsitektur VGG16 dan ResNet50. Pengujian sistem diimplementasikan pada 500 citra paru-paru X-ray Pneumonia COVID-19, untuk mengetahui arsitektur mana yang terbaik dengan algoritma *Faster R-CNN*. Arsitektur VGG16 merupakan arsitektur yang mempunyai 16 *layer* yang terdiri dari 13 *convolution layer* dan 3 *fully connected layer*. Sedangkan ResNet50 merupakan arsitektur yang menerapkan konsep *residual block* yang terdiri atas 50 *layer residual block* [18]. Penelitian [18] menunjukkan bahwa *Faster R-CNN* memiliki akurasi yang tinggi dalam melakukan klasifikasi objek. Untuk itu, penelitian ini melakukan eksperimen dengan menerapkan *Faster R-CNN* tanpa *YOLO* dengan objek berupa citra Pneumonia X-ray agar diketahui akurasi klasifikasi Pneumonia. *Faster R-CNN* juga menunjukkan hasil akurasi yang tinggi sebagaimana penjelasan dari [19], bahwa pengujian yang dilakukan dengan menggunakan algoritma *Faster R-CNN* memiliki rata-rata presisi mAP yang tinggi.

## 2. METODE

Gambar 2 merupakan diagram alir pada penelitian yang dilakukan.



Gambar 2. Diagram Alir Penelitian

## 2.1. Mengumpulkan Data dan Merancang Model dari Citra Pneumonia

Setelah ditentukan metode yang akan diimplementasikan, penelitian ini mencari *dataset* yang tepat untuk merancang model yang akan dikembangkan. Adapun *dataset* yang digunakan bersumber dari:

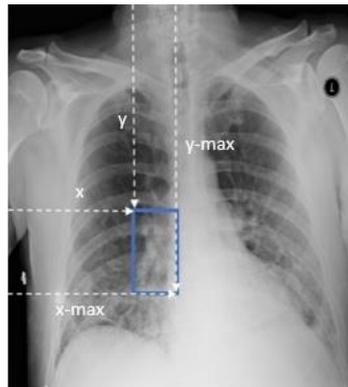
- Citra rontgen paru-paru Pneumonia beserta *ground-truth box* untuk pelatihan data, tautan: [www.kaggle.com/c/rsna-Pneumonia-detection-challenge](http://www.kaggle.com/c/rsna-Pneumonia-detection-challenge)
- Citra rontgen paru-paru Pneumonia COVID-19, tautan: [www.kaggle.com/tawsifurrahman/covid19-radiography-database](http://www.kaggle.com/tawsifurrahman/covid19-radiography-database)

Dari hasil pengumpulan data, diperoleh 30225 citra toraks Pneumonia yang masih mentah atau belum dilakukan *preprocessing*. Selain itu, diperoleh pula 500 citra toraks Pneumonia dari pasien COVID-19.

Setelah dilakukan pengumpulan data, dilakukan perancangan model dengan tahapan sebagai berikut:

### a. *Preprocessing*

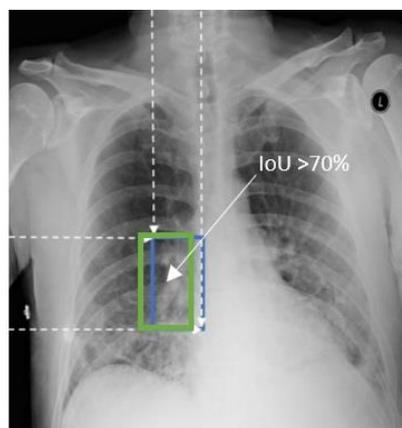
*Preprocessing* merupakan teknik untuk mengubah data mentah (*raw data*) menjadi lebih terstruktur untuk pengolahan selanjutnya. Proses tersebut bisa juga disebut langkah awal untuk mengambil semua informasi yang tersedia dengan cara membersihkan, memfilter, dan menggabungkan data-data tersebut. Untuk *dataset* yang digunakan di unduh pada *website* Kaggle yang terdiri atas gambar X-ray Pneumonia dan juga *ground-truth box* pada setiap gambar. *Dataset* yang digunakan masih dalam data mentah dan terdiri atas 30225 data. Untuk melatih dan menguji mAP (*mean averaging precision*), data memerlukan lokasi *ground truth box* pada gambar, maka harus dibuat sebuah parameter dengan *format* (x, x-max, y, y-max). Setelah data berhasil diolah, selanjutnya adalah membagi data dengan persentase 80% untuk pelatihan dan 20% untuk pengujian mAP. Gambar 3 merupakan contoh peletakan *bounding box* pada objek citra.



Gambar 3. Contoh Penerapan Parameter Bounding Boxes

#### b. Konfigurasi Model

Pada penelitian ini, *feature map* digunakan sebagai *proposed region* atau bagian gambar yang mungkin merupakan sebuah objek dan *classifier* (menggolongkan objek). Proses kedua adalah mengambil *feature map* dari lapisan konvolusi dasar ke lapisan *base layer* 3 x 3 untuk menentukan RPN dengan mempersiapkan *dataset* yang telah memiliki *ground-truth box*. *Anchor* merupakan area untuk menilai bahwa di area tersebut ada objek atau tidak dengan cara menghitung IoU (*Intersection Over Union*). Pada pembuatan model ditentukan presentase IoU terhadap objek yang dilatih yaitu sebesar 70%, jika persentase terhadap objek di bawah 70% maka lokasi tersebut tidak *valid*, jika persentase di atas 70% maka objek tersebut *valid* dan dapat meneruskan ke proses selanjutnya. Gambar 4 merupakan representasi IoU yang digunakan pada pelatihan.



Gambar 4. Representasi IoU yang Digunakan untuk Proses Pelatihan Model

Keluaran RPN menentukan dan memprediksi tingkat keberadaan objek (*objectness score*) pada tiap *region proposal*, Sehingga keluaran RPN adalah sejumlah *region proposal* beserta *objectness score* masing-masing yang disebut dengan *anchor*. Pada penelitian, digunakan 3 rasio *anchor box* yang berbeda dengan perbandingan 1:1, 1:2, 2:1, sehingga dari perbandingan tersebut dihasilkan 9 jenis *anchor*. Fitur tersebut dimasukkan ke dalam dua lapisan yang saling

terhubung sepenuhnya – lapisan kotak regresi (*regression layer*) dan klasifikasi *box* (*classifier layer*).

*Region of Interest* (ROI) merupakan teknik segmentasi dari proses pengolahan citra dimana pengguna mampu mengolah citra yang mengandung informasi data citra yang dikehendaki. Pada proses penelitian ini *roi pooling* digunakan untuk menggabungkan *region of interest* dari RPN dengan *feature map*. Dari proses tersebut maka dihasilkan *feature map* yang telah ada *bounding box* dan *classifier*.

Proses ketiga adalah menggabungkan fitur proposal, untuk mendapatkan probabilitas kelas dan menghitung regresi kotak setiap proposal agar mendapatkan prediksi *proposal region* yang bersifat final menggunakan algoritma R-CNN. R-CNN merupakan deteksi objek yang mengkombinasikan algoritma *Region Proposal Network* (RPN) dan *Convolutional Neural Network* (CNN). Pada penelitian ini, R-CNN digunakan untuk mencari *region* atau bagian gambar yang mungkin merupakan sebuah objek, dengan metode *region proposal*. Langkah terakhir adalah proses menggabungkan RPN dengan R-CNN untuk menggolongkan objek yang dideteksi apakah merupakan objek Pneumonia atau bukan.

Dalam mengimplementasikan metode *Faster R-CNN*, arsitektur yang digunakan adalah VGG16 dan ResNet50. VGG16 merupakan model CNN yang memanfaatkan *convolutional layer* dengan spesifikasi *convolutional filter* yang kecil (3x3). Dengan ukuran *convolutional filter* tersebut, kedalaman *neural network* dapat ditambah dengan lebih banyak lagi *convolutional layer*. Model VGG16 mempunyai 16 layer yang terdiri atas 13 *convolutional layer* dan 3 *fully connected layer*, dan terdapat 5 *pooling layer* yang bertipe *max pooling*. Adapun ilustrasi penerapan VGG16 secara umum diilustrasikan pada Gambar 5.

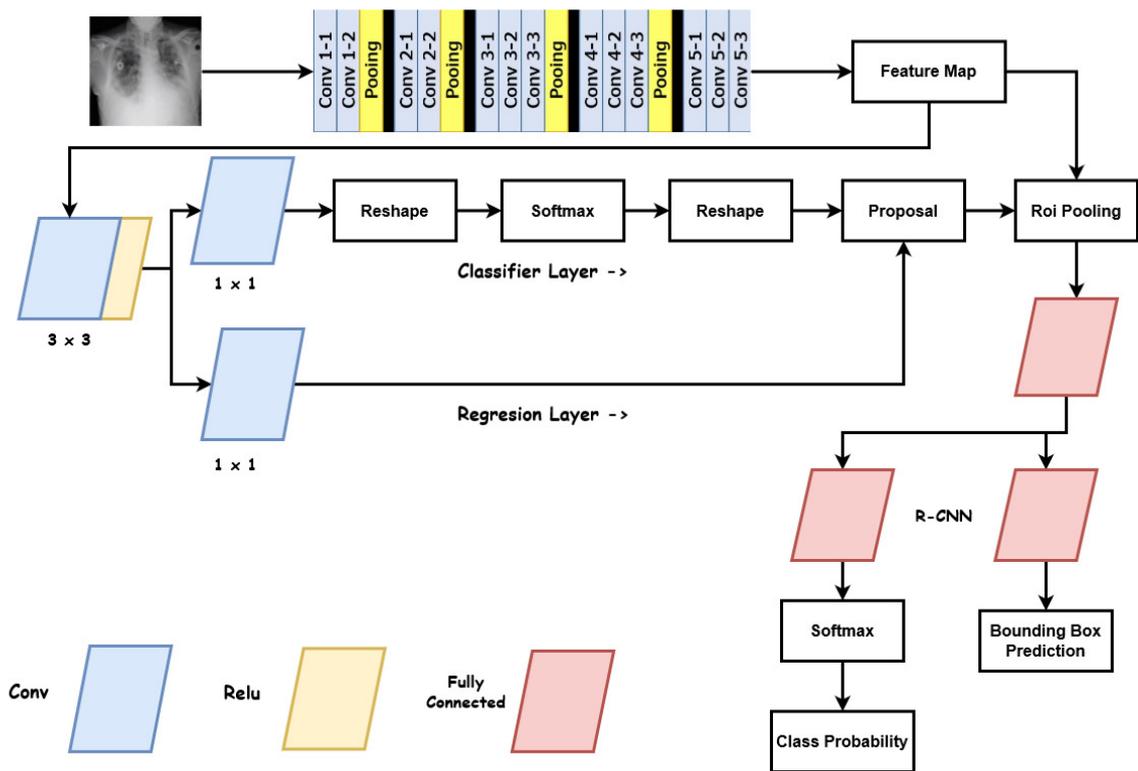
ResNet50 merupakan salah satu varian ResNet yang memiliki 50 layer. ResNet50 menggunakan 3 layer skip connection / residual block dan terdapat 1x1 convolution layer. Residual block pada ResNet dapat dilakukan apabila dimensi data masukan sama dengan dimensi data keluaran. Data lapisan awal dari arsitektur ResNet melakukan convolution 7 x 7 dan max pooling berukuran 3 x 3 dengan jumlah stride 2. Adapun ilustrasi penerapan ResNet50 secara umum disajikan pada Gambar 6.

#### c. Training Data

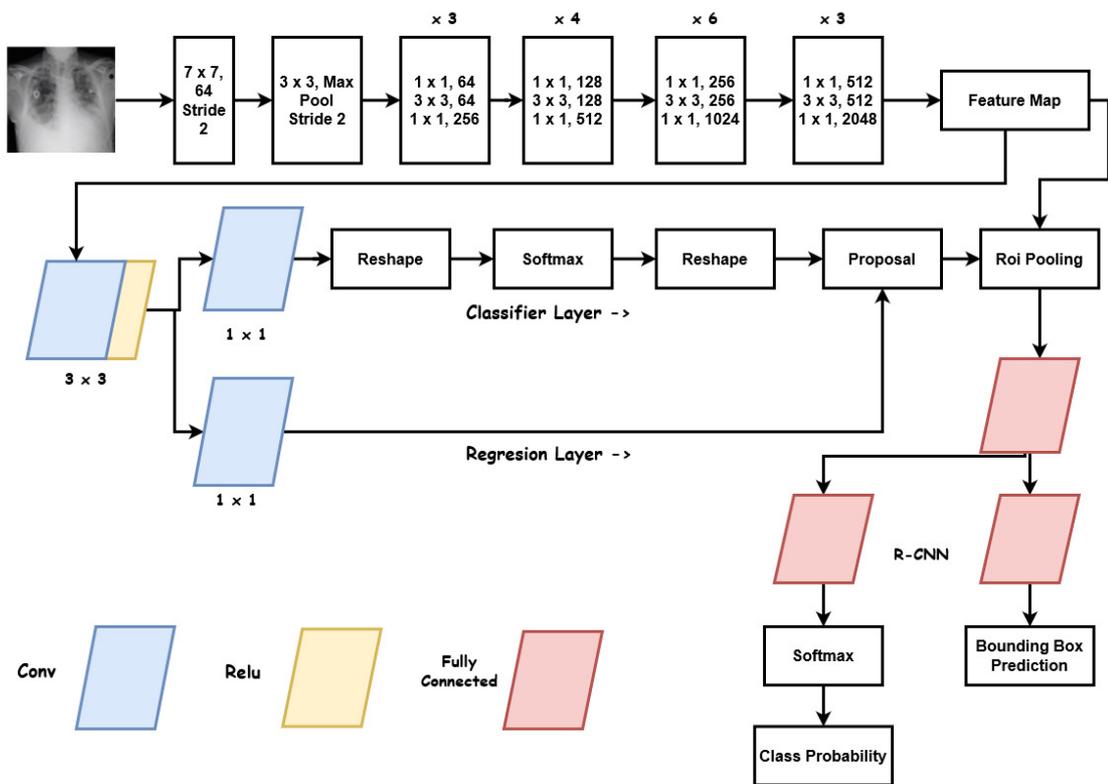
*Training* data merupakan proses implementasikan seluruh desain ke dalam kode program. Kode program tersebut masih berupa modul-modul yang diimplementasikan menjadi kesatuan yang lengkap. Pada tahap ini *dataset* dilatih sehingga menghasilkan sebuah model yang bisa digunakan untuk mendeteksi objek.

#### d. Testing Data

Dilakukan pengujian untuk menganalisis apakah model yang telah dibuat sesuai dengan kriteria dan semua fungsi pada sistem terdapat *error* atau tidak. Untuk mengukur kedua arsitektur yang diterapkan, akan ditampilkan nilai akurasi pendeteksian terhadap data pengujian dari citra Pneumonia .



Gambar 5. Arsitektur VGG16



Gambar 6. Arsitektur ResNet50

## 2.2. Implementasi Model dari Citra Pneumonia COVID-19

Dilakukan implementasi model terhadap citra Pneumonia COVID-19. Pengujian dilakukan untuk mengukur seberapa baik akurasi yang dihasilkan dari model terhadap citra. Untuk mengetahui tingkat akurasi dari VGG16 dan ResNet50, penelitian ini menggunakan metode mAP (*Mean Average Precision*) dari data citra *rontgen* Pneumonia COVID-19.

## 3. HASIL DAN PEMBAHASAN

Pada penelitian yang dilakukan *dataset* yang digunakan untuk pelatihan dan pengujian berupa citra paru-paru Pneumonia yang dapat diunduh pada *website* Kaggle. Data yang digunakan adalah data RSNA (*Radiological Society of North America*) yang terdiri atas citra Pneumonia dan lokasi *ground truth box*. Sedangkan untuk pengujian citra Pneumonia COVID-19 juga diunduh pada *website* Kaggle.

### 3.1. Preprocessing

*Dataset* dari *website* Kaggle terdiri atas data pasien Pneumonia beserta gambar X-ray Pneumonia. *Dataset* tersebut terdiri atas data *dicom* (*Digital Imaging and Communication in Medicine*), data *ground truth box*, dan juga data pasien. [Tabel 1](#) dan [Tabel 2](#) menunjukkan hasil pembacaan data *dicom* yang telah diolah agar dapat dibaca dan diinterpretasi oleh *library* Python-Panda. Kemudian, [Tabel 1](#) dan [Tabel 2](#) digabungkan menjadi [Tabel 3](#) agar pengambilan dan penyajian data menjadi lebih rapi.

Tabel 1. Data Pasien dan *Class*

No	patientId	Class
0	0004cfab-14fd-4e49-80ba-63a80b6bddd6	<i>No Lung Opacity / Not Normal</i>
1	00313ee0-9eaa-42f4-b0ab-c148ed3241cd	<i>No Lung Opacity / Not Normal</i>
2	00322d4d-1c29-4943-afc9-b6754be640eb	<i>No Lung Opacity / Not Normal</i>
3	003d8fa0-6bf1-40ed-b54c-ac657f8495c5	<i>No Lung Opacity / Not Normal</i>
4	00436515-870c-4b36-a041-de91049b9ab4	<i>Lung Opacity</i>
...	...	...
30221	c1ec14ff-f6d7-4b38-b0cb-fe07041cbdc8	<i>Lung Opacity</i>
30222	c1ec14ff-f6d7-4b38-b0cb-fe07041cbdc8	<i>Lung Opacity</i>
30223	c1edf42b-5958-47ff-a1e7-4f23d99583ba	<i>Normal</i>
30224	c1f6b555-2eb1-4231-98f6-50a963976431	<i>Normal</i>
30225	c1f7889a-9ea9-4acb-b64c-b737c929599a	<i>Lung Opacity</i>

Tabel 2. Data Pasien Beserta Letak *Ground Truth Box* pada Citra

No	patientId	x	y	width	height	Target
0	0004cfab-14fd-4e49-80ba-63a80b6bddd6	NaN	NaN	NaN	NaN	0
1	00313ee0-9eaa-42f4-b0ab-c148ed3241cd	NaN	NaN	NaN	NaN	0
2	00322d4d-1c29-4943-afc9-b6754be640eb	NaN	NaN	NaN	NaN	0
3	003d8fa0-6bf1-40ed-b54c-ac657f8495c5	NaN	NaN	NaN	NaN	0
4	00436515-870c-4b36-a041-de91049b9ab4	256.0	152.0	213.0	379.0	1
...	...	...	...	...	...	...
30221	c1ec14ff-f6d7-4b38-b0cb-fe07041cbdc8	609.0	464.0	240.0	284.0	1
30222	c1ec14ff-f6d7-4b38-b0cb-fe07041cbdc8	185.0	298.0	228.0	379.0	1
30223	c1edf42b-5958-47ff-a1e7-4f23d99583ba	NaN	NaN	NaN	NaN	0
30224	c1f6b555-2eb1-4231-98f6-50a963976431	NaN	NaN	NaN	NaN	0
30225	c1f7889a-9ea9-4acb-b64c-b737c929599a	570.0	393.0	261.0	345.0	1

Keterangan:

- Kolom "No" : nomor data
- Kolom "patientID" : nama data
- Kolom "class" : kategori data
- Kolom "x" dan "y" : kordinat boundingbox dari groundtruth
- Kolom "width" : lebar label boundingbox
- Kolom "height" : panjang label boundingbox
- Kolom "Target" : kategori data

Tabel 3. Tabel Setelah Digabungkan

No	patientId	x	x-max	y	y-max	class
4	Dataset/Open Images Dataset v4 (Bounding Boxes)/train/00436515-870c-4b36-a041-de91049b9ab4.jpg	264.0	477.0	152.0	531.0	Lung Opacity
5	Dataset/Open Images Dataset v4 (Bounding Boxes)/train/00436515-870c-4b36-a041-de91049b9ab4.jpg	562.0	818.0	152.0	605.0	Lung Opacity
8	Dataset/Open Images Dataset v4 (Bounding Boxes)/train/00704310-78a8-4b38-8475-49f4573b2dbb.jpg	323.0	483.0	577.0	681.0	Lung Opacity
9	Dataset/Open Images Dataset v4 (Bounding Boxes)/train/00704310-78a8-4b38-8475-49f4573b2dbb.jpg	695.0	857.0	575.0	712.0	Lung Opacity
14	Dataset/Open Images Dataset v4 (Bounding Boxes)/train/00aecb01-a116-45a2-956c-08d2fa55433f.jpg	288.0	382.0	322.0	457.0	Lung Opacity

### 3.2. Konfigurasi Model

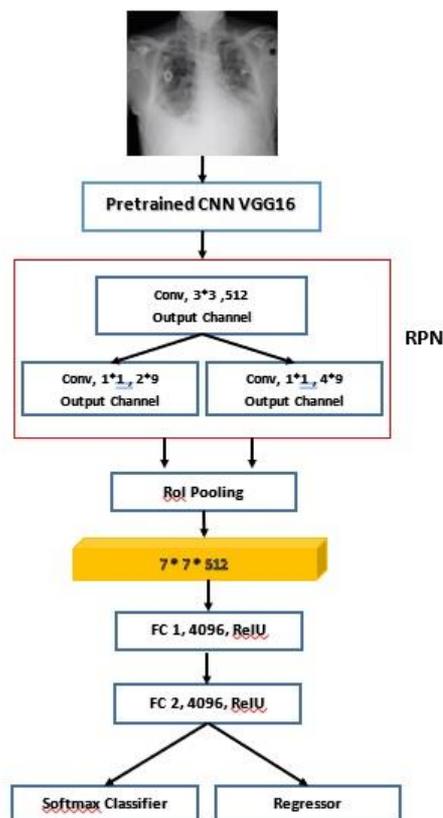
#### 1) Input Data dan Menyimpan Model Pelatihan

Tahap pertama adalah mengimplementasikan lokasi *dataset* dan juga tempat untuk menyimpan hasil dari pelatihan *program* yang telah dibuat. *Dataset* berupa citra *rontgen* paru paru sebanyak 30225 gambar citra X-ray. Dari 30225 citra setelah di *preprocessing* dan digabungkan dihasilkan 9555 gambar beserta *ground truth box* yang bisa untuk di latih.

Persentase pembagian *dataset* yang digunakan untuk pelatihan sebesar 80% dan untuk pengujian sebesar 20%. Untuk *dataset* pelatihan yang digunakan adalah 7644 data, dan untuk pengujian sebanyak 1911 data. Data-data ini kemudian disimpan dalam format *h5*.

#### 2) Implementasi Arsitektur VGG16 *Faster R-CNN*

VGG-16 memanfaatkan *convolutional layer* dengan spesifikasi *convolutional filter* yang kecil (3x3). Dengan ukuran *convolutional filter* sebesar 3x3, kedalaman *neural network* dapat ditambah dengan lebih banyak lagi *convolutional layer* dan hasilnya model CNN menjadi lebih akurat dari pada model-model CNN yang lainnya. Model VGG16 mempunyai 16 *layer* yang terdiri dari 13 *convolutional layer* dan 3 *fully connected layer*. Gambar 7 menunjukkan arsitektur VGG16 pada penelitian ini.



Gambar 7. Implementasi Arsitektur VGG16 pada *Faster R-CNN* untuk mendeteksi Pneumonia

Setelah mengimplementasikan arsitektur VGG16, *feature map* diambil pada *layer* terakhir VGG16, yang digunakan sebagai *propose region* dan *classifier*. Untuk presentase IoU yang digunakan adalah 70%, jika objek tersebut mempunyai IoU diatas 70% maka objek tersebut dikategorikan sebagai objek yang *valid*. Sedangkan jika IoU di bawah 70% maka objek tersebut dikategorikan sebagai objek *non-valid* dan akan menjadi *background* saja.

*Featuremap* dari lapisan konvolusi dasar diambil dan diterapkan pada lapisan *base layer*  $3 \times 3 \times 512$  *channels convolution layer*. Pertahankan *padding* 'same' untuk mempertahankan ukuran *feature map size*. Langkah selanjutnya mengambil (1,1) *convolution layer* yang digunakan untuk *classifier* dan *regresor*.

Regressor digunakan untuk membuat *bounding box* pada gambar. Sedangkan *classifier* berfungsi sebagai *Fast R-CNN*, digabungkan dengan RPN maka akan dihasilkan algoritma *Faster R-CNN*. *Classifier* mengambil data dari *feature map* yang kemudian digabungkan dengan *proposal network* dari RPN sebagai objek yang akan di deteksi.

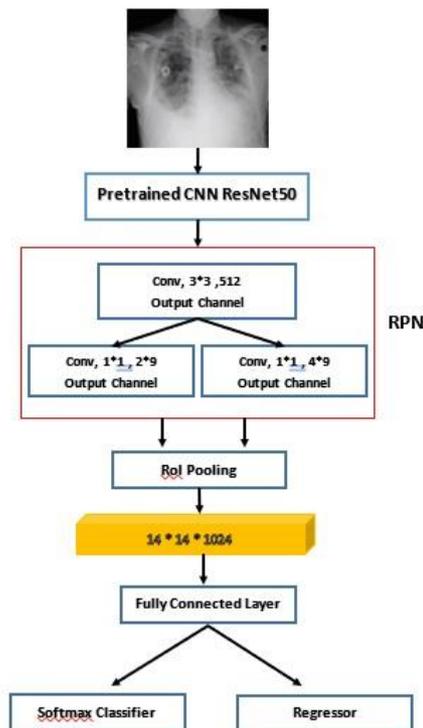
### 3) Implementasi Arsitektur ResNet50 *Faster R-CNN*

*Deep Residual Network* atau biasa disebut dengan ResNet merupakan salah satu arsitektur dari CNN yang dibangun untuk mengatasi permasalahan pada pelatihan *deep learning*, karena pelatihan *deep learning* pada umumnya memakan cukup banyak waktu dan terbatas pada jumlah lapisan tertentu. Solusi permasalahan yang diusulkan oleh ResNet adalah dengan menerapkan *skip connection* atau *shortcut*. Arsitektur ResNet50 pada *object detection* terdiri dari 50 *layer* yang berbeda. *Residual block* pada ResNet dapat dilakukan apabila dimensi data masukan sama dengan data keluaran. Dua lapisan awal dari arsitektur ResNet melakukan *convolution*  $7 \times 7$  dan *max pooling* berukuran  $3 \times 3$  dengan jumlah *stride* 2.

Gambar 8 merupakan alur proses pelatihan *Faster R-CNN* menggunakan ResNet50. Dengan menggunakan *transfer learning/fine tuning* membuat parameter yang dihasilkan lebih sedikit daripada yang dilakukan tanpa *transer learning*. Fungsi *transfer learning* adalah mengambil *knowledge* parameter pada model yang kemudian program akan mengganti jumlah *layer* yang dipakai, mengganti nilai *learning rate*, mengganti ukuran *filter*, dan lain sebagainya.

Dengan menggunakan *transfer learning* akan mempercepat proses pelatihan data daripada menggunakan ResNet murni. Presentase IoU yang digunakan adalah 70%, jika objek tersebut mempunyai IoU diatas 70% maka objek tersebut dikategorikan sebagai objek yang *valid*. Sedangkan jika IoU di bawah 70% maka objek tersebut dikategorikan sebagai objek *non-valid* dan akan menjadi *background* saja.

*Featuremap* diambil dari lapisan konvolusi dasar ke lapisan *base layer*  $3 \times 3 \times 512$  *channels convolution layer*. Regressor digunakan untuk membuat *bounding box* pada gambar. Sedangkan *classifier* berfungsi sebagai *Fast R-CNN*, digabungkan dengan RPN maka akan dihasilkan algoritma *Faster R-CNN*. *Classifier* mengambil data dari *feature map* yang kemudian digabungkan dengan *proposal network* dari RPN sebagai objek yang akan di deteksi. Arsitektur yang digunakan menggunakan ResNet50 dengan *pretrained model* yang berbeda dengan RPN. RPN hanya membuat *region proposal*, kemudian *Fast R-CNN* sebagai *classifier*.



Gambar 8. Implementasi Arsitektur ResNet50 pada *Faster R-CNN* untuk mendeteksi Pneumonia

#### 4) *Region Of Interest Pooling (ROI Pooling)*

*Region of Interest (ROI)* diimplementasikan sebagai teknik segmentasi dari proses pengolahan citra agar sistem mampu mengolah citra yang mengandung informasi data citra yang dikehendaki. Pada proses ini *ROI pooling* digunakan untuk menggabungkan *region of interest* dari RPN dengan *feature map*. Dari proses tersebut maka dihasilkan *feature map* dengan *bounding box* dan *classifier*.

Dari proses tersebut dihasilkan *ROI pooling layer* untuk *layer convolusi*, dengan *pool size*  $7 \times 7 \times 512$  untuk VGG16 dan  $14 \times 14 \times 1028$  untuk ResNet50. Dari proses *pooling* akan terbentuk sebuah kesatuan dimana program dapat menghasilkan deteksi objek pada gambar. Setelah melewati *roi pooling* maka *feature map* akan di sederhanakan dengan *fully connected layer*.

### 3.3. Training Data Model

*Training data* dilakukan menggunakan *dataset* dari *website Kaggle* yang telah di pisah 80% untuk pelatihan data dari 9555 data citra. Pada saat melatih data tersebut, penelitian menggunakan 20000 *batch layer*. Hasil dari pelatihan model dipresentasikan pada Tabel 4. *Epoch* menunjukkan urutan pada proses perulangan, akurasi menunjukkan ketepatan sistem dalam kinerjanya mengklasifikasi Pneumonia, dan *loss* menunjukkan evaluasi prediksi model.

Dari proses pelatihan tersebut dihasilkan model untuk deteksi objek menggunakan arsitektur VGG16 dan juga ResNet50. Untuk waktu pelatihan arsitektur VGG16 membutuhkan waktu kurang lebih 35 jam, sedangkan ResNet50 membutuhkan waktu pelatihan kurang lebih 13 jam. Dari hasil pelatihan dihasilkan

akurasi untuk arsitektur VGG16 sebesar 70,3 % dan ResNet50 sebesar 71,2%. Kemudian untuk *total loss* dari VGG16 sebesar 2.712, dan ResNet50 sebesar 2.66.

Tabel 4. Hasil Pelatihan Model

Epoch	Akurasi (VGG16)	Akurasi (ResNet50)	Loss (VGG16)	Loss (ResNet50)
1	0.683	0.638	5.630	5.201
2	0.658	0.627	4.453	3.736
3	0.674	0.654	4.114	3.915
4	0.678	0.656	3.722	3.506
5	0.683	0.689	3.583	3.237
...	...	...	...	...
16	0.708	0.705	2.796	2.794
17	0.720	0.709	2.804	2.637
18	0.715	0.708	2.829	2.747
19	0.709	0.718	2.913	2.63
20	0.703	0.712	2.712	2.66

### 3.4. Testing Data Model

Pada pengujian yang dilakukan penguji menggunakan mAP (*mean verage precision*) yang diperoleh dari rata-rata *average precision*. Presisi (1) digunakan untuk mengukur seberapa akurat memprediksi sebuah objek. Rata-rata presisi menghitung nilai presisi rata-rata untuk nilai *recall* lebih dari 0 hingga 1. Sedangkan, *recall* (2) mengukur seberapa baik mesin menemukan semua objek positif. Jadi, pada prosesi ini adalah menghitung nilai presisi objek terhadap *bounding box regressor* berdasarkan RPN. Nilai TP, FP, dan FN diperoleh dari perhitungan standar pada *confusion matrix* secara umum. *Mean Average Precision* (3) sendiri merupakan metode pengujian yang diperoleh dari rata rata presisi dari sebuah deteksi objek.

Pengujian yang dilakukan menggunakan 1911 gambar yang terdapat *ground truth box* yang selanjutnya dihitung rata-rata presisi pada objek tersebut. Data pengujian diambil dari proses *split* data yaitu sebanyak 20%. Dari proses pelatihan 20000 batch yang dilakukan menggunakan algoritma *Faster R-CNN* membuktikan bahwa arsitektur VGG16 mempunyai rata-rata presisi yang lebih tinggi dari pada ResNet50. [Tabel 5](#) menunjukkan hasil pengujian mAP.

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$mAP = \frac{1}{n} \sum_{Recall}^n Precision(Recall_i) \tag{3}$$

Tabel 5. Hasil Pengujian Map (*Mean Average Precision*)

Arsitektur	mAP
VGG16	14,1 %
ResNet50	11,9 %

### 3.5. Implementasi Deteksi Pneumonia COVID-19 Arsitektur VGG16 dan ResNet50

Pengujian dilakukan dengan menggunakan gambar *rontgen* paru-paru Pneumonia sebanyak 500 gambar citra Pneumonia COVID-19 dari *website* Kaggle. Dari pengujian yang dilakukan arsitektur VGG16 mempunyai akurasi tertinggi yaitu sebesar 86% sedangkan ResNet sebesar 84%. Tabel 6 menyajikan hasil implementasi menggunakan arsitektur VGG16 dengan menggunakan 500 citra *rontgen* paru-paru Pneumonia COVID-19. Nilai akurasi dihitung dengan menggunakan (4).

$$Akurasi = \frac{Jumlah\ prediksi\ benar}{Jumlah\ semua\ data} \times 100 \quad (4)$$

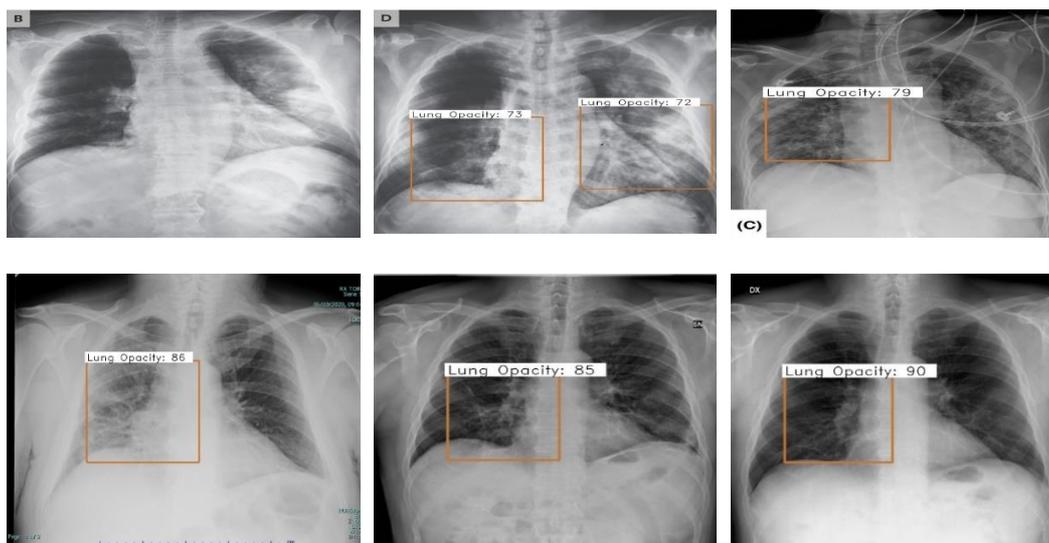
Tabel 6. Hasil Pengujian Menggunakan 500 Data Citra Pneumonia Covid-19

Arsitektur	Prediksi Benar	Prediksi Salah	Akurasi
VGG16	430	70	86%
ResNet50	420	80	84%

Dari kedua model yang dibuat keduanya menunjukkan tingkat akurasi yang tidak jauh berbeda. Dari segi akurasi arsitektur VGG16 mempunyai nilai akurasi tertinggi yaitu sebesar 86% sedangkan ResNet50 mempunyai nilai akurasi sebesar 84%. Dari hasil pengujian yang dilakukan arsitektur VGG16 lebih baik dalam mendeteksi adanya Pneumonia COVID-19 pada paru-paru. Tabel 5 menunjukkan hasil pengujian dengan 500 data citra Pneumonia COVID-19.

Penelitian yang dikembangkan hanya dapat memprediksi adanya Pneumonia pada paru-paru, tidak bisa memprediksi gambar paru-paru normal. Dalam penerapan algoritma *Faster R-CNN* hanya dapat memprediksi adanya objek pada gambar. Jika dalam pengujian di uji gambar paru-paru normal maka program tidak bisa memprediksi bahwa gambar tersebut gambar paru-paru normal.

Penelitian ini menggunakan presentasi IoU untuk gambar sebesar 70%. IoU merupakan nilai berdasarkan statistik kesamaan dan keragaman set sampel yang bertujuan untuk mengevaluasi area tumpang tindih (area yang beririsan) antara dua *bounding box*, yaitu *bounding box* hasil prediksi dan *bounding box ground truth* (kebenaran). Dari proses tersebut dihasilkan sebuah gambar prediksi dengan skor IoU lebih dari 70%, jika gambar tersebut prediksi *bounding box* kurang dari 70% maka gambar tersebut tidak valid atau kurang adanya kesamaan antara model yang telah di latih dan data pengujian. Gambar 9 menunjukkan hasil implementasi model terhadap citra toraks Pneumonia COVID-19.



Gambar 9. Contoh Hasil Pengujian Beserta Bounding Box Pada Citra Pneumonia COVID-19

Hasil prediksi yang dihasilkan pada Gambar 9 kiri atas tidak menghasilkan prediksi *bounding box*. Gambar kedua, ketiga, sampai keenam memiliki prediksi *bounding box*, dengan akurasi yang berbeda beda. “Lung Opacity” menunjukkan deskripsi tingkat akurasi dari setiap pendeteksian pada citra toraks Pneumonia COVID-19. Dari hasil pengujian yang dilakukan, setiap gambar tidak memiliki karakteristik adanya bentuk dan warna yang spesifik dari setiap hasil prediksi. Jadi, dari hasil implementasi yang dihasilkan tidak terpengaruh oleh bentuk dan warna dalam proses klasifikasi.

#### 4. KESIMPULAN

Penelitian yang dikembangkan dapat memprediksi adanya Pneumonia pada paru-paru, tidak bisa memprediksi citra paru-paru normal. Model berhasil dibuat tanpa ada eror dan dapat mendeteksi objek Pneumonia pada citra dengan baik. Dalam penerapan algoritma *Faster R-CNN* hanya dapat memprediksi adanya objek pada citra. Jika dalam pengujian di uji dengan citra paru-paru normal maka program tidak bisa memprediksi bahwa citra tersebut citra paru-paru normal.

Secara garis besar perancangan model pada penelitian ini dibagi menjadi 4 tahap utama, yaitu *preprocessing*, *implementation*, *training data*, dan *testing data*. Algoritma yang digunakan menggunakan metode *Faster R-CNN* dengan dua arsitektur CNN yang berbeda yaitu VGG16 dan ResNet50. Dari pelatihan yang dilakukan arsitektur VGG16 mempunyai nilai akurasi sebesar 70,3% dengan *loss* sebesar 2,712, sedangkan ResNet mempunyai nilai akurasi 71,2% dengan *loss* sebesar 2,66. Dari hasil pelatihan yang dilakukan, dilakukan *transfer learning* agar data pelatihan dapat diimplementasikan pada data 500 citra *X-ray* paru-paru COVID-19.

Dari pengujian yang dihasilkan model VGG16 mempunyai mAP sebesar 14,1% sedangkan ResNet mempunyai nilai presisi mAP sebesar 11,9%. Kemudian untuk pengujian menggunakan 500 data *X-ray* paru-paru Pneumonia COVID-19, arsitektur VGG16 mempunyai nilai akurasi tertinggi yaitu sebesar 86% sedangkan

ResNet50 mempunyai nilai akurasi sebesar 84%. Hasil pengujian yang dilakukan arsitektur VGG16 lebih baik dalam mendeteksi adanya Pneumonia COVID-19 pada paru-paru hal tersebut ditunjukkan dengan nilai mAP yang tinggi pada pengujian data COVID-19 dibandingkan dengan ResNet50.

## REFERENSI

- [1] D. Mopuri, "a Review Article on Pneumonia," *World J Pharm Res*, vol. 8, no. 8, pp. 134–153, 2017, doi: 10.20959/wjpr201710-8985.
- [2] J. M. Galván and O. Rajas, "Review of Non-bacterial Infections in Respiratory Medicine : Viral Pneumonia &," vol. 51, no. 11, pp. 590–597, 2015.
- [3] L. Gattinoni *et al.*, "COVID-19 Pneumonia: pathophysiology and management," *European Respiratory Review*, no. June, pp. 1–13, 2021, doi: 10.1183/16000617.0138-2021.
- [4] W. H. Organization, "Middle East respiratory syndrome coronavirus (MERS-CoV)," *WHO WEB*, 2019. [https://www.who.int/news-room/fact-sheets/detail/middle-east-respiratory-syndrome-coronavirus-\(mers-cov\)](https://www.who.int/news-room/fact-sheets/detail/middle-east-respiratory-syndrome-coronavirus-(mers-cov)) (accessed Feb. 28, 2022).
- [5] B. Baskara, "Rangkaian Peristiwa Pertama Covid-19," *Harian Kompas*, 2020. <https://www.kompas.id/baca/riset/2020/04/18/rangkaian-peristiwa-pertama-covid-19> (accessed Dec. 29, 2021).
- [6] G. S. Putri, "WHO Resmi Sebut Virus Corona Covid-19 sebagai Pandemi Global," *Harian Kompas*, 2020. <https://www.kompas.com/sains/read/2020/03/12/083129823/who-resmi-sebut-virus-corona-covid-19-sebagai-pandemi-global?page=all> (accessed Dec. 29, 2021).
- [7] Team, "COVID-19 situation update worldwide, as of week 7, updated 24 February 2022," *European Centre for Disease Prevention and Control*, 2022. <https://www.ecdc.europa.eu/en/geographical-distribution-2019-ncov-cases> (accessed Feb. 28, 2022).
- [8] D. Mopuri, "a Review Article on Pneumonia ," *World J Pharm Res*, vol. 8, no. 8, pp. 134–153, 2017, doi: 10.20959/wjpr201710-8985.
- [9] Y. T. Y. Steve, "Penjelasan tentang Pneumonia & Novel Coronavirus COVID-19," *mountelizabeth*, 2020. <https://www.mountelizabeth.com.sg/id/healthplus/article/Pneumonia-and-coronavirus> (accessed Mar. 01, 2022).
- [10] N. Yudistira, A. W. Widodo, and B. Rahayudi, "Deteksi Covid-19 pada Citra Sinar-X Dada Menggunakan Deep Learning yang Efisien," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 6, p. 1289, 2020, doi: 10.25126/jtiik.2020763651.
- [11] R. Yamashita, M. Nishio, R. K. G. Do, and K. Togashi, "Convolutional neural networks: an overview and application in radiology," *Insights Imaging*, vol. 9, no. 4, pp. 611–629, 2018, doi: 10.1007/s13244-018-0639-9.
- [12] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Trans Pattern Anal Mach Intell*, vol. 39, no. 6, pp. 1137–1149, 2017, doi: 10.1109/TPAMI.2016.2577031.
- [13] N. Nasrullah, J. Sang, M. S. Alam, M. Mateen, B. Cai, and H. Hu, "Automated Lung Nodule Detection and Classification Using Deep Learning Combined with Multiple Strategies.," *Sensors (Basel)*, vol. 19, no. 17, Aug. 2019, doi: 10.3390/s19173722.
- [14] W. Swastika, "Studi Awal Deteksi Covid-19 Menggunakan Citra Ct Berbasis Deep Preliminary Study of Covid-19 Detection Using Ct Image Based on," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 7, no. 3, pp. 629–634, 2020, doi: 10.25126/jtiik.202073399.
- [15] R. Girshick, "Fast R-CNN," *Microsoft Research*, Apr. 2015.
- [16] Deval Shah, "Mean Average Precision (mAP) Explained: Everything You Need to Know," *v7labs*, Oct. 2022. <https://www.v7labs.com/blog/mean-average-precision#> (accessed Nov. 06, 2022).
- [17] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in *Advances in Neural Information Processing Systems*, 2015, vol. 28. [Online]. Available:

<https://proceedings.neurips.cc/paper/2015/file/14bfa6bb14875e45bba028a21ed38046-Paper.pdf>

- [18] M. Afif, A. Fawwaz, K. N. Ramadhani, and F. Sthevanie, "Klasifikasi Ras pada Kucing menggunakan Algoritma Convolutional Neural Network(CNN)," *Jurnal Tugas Akhir Fakultas Informatika*, vol. 8, no. 1, pp. 715–730, 2020.
- [19] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2016-Decem, pp. 779–788, 2016, doi: 10.1109/CVPR.2016.91.