

## BAB 4

### IMPELENTASI DAN PENGUJI

#### 4.1 Implementasi

Tahap implementasi merupakan tahap pembangunan perangkat lunak pengenalan pola citra sandi rumput sesuai dengan 0analisis dan perancangan yang sudah dilakukan pada tahap sebelumnya. Tahap ini merupakan tahap di mana perangkat lunak siap untuk digunakan, yang terdiri dari pembahasan mengenai lingkungan implementasi, baik lingkungan perangkat keras maupun perangkat lunak. Berikut merupakan hasil implementasi yang dilakukan.

##### 4.1.1 Implementasi Perangkat Keras

Spesifikasi perangkat keras yang digunakan dalam pembangunan perangkat lunak pengenalan Anomali *Pneumonia* dengan R-CNN dapat dilihat pada Tabel 4.1.

**Tabel 4. 1 Spesifikasi Perangkat Keras**

No	Perangkat	Spesifikasi
1	Processor	Intel Core i7
2	Memory	16GB DDR4
3	Harddisk	SSD EVO 970 250GB
4	VGA	GTX1050 4GB

##### 4.1.2 Implementasi Perangkat Lunak

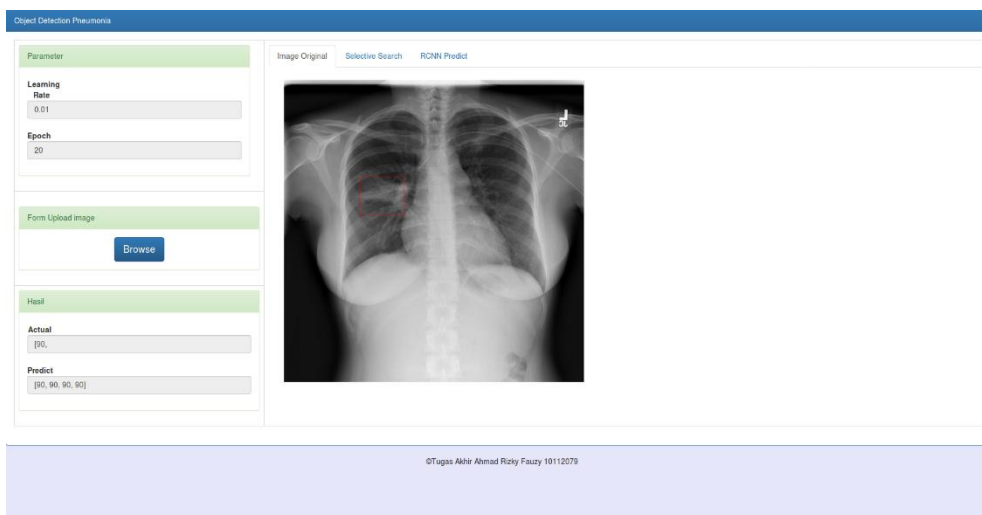
Adapun spesifikasi perangkat lunak yang digunakan dalam pembangunan perangkat lunak pengenalan Anomali *Pneumonia* dengan R-CNN dapat dilihat pada dapat dilihat pada Tabel 4.2.

**Tabel 4. 2 Spesifikasi Perangkat Lunak**

No	Perangkat	Spesifikasi
1	Sistem Operasi	Windows 10
2	Tools Pemrograman	Pycharm 2019.1.3
3	Bahasa Pemrograman	Python 3.6.8

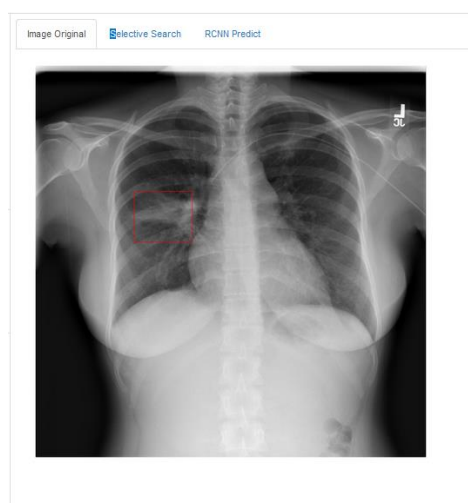
### 4.1.3 Implementasi Antarmuka

Berikut merupakan implementasi dari tampilan antarmuka pada perangkat lunak yang dibangun terdapat satu Tampilan Antarmuka Menu Pengujian, ditunjukkan pada Gambar 4.1.

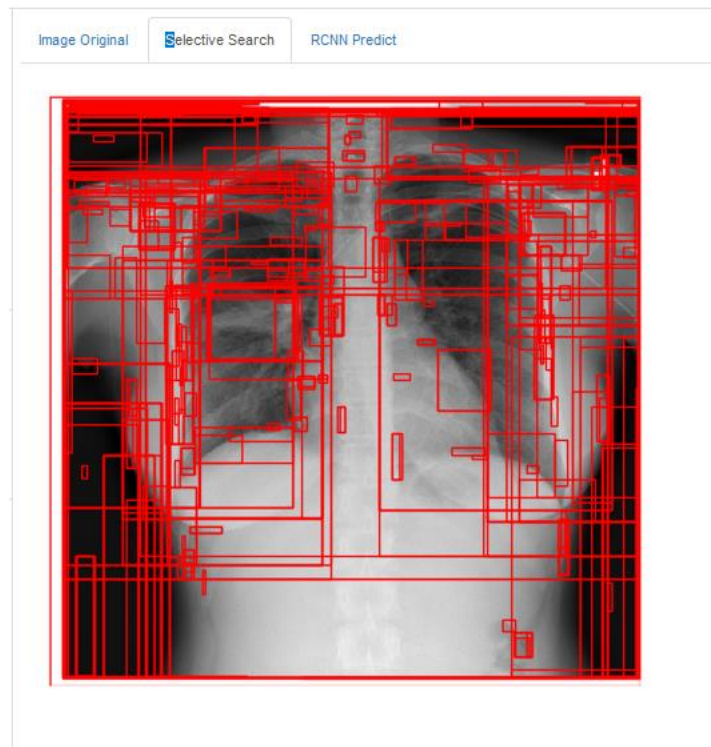


**Gambar 4. 1 Tampilan Antarmuka Pengujian**

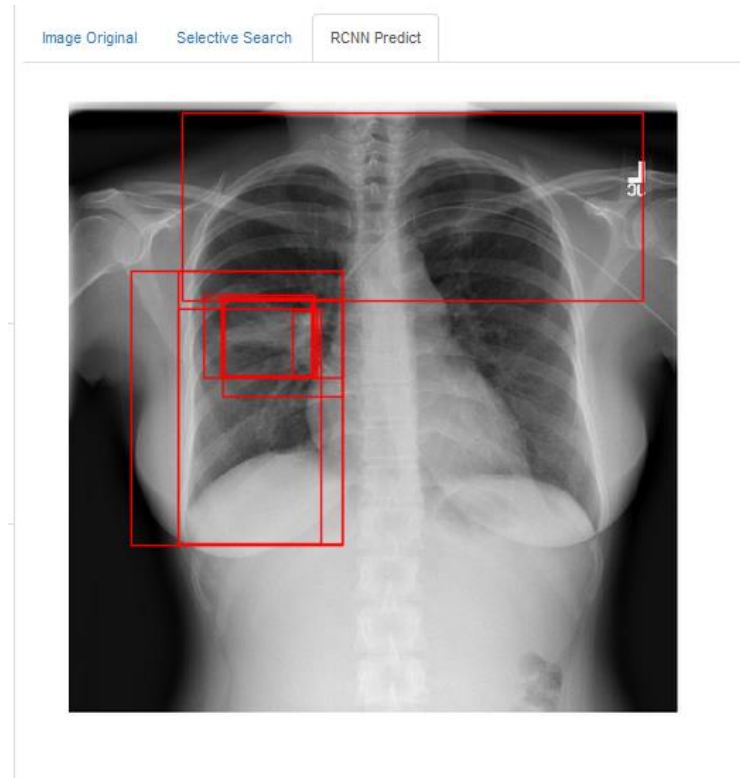
Pada pengujian, terdapat satu tombol yaitu tombol browse. Tombol browse berfungsi untuk memasukan cita yang akan masuk ke proses pengujian. Adapun hasil dari pengujian terdiri dari tiga menu adalah menu Image Original, Selective Search dan RCNN Predict ditunjukkan pada Gambar 4.2, Gambar 4.3, dan Gambar 4.4.



**Gambar 4. 2 Antarmuka Image Original**



**Gambar 4. 3 Antarmuka Selective Search**



**Gambar 4. 4 Antarmuka RCNN Predict**

## 4.2 Pengujian

Pengujian merupakan tahap yang bertujuan untuk menemukan kesalahan-kesalahan atau kekurangan yang ada pada sistem sudah dibangun sebelum digunakan oleh pengguna. Tahap pengujian yang dilakukan meliputi pengujian white box, pengujian black box dan pengujian akurasi.

### 4.2.1 Pengujian Black Box

Pengujian fungsionalitas dilakukan dengan menggunakan metode pengujian black box. Pengujian black box memungkinkan pembangunan perangkat lunak mendapatkan sejumlah kondisi data masukan yang sepenuhnya untuk suatu program. Pengujian fungsionalitas terdiri dari rencana pengujian serta kasus dan hasil pengujian.

#### 4.2.1.1 Rencana Pengujian

Adapun rencana pengujian black box yang akan dilakukan dapat dilihat pada Tabel 4.3.

**Tabel 4. 3 Rencana Pengujian *Black Box***

NO	Item Uji	Butir Uji	Jenis Pengujian
1	Preprocessing	Memasukan file citra thorax	<i>Black Box</i>
		Melakukan proses pengolahan citra	
2	Pelatihan	Melakukan Proses Pelatihan	<i>Black Box</i>
3	Pengujian	Melakukan Proses Pengujian	<i>Black Box</i>

#### 4.2.1.2 Kasus dan Hasil Pengujian

Tahap pengujian dilakukan dengan cara menguji setiap fungsi yang ada pada perangkat lunak untuk semua kemungkinan yang terjadi berdasarkan rencana pengujian. Berikut merupakan poin-poin pengujian yang dilakukan.

##### 1. Pengujian memasukkan file citra thorax

Berikut merupakan pengujian untuk memasukkan file citra thorax yang ditunjukkan pada Tabel 4.4.

**Tabel 4. 4 Pengujian Memasukkan File Citra thorax**

Kasus dan Hasil Pengujian (Data Benar)				
Data Uji	Data Masukan	Harapan	Keluaran	Kesimpulan
Citra	Image_101.png	Citra tampil didalam aplikasi	Citra dapat ditampilkan didalam aplikasi	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Kasus dan Hasil Pengujian (Data Salah)				
Data Uji	Data Masukan	Harapan	Keluaran	Kesimpulan
Dokumen	Document1.docx	Dokumen tidak tampil di dalam aplikasi	Dokumen tidak dapat ditampilkan di dalam aplikasi	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak

2. Pengujian proses pengolahan citra

Berikut merupakan pengujian proses pengolahan citra yang ditunjukkan pada Tabel 4.5.

**Tabel 4. 5 Pengujian Proses Pengolahan Citra**

Kasus dan Hasil Pengujian (Data Benar)				
Data Uji	Data Masukan	Harapan	Keluaran	Kesimpulan
Citra, Nama Kelas	Citra : Image_101.png Nama Kelas : <i>Pneumonia</i>	Citra diproses hingga tahapan segmentasi	Citra dapat diproses hingga tahapan segmentasi	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak
Kasus dan Hasil Pengujian (Data Salah)				
Data Uji	Data Masukan	Harapan	Keluaran	Kesimpulan
Citar, Nama Kelas	Citra : <i>Null</i> Nama Kelas : <i>Null</i>	Citra tidak diproses	Citra tidak dapat diproses	<input checked="" type="checkbox"/> Diterima <input type="checkbox"/> Ditolak

3. Pengujian proses pelatihan

Berikut merupakan pengujian proses pelatihan yang ditunjukkan pada Tabel 4.6.

**Tabel 4. 6 Pengujian Proses Pelatihan**

Kasus dan Hasil Pengujian (Data Benar)				
Data Uji	Data Masukan	Harapan	Keluaran	Kesimpulan
Kumpulan kelas citra, Learning Rate, Epoch	Kumpulan Kelas Citra Learning Rate : 0.0001 Epoch : 100	Kumpulan kelas citra dilatih “Pelatihan Selesai!”	Kumpulan kelas Citra dapat dilatih	[√] Diterima [ ] Ditolak
Kasus dan Hasil Pengujian (Data Salah)				
Data Uji	Data Masukan	Harapan	Keluaran	Kesimpulan
Kumpulan kelas citra,	Kumpulan Kelas Citra : <i>Null</i> Learning Rate : <i>Null</i> Epoch : <i>Null</i>	Kumpulan kelas citra tidak dilatih	Kumpulan kelas citra tidak dapat dilatih	[√] Diterima [ ] Ditolak

## 4. Pengujian proses pengujian

Berikut merupakan pengujian proses pengujian yang ditunjukkan pada Tabel 4.7.

**Tabel 4. 7 Pengujian Proses Pengujian**

Kasus dan Hasil Pengujian (Data Benar)				
Data Uji	Data Masukan	Harapan	Keluaran	Kesimpulan
Kumpulan Citra uji yang sudah tersegmentasi	Kumpulan Citra uji	Kumpulan citra terklasifikasi	Kumpulan citra Dapat mengenali objek	[√] Diterima [ ] Ditolak
Kasus dan Hasil Pengujian (Data Salah)				
Data Uji	Data Masukan	Harapan	Keluaran	Kesimpulan

Kumpulan Citra uji yang sudah tersegmentasi	Tidak Ada Kumpulan Citra uji	Proses klasifikasi tidak berjalan	Proses klasifikasi objek tidak dapat berjalan	[√] Diterima [ ] Ditolak
---	------------------------------	-----------------------------------	---	-----------------------------

#### 4.2.1.3 Kesimpulan Pengujian Fungsionalitas Sistem

Berdasarkan hasil pengujian fungsionalitas yang telah dilakukan menggunakan metode Black Box, dapat ditarik kesimpulan bahwa perangkat lunak yang dibangun sudah memiliki fungsionalitas yang baik dan sudah menampilkan output yang diharapkan namun masih memungkinkan terjadi kesalahan.

#### 4.2.2 Pengujian Akurasi

Pada tahap pengujian akurasi akan dilakukan perhitungan nilai akurasi atau kecocokan dari data baru yang masuk ke dalam aplikasi dengan data yang sudah dilatih sebelumnya. Data baru ini merupakan data yang tidak termasuk ke dalam kumpulan data latih. Pengujian dilakukan setelah proses pelatihan data training dilakukan. Adapun metode perhitungan akurasi yang akan dilakukan yaitu dengan menggunakan Mean Average *Precision*. [21] Mean Average *Precision* adalah metrik populer dalam mengukur akurasi detektor objek seperti R-CNN, Faster R-CNN, SSD, dll. *Average Precision* menghitung nilai *average precision* untuk nilai penarikan lebih dari 0 hingga 1. Adapun rumus menghitung Mean Average *Precision* (mAP) menggunakan persamaan sebagai berikut :

$$mAP = \frac{1}{N} \sum_{i=1}^N AP_i$$

keterangan :

N : Jumlah data *AP*

*AP* : *Average precision*

Adapun untuk mendapatkan *AP* di dapatkan dari persamaan berikut :

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

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


$TP$  = *True Positive* (TP), kondisi dimana model mengklasifikasikan data sebagai ya (*TRUE*) dan jawaban aktualnya adalah ya (*TRUE*)

$TN$  = *True Negative* (TN), kondisi dimana model mengklasifikasikan data sebagai tidak (*FALSE*) dan jawaban aktualnya adalah tidak (*FALSE*)

$FP$  = *False Positive* (FP), kondisi dimana model mengklasifikasikan data sebagai ya (*TRUE*) dan jawaban aktualnya adalah tidak (*FALSE*)

$FN$  = *False Negative* (FN), kondisi dimana model mengklasifikasikan data sebagai tidak (*FALSE*) dan jawaban aktualnya adalah ya (*TRUE*)

Adapun untuk melakukan perhitungan  $AP$  untuk deteksi objek maka pencarian prediksi menggunakan persamaan berikut :

$$IoU = \frac{\text{Area of Overlap}}{\text{Area of Union}}$$


Jika nilai  $IoU > 0.5$  maka bernilai *True Positive* sedangkan jika nilai  $IoU < 0.5$  maka bernilai *False Positive*. [22] Adapun untuk mendapatkan nilai *Precision Inter* untuk  $AP_i$  adalah dengan mengambil *Recall* maksimum menggunakan persamaan berikut :

$$AP_i = \max AP_i$$



#### 4.2.2.1 Proses Pengujian Akurasi

Proses pengujian dilakukan terhadap data baru yang akan dikenali oleh sistem. Sebelum ke proses pengenalan, terlebih dahulu dilakukan proses pelatihan terhadap data latih yang sudah ada. Data latih tersebut berjumlah 160 data yang terbagi ke dalam 2 kelas yaitu kelas *Pneumonia* dan Normal. Adapun data uji yang digunakan berjumlah 10 data. Parameter-parameter yang digunakan pada pengujian ini yaitu nilai maksimum epoch dan nilai learning rate. Untuk jumlah maksimum epoch ditetapkan sebesar 500 dengan nilai learning rate yang berubah-ubah. Berikut merupakan proses pengujian yang dilakukan.

##### 1. Pengujian Akurasi Pertama

Pada pengujian pertama dilakukan pelatihan data latih sebanyak 160 data dengan nilai maksimum epoch sebesar 500 dan nilai *learning rate* sebesar 0.0001. Untuk pengujian menggunakan data uji sebanyak 10 data. Adapun hasil pengujian dapat dilihat pada Lampiran E Data uji 1.

Dari Lampiran E Data uji 1 untuk mendapatkan nilai IoU dari *prediksi image* 1 dengan menggunakan persamaan berikut :

$$IoU = \frac{\text{area of Overlap}}{\text{area of Union}}$$

Kordinat irisan pada kotak actual dan predisksi pada *image* 1

a : merupakan kordinat actual dan b : merupakan kordinat prediksi

x1: nilai <i>max</i> dari a(x) dan b(x)	x2 : nilai <i>min</i> dari a( <i>width</i> ) dan b( <i>width</i> )
y1 : nilai <i>max</i> dari a(y) dan b(y)	y2 : nilai <i>min</i> dari a( <i>height</i> ) dan b( <i>height</i> )
x1 : nilai <i>max</i> dari a(258) dan b(448)	x2 : nilai <i>min</i> dari a(152) dan b(41)
<b>448</b>	<b>41</b>
y1 : nilai <i>max</i> dari a(328) dan b(226)	y2 : nilai <i>min</i> dari a(132) dan b(142)
<b>382</b>	<b>132</b>

**Area overlap**

$$\begin{aligned} \text{width} &= (x_2 - x_1) \\ &= (41 - 448) \\ &= \mathbf{-407} \end{aligned}$$

$$\begin{aligned} \text{height} &= (y_2 - y_1) \\ &= (132 - 382) \\ &= \mathbf{-252} \end{aligned}$$

Untuk menangani kordinat di mana TIDAK ada tumpang tindih adalah dengan cara jika nilai dari *width* atau *height* kurang dari 0 maka nilainya manjadi 0 maka nilai *area overlap* sebagai berikut :

$$\begin{aligned} \text{Area overlap} &= \text{Width} \times \text{height} \\ &= 0 \times 0 \\ &= \mathbf{0} \end{aligned}$$

**Area Union**

$$\text{Kordinat a} = (a(\text{width}) - a(x)) \times (a(\text{height}) - a(y))$$

$$\text{Kordinat b} = (b(\text{width}) - b(x)) \times (b(\text{height}) - b(y))$$

$$\text{Area union} = \text{Kordinat a} + \text{Kordinat b} - \text{area\_overlap}$$

$$\begin{aligned} \text{Kordinat a} &= (a(152) - a(258)) \times (a(132) - a(328)) \\ &= (-106) \times (-196) \\ &= \mathbf{20776} \end{aligned}$$

$$\begin{aligned} \text{Kordinat b} &= (b(41) - b(448)) \times (b(142) - b(226)) \\ &= (-407) \times (-84) \\ &= \mathbf{34188} \end{aligned}$$

$$\begin{aligned} \text{Area union} &= \text{Kordinat a} + \text{Kordinat b} - \text{area\_overlap} \\ &= 20776 + 34188 - 0 \\ &= \mathbf{54964} \end{aligned}$$

$$IoU = \frac{\text{area of Overlap}}{\text{area of Union}}$$

$$IoU = \frac{0}{54964}$$

$$IoU = 0$$

Dari Lampiran E Data uji 1. untuk mendapatkan nilai *Precision* dari prediksi image 1 dengan menggunakan persamaan berikut :

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

$$Precision = \frac{0}{0 + 1}$$

$$Precision = 0$$

Dari Lampiran E Data uji 1. untuk mendapatkan nilai *Recall* dari prediksi image 1 dengan menggunakan persamaan berikut :

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

$$Recall = \frac{0}{0 + 0}$$

$$Recall = 0$$

Dari Lampiran E Data uji 1, untuk mendapatkan nilai *Precision\_inter* dari prediksi image 1 dengan menggunakan persamaan berikut :

$$AP_i = \max AP_i$$

$$AP_1 = \max 0_1$$

$$AP_1 = 0$$

Dari Lampiran E Data uji 1, dapat diukur tingkat mAP dari prediksi dengan menggunakan persamaan sebagai berikut.

$$\begin{aligned} \text{mAP} &= \frac{1}{N} \sum_{i=1}^N AP_i \\ &= \frac{1}{63} (0 + 0 + 0 + 0 + 0 + \dots + 0.22 + 0.22 + 0.22 + 0.22 + 0.22) \\ &= 0.007 \\ &= 1\% \end{aligned}$$

## 2. Pengujian Akurasi Kedua

Pada pengujian kedua dilakukan pelatihan data latih sebanyak 160 data dengan nilai maksimum epoch sebesar 100 dan nilai *learning rate* sebesar 0.0003. Untuk

pengujian menggunakan data uji sebanyak 10 data. Adapun hasil pengujian dapat dilihat pada Lampiran E Data uji 2.

Dari Lampiran E Data uji 2. untuk proses perhitungan IoU, *precision*, *recall*, dan *precision inter* sama dengan pengujian pertama. adapun untuk mAP dari prediksi data uji 2 dengan menggunakan persamaan sebagai berikut

$$\begin{aligned}
 \text{mAP} &= \frac{1}{N} \sum_{i=1}^N AP_i \\
 &= \frac{1}{39} (0 + 0 + 0 + 0 + 0 + \dots + 0.22 + 0.22 + 0.22 + 0.22 + 0.22) \\
 &= 0,013 \\
 &= 1.33\%
 \end{aligned}$$

### 3. Pengujian Akurasi Ketiga

Pada pengujian kedua dilakukan pelatihan data latih sebanyak 160 data dengan nilai maksimum epoch sebesar 100 dan nilai learning rate sebesar 0.0005. Untuk pengujian menggunakan data uji sebanyak 10 data. Adapun hasil pengujian dapat dilihat pada Lampiran E Data uji 3.

Dari Lampiran E Data uji 3. untuk proses perhitungan IoU, *precision*, *recall*, dan *precision inter* sama dengan pengujian pertama. adapun untuk tingkat mAP dari prediksi data uji 3 dengan menggunakan persamaan sebagai berikut

$$\begin{aligned}
 \text{mAP} &= \frac{1}{N} \sum_{i=1}^N AP_i \\
 &= \frac{1}{51} (0 + 0 + 0 + 0.25 + 0.25 + \dots + 0.77 + 0.77 + 0.77 + 0.08) \\
 &= 0.27 \\
 &= 27\%
 \end{aligned}$$

### 4. Pengujian Akurasi Keempat

Pada pengujian kedua dilakukan pelatihan data latih sebanyak 160 data dengan nilai maksimum epoch sebesar 100 dan nilai learning rate sebesar 0.0007. Untuk pengujian menggunakan data uji sebanyak 10 data. Adapun hasil pengujian dapat dilihat pada Lampiran E data Uji 4.

Dari Lampiran E data Uji 4. untuk proses perhitungan IoU, *precision*, *recall*, dan *precision inter* sama dengan pengujian pertama. adapun untuk tingkat mAP dari prediksi data uji 4 dengan menggunakan persamaan sebagai berikut

$$\begin{aligned}
 \text{mAP} &= \frac{1}{N} \sum_{i=1}^N AP_i \\
 &= \frac{1}{47} (1 + 1 + 1 + 1 + 1 + \dots + 0.14 + 0.14 + 0.14 + 0.14) \\
 &= 0.43 \\
 &= 43\%
 \end{aligned}$$

#### 4.2.2.2 Rekapitulasi dan Temuan Pengujian

Dari keseluruhan pengujian yang telah dilakukan, berikut merupakan hasil rekapitulasi pengujian mAP RCNN yang ditunjukkan pada Tabel 4.8.

**Tabel 4. 8 Rekapitulasi Pengujian**

Pengujian	Epoch	$\sigma$	$k$	Learning Rate	Data Uji	<i>Pneumoni a</i>
Pengujian Satu	500	0.9	100	0.0001	10 image	1%
Pengujian Dua	100	0.9	100	0.0003	10 image	5%
Pengujian Tiga	100	0.9	100	0.0005	10 image	21%
Pengujian Empat	100	0.9	100	0.0007	10 image	43%
Pengujian Lima	100	0.9	300	0.0007	10 image	4%

Pada saat proses pengujian, dilakukan percobaan mendapatkan bahwa pada proses *Segmentation Felzenszwalb* dengan menggunakan nilai  $k$  sebesar 300 dapat mempengaruhi terhadap hasil mAP, disebabkan jumlah region terdeteksi sedikit. dibandingkan nilai  $k$  sebesar 100 dengan Epoch dan *Learning Rate* sama didapatkan mAP yang baik sebesar 43%. Dari penjelasan tersebut ditemukan bahwa Jumlah Region pada proses *Segmentation* dapat mempengaruhi tingkat mAP pada pengujian

#### 4.2.2.3 Evaluasi Hasil Pengujian Akurasi

Dari hasil pengujian terhadap 10 data uji yang telah dilakukan, evaluasi hasil mAP yang didapatkan terhadap pengenalan objek *Pneumonia* adalah sebagai berikut.

1. mAP yang didapatkan pada pengujian ke empat cukup baik yaitu 42.84% dengan *learning rate* sebesar 0.0007 dan nilai *k* sebesar 100
2. mAP dengan persentase terendah didapat pada pengujian pertama yaitu 1% dengan menggunakan nilai *learning rate* sebesar 0.001 dan nilai *k* sebesar 100
3. mAP dengan dengan *learning rate* diatas sebesar 0.0005 dan nilai *k* sebesar 100 mendapatkan mAP cukup baik terbukti dari *learning rate* sebesar 0.0005 didapat akurasi mAP sebesar 27% dan *learning rate* sebesar 0.0007 didapat 42.84% ada kenaikan akurasi cukup baik jika menggunakan *learning rate* sebesar 0.0007 dengan nilai *k* sebesar 100

Berdasarkan hasil diatas, nilai *learning rate* dan nilai *k* pada proses pelatihan berpengaruh terhadap tingkat mAP yang didapat. Persentase mAP cenderung fluktuatif seiring semakin besarnya nilai *learning rate*. Namun, semakin kecil nilai *learning rate* dan nilai *k* membuat proses pelatihan akan semakin lambat. Selain itu, tingkat kesamaan pola juga berpengaruh terhadap Prediksi R-CNN pengenalan, semakin banyak cira thorax yang hampir sama, semakin mudah juga sistem mengenali objek yang dilatih tersebut.