

BAB IV

HASIL DAN PEMBAHASAN

Bab ini akan menjelaskan mengenai hasil penelitian dan pembahasan dari masing-masing hasil penelitiannya.

4.1 Pengumpulan Data

Data yang digunakan pada penelitian ini merupakan video rekaman CCTV. Video rekaman CCTV diambil secara manual menggunakan kamera Samsung Galaxy Camera EK-GC100. Data video rekaman CCTV ini memiliki resolusi 1280x720. Terdapat dua set data yang dibuat pada penelitian ini. Set data A digunakan untuk pelatihan, sedangkan yang B akan digunakan untuk pengujian.

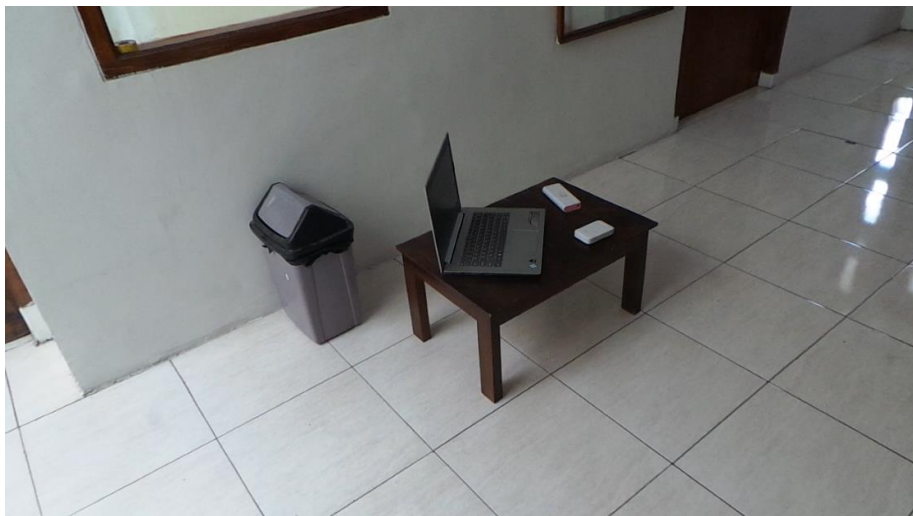
Tabel 1. Pengumpulan set data

	Set data A	Set data B
Total video	10	4
Durasi per video (detik)	60	30
Total objek	5	4
Format set data	{ <i>Path</i> , x1, y1, x2, y2}	{0,1}

Terlihat pada Tabel 1, penelitian ini akan menggunakan dua set data. set data A memiliki total yaitu 10 video yang diambil dengan masing-masing video berdurasi 1 menit. Penelitian ini menggunakan 7 video untuk pelatihan dan 3 video untuk pengujian. Pada setiap video terdapat maksimal 5 objek yang terlihat pada Gambar 19 dan Gambar 20. Waktu pengambilan video dilakukan antara pukul 10 pagi hingga pukul 2 siang. Hal ini dilakukan karena pada pukul tersebut pencahayaannya dinilai paling baik. Dibuat sebuah skenario untuk masing-masing video. Seperti setiap objek di dalam video berpindah tempat setiap 10 detik sehingga terdapat 6 tempat yang berbeda untuk setiap objeknya pada setiap video.

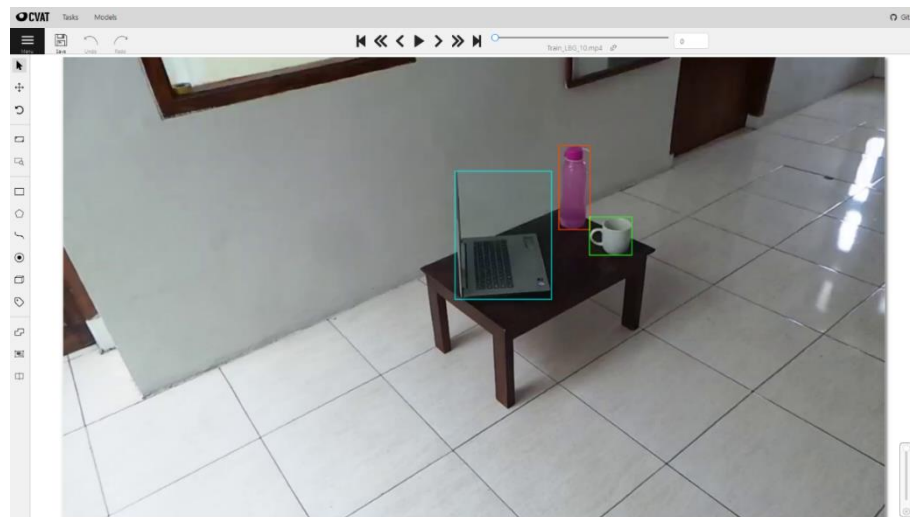


Gambar 19. Contoh rekaman set data A

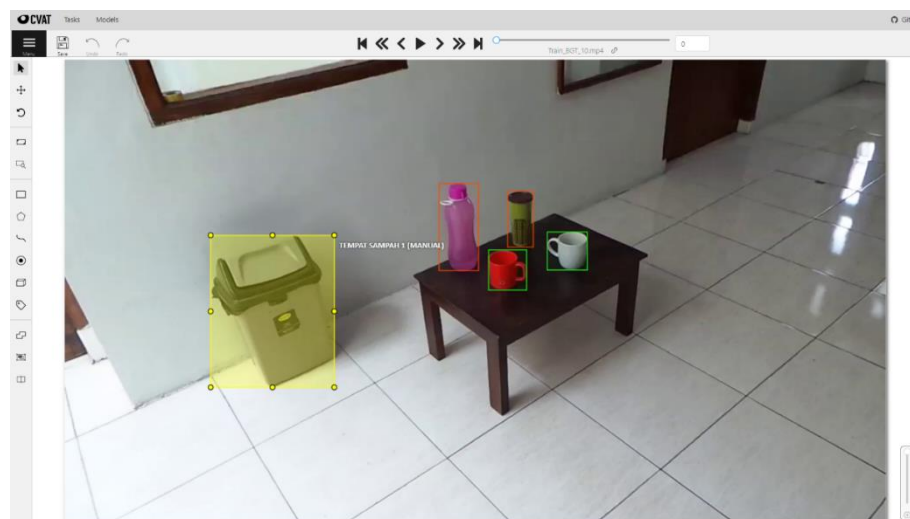


Gambar 20. Contoh rekaman set data A (2)

Anotasi dibantu menggunakan aplikasi *Computer Vision Annotation Tool* (CVAT). CVAT merupakan alat anotasi gambar dan video berbasis web yang digunakan untuk melabeli data. Aplikasi ini merupakan aplikasi *open source*. Berikut contoh pelabelan menggunakan CVAT yang terlihat pada Gambar 21 dan Gambar 22.



Gambar 21. Melabeli menggunakan CVAT



Gambar 22. Melabeli menggunakan CVAT (2)

Format set data yang digunakan untuk penelitian ini adalah format sederhana $\{Path, x1, y1, x2, y2\}$. Pada aplikasi CVAT tidak terdapat format sederhana tersebut sehingga peneliti harus membuat format sederhana dengan mengambil informasi yang dibutuhkan saja dengan menggunakan format set data yang terdapat pada CVAT. Setelah melabeli dengan CVAT, format set data yang dipilih adalah Pascal VOC. Setelah itu dibuatlah format sederhana dengan mengambil informasi yang dibutuhkan yang terdapat pada format Pascal VOC seperti pada Gambar 23.

Tia Pusparini, 2021

REGION CONVOLUTIONAL NEURAL NETWORK SIAMESE UNTUK DETEKSI OBJEK REFERENSI PADA VIDEO REKAMAN CCTV

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

```

/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000000.jpg,775,406,963,641,Laptop
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000000.jpg,569,282,617,340,Gelas
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000000.jpg,659,315,708,377,Gelas
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000000.jpg,726,244,771,337,Botol Minum
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000000.jpg,811,177,874,315,Botol Minum
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000001.jpg,775,406,963,641,Laptop
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000001.jpg,569,282,617,340,Gelas
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000001.jpg,659,315,708,377,Gelas
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000001.jpg,726,244,771,337,Botol Minum
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000001.jpg,811,177,874,315,Botol Minum
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000002.jpg,775,406,963,641,Laptop
/content/drive/My Drive/Skripsi/Aplikasi/frcnn/DATASET/Train/Train_BGL/frame_000002.jpg,569,282,617,340,Gelas

```

Gambar 23. Format set data

Set data B memiliki 4 video yang masing-masing berdurasi 30 detik. Set data ini terdapat dua macam skenario yaitu ideal dan nonideal, di mana skenario ideal memiliki kondisi bentuk objek yang ideal. Sedangkan, nonideal memiliki bentuk objek dengan variasi cahaya dan *angle* yang signifikan. Pada set data ini digunakan objek yang berbeda dengan objek yang dipakai pada proses pelatihan. Hal ini bertujuan untuk menguji kemampuan model dalam mencari objek yang tidak dikenal sebagai referensi.

Keempat video pada set data B ini diberi nama Ex_2_1, Ex_2_2, Ex_2_3, dan Ex_2_4. Video tersebut akan berdurasi 30 detik dan dibuat skenario yang berbeda. berupa pergantian objek setiap 10 detik. Setiap videonya, terdapat 4 objek yang berbeda baik itu dari aspek warna, bentuk maupun ukurannya. Video Ex_2_4 dibuat skenario yang berbeda dengan ketiga video lainnya, yaitu objek yang direkam memiliki variasi cahaya dan *angle* yang signifikan.

Pada video Ex_2_1 terdapat 3 objek berupa kotak merah, tas kuning, dan galon yang terlihat pada Gambar 24. Pergantian objek akan dilakukan setiap 10 detik. Video yang ditampilkan terdapat dua objek pada tiap *frame*-nya. Pada 10 detik pertama, akan menampilkan objek berupa kotak merah dan tas kuning. Setelah itu pada 10 detik kedua akan menampilkan kotak merah dan galon. Kemudian pada 10 detik ketiga akan menampilkan tas kuning dan galon.



Gambar 24. Contoh rekaman video Ex_2_1

Pada video Ex_2_2 terdapat 3 objek yang sama seperti pada video Ex_2_1 yaitu kotak merah, tas kuning, dan galon yang terlihat pada Gambar 25. Pergantian objek akan dilakukan setiap 10 detik. Video yang ditampilkan terdapat hanya satu objek pada tiap *frame*-nya. Objek akan tampil secara berurutan yaitu galon, tas kuning, dan kotak merah.



Gambar 25. Contoh rekaman video Ex_2_2

Pada video Ex_2_3 terdapat perbedaan objek dengan video sebelumnya. Tiga objek yang terdapat pada video ini yaitu kotak merah, tas kuning, dan sepatu hitam yang terlihat pada Gambar 26. Pergantian objek akan dilakukan setiap 10 detik. Video yang ditampilkan terdapat hanya satu objek pada tiap *frame*-nya.

Objek akan tampil secara berurutan yaitu kotak merah, tas kuning, dan sepatu hitam.



Gambar 26. Contoh rekaman video Ex_2_3

Pada video Ex_2_4 terdapat empat objek yaitu kotak merah, tas kuning, galon, dan sepatu hitam yang terlihat pada Gambar 27. Pergantian objek akan dilakukan setiap 10 detik, di mana setiap 5 detiknya *angle* pada objek akan berubah. Video yang ditampilkan terdapat dua objek pada tiap *frame*-nya.



Gambar 27. Contoh rekaman video Ex_2_4

Set data B ini dibuat untuk melakukan pengujian model. Untuk menguji model perlu dibuat label. Sebelum membuat label, dilakukan *frame extraction* menggunakan *library* OpenCV. Setelah itu, gambar yang berbeda dengan target

dilabeli dengan angka 0, dan gambar yang sama dengan target dilabeli dengan angka 1.

Kedua set data ini dibuat untuk mendukung proses penelitian ini. Set data A yang memiliki 10 video akan digunakan untuk pelatihan. Setiap videonya memiliki durasi 1 menit dan didalamnya terdapat 5 objek. Selain itu, format set data yang dipakai yaitu format sederhana yang dapat dilihat pada Gambar 23. Sedangkan, set data B dibuat untuk pengujian model yang dibuat. Set data B memiliki 3 video yang masing-masing videonya memiliki durasi 30 detik dan didalamnya terdapat 3 objek serta format set data yang dipakai yaitu angka 0 untuk gambar yang berbeda dengan target dan angka 1 untuk gambar yang sama dengan target.

4.2 Eksperimen

Pada tahap ini terdapat 3 proses yaitu implementasi model, pelatihan, dan pengujian. Pada implementasi model akan dipaparkan susunan lapisan yang akan digunakan pada setiap proses yang ada pada arsitektur model. Setelah itu terdapat tahap pelatihan, pada tahap ini akan dijelaskan konfigurasi yang dipakai oleh tiap arsitektur model serta hasil dari model tersebut. Pada tahap akhir terdapat tahap pengujian, tahap ini akan dilakukan pengujian menggunakan arsitektur model yang telah melalui tahap pelatihan. Pengujian akan dilakukan menggunakan set data yang telah dibuat sebelumnya.

4.2.1 Implementasi Model

Pada subbab ini akan dipaparkan arsitektur pada setiap modelnya. Setiap arsitektur memiliki lapisan dan konfigurasi yang berbeda. Berikut arsitektur metode yang digunakan pada penelitian ini.

4.2.1.1 Region Proposal Network

Sesuai dengan apa yang telah dijelaskan pada BAB II subbab 2.10, RPN digunakan untuk menghasilkan kotak wilayah proposal deteksi objek yang baik. Kotak wilayah proposal itu bisa disebut juga *anchor*. *Anchor* tersebut akan menjadi masukan pada proses selanjutnya yaitu proses menghitung nilai kesamaan antara dua gambar. Metode ini dipakai pada dua arsitektur yang dibuat untuk penelitian

ini. Berikut lapisan konvolusi yang dipakai pada metode RPN yang ditampilkan pada Tabel 2.

Tabel 2. Susunan lapisan konvolusi RPN

No.	Tipe Lapisan	Bentuk keluaran	Parameter
1.	Masukan	(None, None, None, 3)	0
2.	Conv2d + ReLu	(None, None, None, 64)	1792
3.	Conv2d + ReLu	(None, None, None, 64)	36928
4.	Max Pooling	(None, None, None, 64)	0
5.	Conv2d + ReLu	(None, None, None, 128)	73856
6.	Conv2d + ReLu	(None, None, None, 128)	147584
7.	Max Pooling	(None, None, None, 128)	0
8.	Conv2d + ReLu	(None, None, None, 256)	295168
9.	Conv2d + ReLu	(None, None, None, 256)	590080
10.	Conv2d + ReLu	(None, None, None, 256)	590080
11.	Max Pooling	(None, None, None, 256)	0
12.	Conv2d + ReLu	(None, None, None, 512)	1180160
13.	Conv2d + ReLu	(None, None, None, 512)	2359808
14.	Conv2d + ReLu	(None, None, None, 512)	2359808
15.	Max Pooling	(None, None, None, 512)	0
16.	Conv2d + ReLu	(None, None, None, 512)	2359808
17.	Conv2d + ReLu	(None, None, None, 512)	2359808
18.	Conv2d + ReLu	(None, None, None, 512)	2359808
19.	Conv2d + Sigmoid	(None, None, None, 9)	4617
20.	Conv2d + Linear	(None, None, None, 36)	18468
Total Parameter			17097581

Metode RPN tersusun dari jaringan VGG-16 dan dua lapisan konvolusi yang berfungsi sebagai *fully connected layer*. Jaringan VGG-16 pada metode ini

memiliki 17 lapisan yang terdiri dari 13 lapisan konvolusi dan 4 *max pooling*. Aktivasi yang digunakan pada lapisan konvolusi yaitu *Rectified Linear Unit* (ReLU). Dua lapisan *fully connected layer* digunakan aktivasi *sigmoid* untuk klasifikasi dan aktivasi linear untuk regresi. Total parameter metode ini yaitu 17097581.

4.2.1.2 Siamese CNN

Arsitektur ini akan memiliki dua gambar masukan yaitu gambar target dan gambar yang dihasilkan oleh metode RPN. Metode Siamese CNN ini terdiri dari dua masukan yang terlebih dahulu diproses melalui lapisan konvolusi dengan bobot dan parameter yang identik yang ditampilkan pada Tabel 3. Setelah itu keluaran dua masukan tersebut digabungkan dan diproses pada lapisan terhubung sepenuhnya (*fully connected layer*) yang ditampilkan pada Tabel 4.

Tabel 3. Susunan lapisan konvolusi identik Siamese CNN

No	Tipe Lapisan	Bentuk keluaran	Parameter
1	Conv2d + ReLu	(None, 128, 128, 64)	1792
2	Conv2d + ReLu	(None, 128, 128, 64)	36928
3	Conv2d + ReLu	(None, 126, 126, 64)	36928
4	<i>Max Pooling</i>	(None, 63, 63, 64)	0
5	<i>Dropout</i>	(None, 63, 63, 64)	0
6	Conv2d + ReLu	(None, 63, 63, 128)	73856
7	Conv2d + ReLu	(None, 63, 63, 128)	147584
8	<i>Max Pooling</i>	(None, 31, 31, 128)	0
9	<i>Dropout</i>	(None, 31, 31, 128)	0
10	Conv2d + ReLu	(None, 31, 31, 512)	590336
11	Conv2d + ReLu	(None, 31, 31, 512)	2359808
12	Conv2d + ReLu	(None, 31, 31, 512)	2359808
13	<i>Max Pooling</i>	(None, 15, 15, 512)	0
14	<i>Dropout</i>	(None, 15, 15, 512)	0

No	Tipe Lapisan	Bentuk keluaran	Parameter
15	Conv2d + ReLu	(None, 15, 15, 512)	2359808
16	Conv2d + ReLu	(None, 15, 15, 512)	2359808
17	Conv2d + ReLu	(None, 15, 15, 512)	2359808
18	<i>Flatten</i>	(None, 115200)	0
19	<i>Dense</i>	(None, 32)	3686432
Total Parameter			16372896

Tabel 4. Susunan lapisan Siamese CNN

No	Tipe Lapisan	Bentuk keluaran	Parameter	<i>Connected to</i>
1	Masukan_1	(None, 128, 128, 3)	0	
2	Masukan_2	(None, 128, 128, 3)	0	
3	<i>Sequential</i>	(None, 32)	16372896	Masukan_1[0][0] Masukan_2[0][0]
4	<i>Lambda</i>	(None, 32)	0	<i>Sequential</i> [0][0] <i>Sequential</i> [1][0]
5	<i>Dense</i>	(None, 1)	33	<i>Lambda</i> [0][0]
Total Parameter				16372929

Metode ini memiliki 2 masukan gambar dengan ukuran 128x128 yang akan diproses pada susunan lapisan konvolusi dengan bobot dan parameter yang identik. Susunan lapisan konvolusi identik tersebut memiliki 11 lapisan konvolusi dengan aktivasi ReLU, 3 *max pooling*, 3 *dropout*, dan 1 lapisan *dense* 32 *units* dengan aktivasi *sigmoid*. Kemudian hasil dari proses tersebut akan digabungkan dan pada lapisan terakhir terdapat *dense* satu *units* dengan aktivasi *sigmoid*. Total parameter metode ini adalah 16372929.

4.2.1.3 Siamese CNN *Euclidean*

Arsitektur ini akan memiliki dua gambar masukan yaitu gambar target dan gambar yang dihasilkan oleh metode RPN. Pada arsitektur ini digunakan *euclidean*

distance untuk menghitung nilai perbedaan matriks antara dua gambar masukan. Perhitungan tersebut akan digunakan pada proses pengujian. Susunan lapisan konvolusi dengan bobot dan parameter yang identik ditampilkan pada Tabel 5. Keluaran dua vektor konvolusi identik dari dua gambar masukan tersebut akan digabungkan yang terlihat pada Tabel 6.

Tabel 5. Susunan lapisan konvolusi identik Siamese CNN *Euclidean*

No	Tipe Lapisan	Bentuk keluaran	Parameter
1	Conv2d + ReLu	(None, 128, 128, 64)	1792
2	Conv2d + ReLu	(None, 126, 126, 64)	36928
3	<i>Max Pooling</i>	(None, 63, 63, 64)	0
4	<i>Dropout</i>	(None, 63, 63, 64)	0
5	Conv2d + ReLu	(None, 63, 63, 128)	73856
6	Conv2d + ReLu	(None, 63, 63, 128)	147584
7	Conv2d + ReLu	(None, 63, 63, 128)	147584
8	<i>Max Pooling</i>	(None, 31, 31, 128)	0
9	Conv2d + ReLu	(None, 31, 31, 512)	590336
10	Conv2d + ReLu	(None, 31, 31, 512)	2359808
11	Conv2d + ReLu	(None, 31, 31, 512)	2359808
12	<i>Max Pooling</i>	(None, 15, 15, 512)	2359808
13	Conv2d + ReLu	(None, 15, 15, 512)	2359808
14	Conv2d + ReLu	(None, 15, 15, 512)	2359808
15	Conv2d + ReLu	(None, 15, 15, 512)	2359808
16	<i>Flatten</i>	(None, 115200)	0
Total Parameter			12797120

Tabel 6. Susunan lapisan Siamese CNN *Euclidean*

No	Tipe Lapisan	Bentuk keluaran	Parameter	Connected to
1	Masukan_1	(None, 128, 128, 3)	0	
2	Masukan_2	(None, 128, 128, 3)	0	
3	<i>Sequential</i>	(None, 115200)	12797120	Masukan_1[0][0] Masukan_2[0][0]
4	<i>Concatenate</i>	(None, 230400)	0	<i>Sequential</i> [0][0] <i>Sequential</i> [1][0]
Total Parameter			12797120	

Metode ini memiliki 2 masukan gambar dengan ukuran 128x128 yang akan diproses pada susunan lapisan konvolusi dengan bobot dan parameter yang identik. Susunan lapisan konvolusi identik tersebut memiliki 11 lapisan konvolusi dengan aktivasi ReLU, 3 *max pooling* dan tambahan *dropout*. Kemudian hasil dari proses tersebut akan digabungkan dan diubah menjadi vektor. Vektor tersebut akan digunakan untuk menghitung nilai kesamaan menggunakan perhitungan *euclidean distance* pada saat pengujian. Total parameter metode ini adalah 12797120.

4.2.2 Pelatihan

Pada tahap ini akan dilakukan proses pelatihan berdasarkan arsitektur model yang telah direncanakan sebelumnya yang dijelaskan pada subbab 4.2.1. Model RPN, Siamese CNN dan juga Siamese CNN *Euclidean* dilatih secara terpisah.

4.2.2.1 Region Proposal Network

Proses pelatihan pada arsitektur RPN yang telah dibuat akan menggunakan set data A dengan konfigurasi yang dapat dilihat pada Tabel 7.

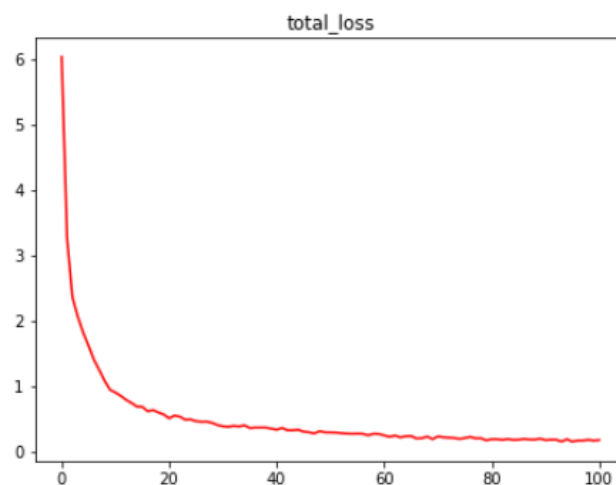
Tabel 7. Konfigurasi metode RPN

No	Konfigurasi	Keterangan
1	Data pelatihan	10007 anotasi

No	Konfigurasi	Keterangan
2	<i>Epoch</i>	100
3	<i>Optimizer</i>	Adam ($lr=1^{e-6}$)
4	<i>Loss function</i>	RPN

Arsitektur RPN dilatih sebanyak 100 *epoch*, setiap *epoch* nya memiliki sebanyak 1000 iterasi. Data yang dipakai sebanyak 10007 anotasi. Arsitektur ini menggunakan Adam sebagai *optimizer* dengan *learning rate* 0.000001. *Loss function* yang dipakai pada metode ini adalah penjumlahan dari *classification loss* dan *bounding box regression loss* yang telah dijelaskan pada BAB II subbab 2.10.

Pada Gambar 28, terlihat grafik *loss* yang dihasilkan oleh metode RPN menggunakan konfigurasi yang telah ditentukan sebelumnya. Hasil pelatihan yang dilakukan pada metode ini mendapatkan *loss* sebesar 0.184. Pada grafik tersebut, terlihat bahwa dari epoch 80 hingga 100 menunjukkan hasil *loss* yang stagnan sehingga dapat dikatakan metode ini optimum pada rentang epoch 80 hingga 100.



Gambar 28. Grafik *loss* RPN

4.2.2.2 Siamese CNN

Proses pelatihan pada arsitektur Siamese CNN yang telah dibuat akan menggunakan set data A dengan konfigurasi yang dapat dilihat pada Tabel 8.

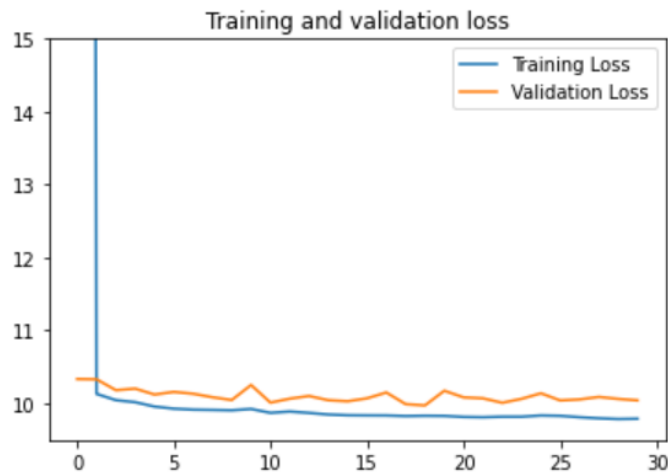
Tabel 8. Konfigurasi metode Siamese CNN

No	Konfigurasi	Keterangan
1	Data pelatihan	9583 pasang
2	Data validasi	4107 pasang
3	<i>Epoch</i>	19
4	<i>Optimizer</i>	Adam ($lr=5^{e-4}$)
5	<i>Loss Function</i>	Binary Crossentropy

Pada awal pelatihan, metode ini dilatih sebanyak 30 *epoch*. Tetapi dapat dilihat pada grafik Gambar 29, pada rentang *epoch* ke 20 hingga 30, *train_loss* masih terus menurun ke angka 9.797 tetapi *val_loss* mulai naik hingga *epoch* 30 dengan angka 10.084 sehingga model dianggap optimum di *val_loss* 9.974 pada *epoch* ke 19. Maka dari itu diambil model yang telah dilatih sebanyak 19 *epoch* yang setiap *epoch* nya memiliki sebanyak 300 iterasi.

Data yang dipakai sebanyak 9583 pasang untuk pelatihan dan 4107 pasang untuk validasi. Arsitektur ini menggunakan *optimizer* Adam dengan *learning rate* 0.0005 dan *loss function* *Binary Crossentropy*. *Loss function* tersebut digunakan karena eksperimen ini merupakan permasalahan klasifikasi biner dimana nilai target akan diset [0, 1].

Hasil *loss* yang dihasilkan metode Siamese CNN ini menggunakan konfigurasi yang dapat dilihat pada Tabel 8. Hasil pelatihan pada metode ini mendapatkan *train_loss* terkecil sebesar 9.832 dengan *val_loss* sebesar 9.974 pada *epoch* ke 19.



Gambar 29. Grafik *loss* Siamese CNN

4.2.2.3 Siamese CNN *Euclidean*

Proses pelatihan pada arsitektur Siamese CNN *Euclidean* yang telah dibuat akan menggunakan set data A dengan konfigurasi yang dapat dilihat pada Tabel 9.

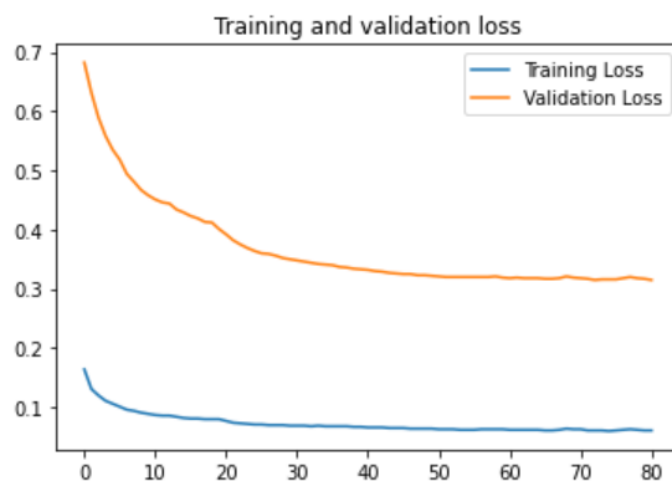
Tabel 9. Konfigurasi metode Siamese CNN *Euclidean*

No	Konfigurasi	Keterangan
1	Data pelatihan	9583 pasang
2	Data validasi	4107 pasang
3	<i>Epoch</i>	81
4	<i>Optimizer</i>	SGD ($lr=5^{e-4}$)
5	<i>Loss Function</i>	Binary Crossentropy

Arsitektur ini dilatih sebanyak 81 *epoch*, setiap *epoch* nya memiliki sebanyak 300 iterasi. Data yang dipakai sebanyak 9583 pasang untuk pelatihan dan 4107 pasang untuk validasi. Banyaknya *epoch* dinilai yang paling optimum dilihat dari grafik yang dihasilkan dari proses pelatihan. Arsitektur ini menggunakan

optimizer Stochastic Gradient Descent dengan *learning rate* 0.0005 dan *loss function Binary Crossentropy*.

Pada Gambar 30, terlihat hasil *loss* yang dihasilkan metode Siamese CNN *Euclidean* menggunakan konfigurasi yang telah ditentukan sebelumnya. Pada grafik tersebut, terlihat bahwa dari *epoch* 73 hingga 81 menunjukkan hasil *loss* yang stagnan sehingga dapat dikatakan metode ini optimum pada rentang *epoch* 73 hingga 81 dengan *train loss* sebesar 0.061 dan *val loss* sebesar 0.315.



Gambar 30. Grafik *loss* Siamese CNN *Euclidean*

4.2.3 Pengujian

Pada tahap ini akan dilakukan pengujian untuk mendapatkan hasil kinerja dari model yang telah dilatih sebelumnya. Hasil kinerja model tersebut akan dibandingkan untuk mendapatkan model terbaik dalam mendeteksi objek pada video rekaman CCTV.

Terdapat dua model yang akan diuji yaitu penggabungan antara model RPN dengan Siamese CNN yang disebut juga RCNN Siamese CNN yang dapat dilihat pada Gambar 32 dan penggabungan antara model RPN dengan Siamese CNN *Euclidean* disebut juga RCNN Siamese CNN *Euclidean* yang dapat dilihat pada Gambar 33. Model yang telah dilatih akan disimpan keseluruhan arsitekturnya dan pada saat dilakukan pengujian hanya perlu memuat model yang telah disimpan tersebut. Kedua arsitektur yang dibuat memiliki keluaran yang berbeda.

Pengujian 2 model tersebut dilakukan menggunakan set data B yang memiliki dua macam skenario yang telah dijelaskan pada subbab pengumpulan data. Set data ini memiliki 4 video yang terdapat 4 objek target yang akan diuji yaitu kotak merah, tas kuning, galon dan sepatu hitam yang terlihat pada Gambar 31. Gambar yang berbeda dengan objek target dilabeli dengan angka 0 dan gambar yang sama dengan objek target dilabeli dengan angka 1. Pengujian dilakukan untuk mendapatkan hasil mAP dan akurasi terbaik.



Gambar 31. Objek-objek target

1) Skenario Ideal

Pada pengujian ini, pengujian akan dilakukan dengan skenario yang memiliki kondisi ideal yang sudah dijelaskan pada subbab pengumpulan data. Skenario ini memiliki 3 video yaitu video Ex_2_1, Ex_2_2, dan Ex_2_3. Beberapa *threshold* digunakan untuk mencari hasil yang optimal pada kedua model yang dibuat yang dapat dilihat pada Tabel 10 hingga Tabel 13.

a. RCNN Siamese CNN

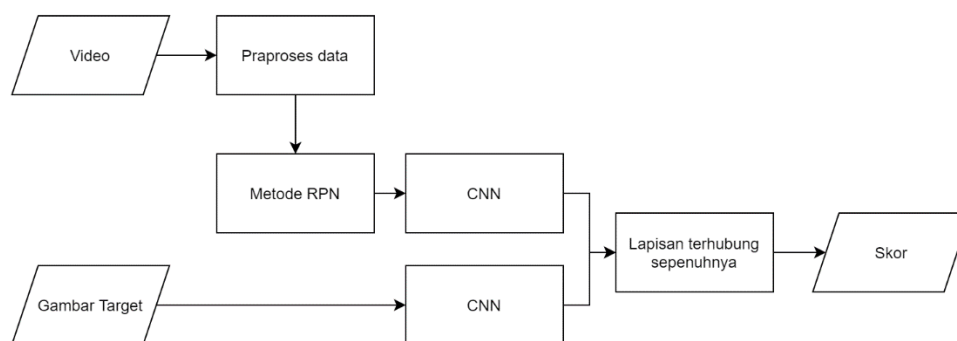
Model ini merupakan penggabungan antara metode RPN dengan metode Siamese CNN yang disebut juga RCNN Siamese CNN yang dapat dilihat pada

Tia Pusparini, 2021

REGION CONVOLUTIONAL NEURAL NETWORK SIAMESE UNTUK DETEKSI OBJEK REFERENSI PADA VIDEO REKAMAN CCTV

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

Gambar 32. Sebelum masuk ke dalam proses pendeteksian objek menggunakan metode RPN, terlebih dahulu dilakukan praproses data berupa *frame extraction*. Model ini memiliki keluaran nilai kesamaan dari hasil prediksi model. Nilai kesamaan yang dihasilkan model ini akan menentukan *threshold* yang digunakan untuk pengujian.



Gambar 32. Alur pengujian RCNN Siamese CNN

Berikut hasil pengujian yang didapatkan model RCNN Siamese CNN menggunakan dua skenario yang berbeda pada set data B.

Tabel 10. Hasil mAP pengujian skenario ideal pada RCNN Siamese CNN

No	Nama Video	Nama Objek	<i>Threshold</i>	mAP	<i>Precision</i>	<i>Recall</i>
1	Ex_2_1	Kotak Merah	0.28	0.99	1.0	0.96875
			0.29	0.99	1.0	0.96875
			0.30	0.99	1.0	0.953125
		Tas Kuning	0.28	0.32	0.0	0.0
			0.29	0.32	0.0	0.0
			0.30	0.32	0.0	0.0
		Galon	0.28	0.88	0.60295	0.68333
			0.29	0.88	0.60295	0.68333
			0.30	0.88	0.95349	0.68333
2	Ex_2_2		0.28	1.00	1.0	1.0

No	Nama Video	Nama Objek	<i>Threshold</i>	mAP	<i>Precision</i>	<i>Recall</i>	
		Kotak Merah	0.29	1.00	1.0	1.0	
			0.30	1.00	1.0	1.0	
		Tas Kuning	0.28	Tidak Terdeteksi			
			0.29				
			0.30				
		Galon	0.28	0.92	1.0	0.875	
			0.29	0.92	1.0	0.875	
			0.30	0.92	1.0	0.875	
		3	Ex_2_3	Kotak Merah	0.28	0.98	0.43243
0.29	0.98				0.45714	1.0	
0.30	0.98				0.91176	0.96875	
Tas Kuning	0.28			0.87	1.0	0.8	
	0.29			0.55	1.0	0.33333	
	0.30			0.44	1.0	0.16666	
Sepatu Hitam	0.28			0.89	0.5	0.83334	
	0.29			0.89	0.51020	0.83334	
	0.30			0.89	0.625	0.83334	

Tabel 10 menampilkan hasil mAP dari pengujian yang dilakukan menggunakan metode RCNN Siamese. Pengujian tersebut dilakukan menggunakan tiga *threshold* yaitu 0.28, 0.29, dan 0.30. Model ini diuji menggunakan 3 video yaitu Ex_2_1, Ex_2_2, dan Ex_2_3. Selain itu, terdapat 4 objek target yang akan dideteksi yaitu kotak merah, tas kuning, galon, dan sepatu hitam.

Pada video Ex_2_1 terdapat tiga objek yang dideteksi yaitu kotak merah, galon, dan tas kuning. Kotak merah mendapatkan mAP terbesar yaitu mencapai 0.99 dengan menggunakan *threshold* 0.28 dan 0.29. Galon mendapatkan mAP terbesar yaitu mencapai 0.88 dengan menggunakan *threshold* 0.30. Sedangkan tas kuning

mendapatkan mAP terbesar yaitu mencapai 0.32 dengan menggunakan semua *threshold*.

Pada video Ex_2_2 terdapat tiga objek yang sama yang terdapat pada video Ex_2_1 yaitu kotak merah, galon, dan tas kuning. Kotak merah mendapatkan mAP terbesar yaitu mencapai 1.00 dengan menggunakan semua *threshold*. Galon mendapatkan mAP terbesar yaitu mencapai 0.92 dengan menggunakan semua *threshold*. Sedangkan tas kuning yang terdapat pada video ini tidak terdeteksi sehingga tidak dapat dihitung mAPnya.

Pada video Ex_2_3 terdapat tiga objek yang dideteksi yaitu kotak merah, sepatu hitam, dan tas kuning. Kotak merah mendapatkan mAP terbesar yaitu mencapai 0.98 dengan menggunakan *threshold* 0.30. Sepatu hitam mendapatkan mAP terbesar yaitu mencapai 0.89 dengan menggunakan *threshold* 0.30. Sedangkan tas kuning mendapatkan mAP terbesar yaitu mencapai 0.87 dengan menggunakan *threshold* 0.28.

Tabel 11. Hasil akurasi pengujian skenario ideal pada RCNN Siamese CNN

No	Nama Video	Nama Objek	<i>Threshold</i>	<i>Frame</i> Terdeteksi		Akurasi
				Benar	Salah	
1	Ex_2_1 (93)	Kotak Merah	0.28	91	2	0.979
			0.29	91	2	0.979
			0.30	90	3	0.9678
		Tas Kuning	0.28	49	44	0.527
			0.29	49	44	0.527
			0.30	49	44	0.527
		Galon	0.28	47	46	0.505
			0.29	57	36	0.613
			0.30	72	21	0.774
2	Ex_2_2 (93)	Kotak Merah	0.28	93	0	1
			0.29	93	0	1

No	Nama Video	Nama Objek	Threshold	Frame Terdeteksi		Akurasi		
				Benar	Salah			
3	Ex_2_3 (92)		0.30	93	0	1		
			Tas Kuning	0.28	Tidak Terdeteksi			
				0.29				
		0.30						
		Galon	0.28	89	4	0.957		
			0.29	89	4	0.957		
			0.30	89	4	0.957		
				Kotak Merah	0.28	50	42	0.544
					0.29	54	38	0.587
					0.30	88	4	0.957
				Tas Kuning	0.28	86	6	0.935
					0.29	72	20	0.783
0.30	67				25	0.728		
Sepatu Hitam	0.28			62	30	0.674		
	0.29			63	29	0.685		
	0.30			72	20	0.783		

Tabel 11 menampilkan hasil akurasi dari pengujian yang dilakukan menggunakan metode RCNN Siamese CNN. Pengujian tersebut dilakukan menggunakan tiga *threshold* yaitu 0.28, 0.29, dan 0.30. Model ini diuji menggunakan 3 video yaitu Ex_2_1, Ex_2_2, dan Ex_2_3. Selain itu, terdapat 4 objek target yang akan dideteksi yaitu kotak merah, tas kuning, galon, dan sepatu hitam.

Pada video Ex_2_1 terdapat tiga objek yang dideteksi yaitu kotak merah, galon, dan tas kuning. Kotak merah mendapatkan akurasi terbesar yaitu mencapai 0.979 pada *threshold* 0.28 dan 0.29. Galon mendapatkan akurasi terbesar yaitu mencapai

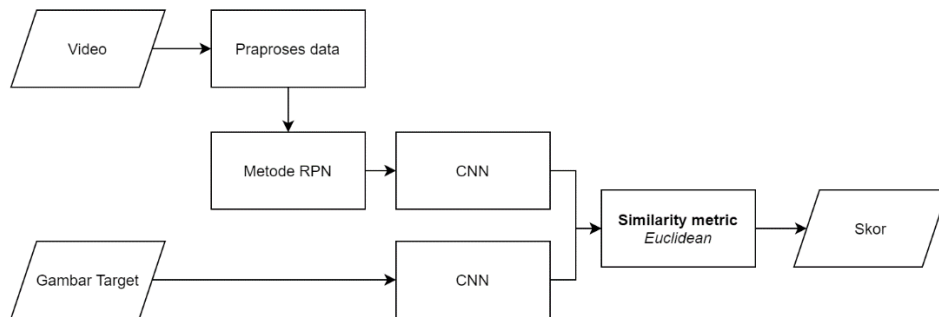
0.774 pada *threshold* 0.30. Sedangkan tas kuning mendapatkan akurasi terbesar yaitu mencapai 0.527 pada semua *threshold*.

Pada video Ex_2_2 terdapat tiga objek yang sama yang terdapat pada video Ex_2_1 yaitu kotak merah, galon, dan tas kuning. Kotak merah mendapatkan akurasi terbesar yaitu mencapai 1 pada semua *threshold*. Galon mendapatkan akurasi terbesar yaitu mencapai 0.957 pada semua *threshold*. Sedangkan tas kuning yang terdapat pada video ini tidak terdeteksi sehingga tidak dapat dihitung akurasinya.

Pada video Ex_2_3 terdapat tiga objek yang dideteksi yaitu kotak merah, sepatu hitam, dan tas kuning. Kotak merah mendapatkan akurasi terbesar yaitu mencapai 0.957 pada *threshold* 0.30. Sepatu hitam mendapatkan akurasi terbesar yaitu mencapai 0.783 pada *threshold* 0.30. Sedangkan tas kuning mendapatkan akurasi terbesar yaitu mencapai 0.935 pada *threshold* 0.28.

b. RCNN Siamese CNN *Euclidean*

Model ini merupakan penggabungan antara metode RPN dan metode Siamese CNN *Euclidean* yang disebut juga RCNN Siamese CNN *Euclidean* yang dapat dilihat pada Gambar 33. Sebelum masuk ke dalam proses pendeteksian objek menggunakan metode RPN, terlebih dahulu dilakukan praproses data berupa *frame extraction*. Model ini memiliki keluaran vektor, yang mana vektor tersebut akan dihitung menggunakan *euclidean distance* untuk menghasilkan nilai kesamaan antara dua masukan. Untuk menghitung *euclidean distance* akan digunakan *library sklearn*.



Gambar 33. Alur pengujian RCNN Siamese CNN *Euclidean*

Beberapa *threshold* digunakan untuk mencari hasil yang optimal pada model yang dibuat. Berikut hasil pengujian yang didapatkan model RCNN Siamese CNN *Euclidean* menggunakan set data B.

Tabel 12. Hasil mAP pengujian skenario ideal pada RCNN Siamese CNN *Euclidean*

No	Nama Video	Nama Objek	<i>Threshold</i>	mAP	<i>Precision</i>	<i>Recall</i>	
1	Ex_2_1	Kotak Merah	7.4	0.92	1.0	0.734375	
			7.6	1.00	1.0	0.984375	
			7.8	1.00	1.0	0.984375	
		Tas Kuning	7.4	0.32	0.0	0.0	
			7.6	0.32	0.0	0.0	
			7.8	0.32	0.0	0.0	
		Galon	7.4	0.89	1.0	0.683333	
			7.6	0.89	1.0	0.683333	
			7.8	0.89	1.0	0.683333	
2	Ex_2_2	Kotak Merah	7.4	0.95	1.0	0.933333	
			7.6	1.00	1.0	1.0	
			7.8	1.00	1.0	1.0	
		Tas Kuning	7.4	Tidak Terdeteksi			
			7.6				

No	Nama Video	Nama Objek	<i>Threshold</i>	mAP	<i>Precision</i>	<i>Recall</i>		
3	Ex_2_3	Galon	7.8					
			7.4	0.92	1.0	0.875		
			7.6	0.92	1.0	0.875		
					7.8	0.92	1.0	0.875
		Kotak Merah	7.4	1.00	1.0	1.0		
			7.6	1.00	1.0	1.0		
			7.8	1.00	1.0	1.0		
		Tas Kuning	7.4	1.00	1.0	1.0		
			7.6	1.00	1.0	1.0		
7.8	1.00		1.0	1.0				
Sepatu Hitam	7.4	0.62	1.0	0.433333				
	7.6	0.62	1.0	0.433333				
	7.8	0.62	1.0	0.433335				

Tabel 12 menampilkan hasil mAP dari pengujian menggunakan metode RCNN Siamese *Euclidean*. Pengujian tersebut dilakukan menggunakan tiga *threshold* yaitu 7.4, 7.6, dan 7.8. Model ini diuji menggunakan 3 video yaitu Ex_2_1, Ex_2_2, dan Ex_2_3. Selain itu, terdapat 4 objek target yang akan dideteksi yaitu kotak merah, tas kuning, galon, dan sepatu hitam.

Pengujian pada video Ex_2_1 terdapat tiga objek yang dideteksi yaitu kotak merah, galon, dan tas kuning. Kotak merah mendapatkan mAP terbesar yaitu mencapai 1.00 dengan menggunakan *threshold* 7.6 dan 7.8. Galon mendapatkan mAP sebesar 0.89 dengan menggunakan semua *threshold*. Sedangkan tas kuning mendapatkan mAP yang paling terkecil diantara objek target lainnya yaitu sebesar 0.32 dengan menggunakan semua *threshold*.

Pengujian pada video Ex_2_2 dilakukan deteksi dengan tiga objek yang sama dengan video Ex_2_1 yaitu kotak merah, galon, dan tas kuning. Kotak merah

mendapatkan mAP 1.00 dengan menggunakan *threshold* 7.6 dan 7.8. Galon mendapatkan mAP 0.92 dengan menggunakan semua *threshold*. Sedangkan tas kuning yang terdapat pada video ini tidak terdeteksi sehingga tidak dapat dihitung menggunakan mAP.

Pengujian pada video Ex_2_3 terdapat tiga objek yang dideteksi yaitu kotak merah, sepatu hitam, dan tas kuning. Kotak merah mendapatkan mAP sebesar 1.00 dengan menggunakan semua *threshold*. Sepatu hitam mendapatkan mAP sebesar 0.62 dengan menggunakan semua *threshold*. Sedangkan tas kuning mendapatkan mAP sebesar 1.00 dengan menggunakan semua *threshold*.

Tabel 13. Hasil akurasi pengujian skenario ideal pada RCNN Siamese CNN *Euclidean*

No	Nama Video	Nama Objek	<i>Threshold</i>	<i>Frame</i> Terdeteksi		Akurasi	
				Benar	Salah		
1	Ex_2_1 (93)	Kotak Merah	7.4	76	17	0.817	
			7.6	92	1	0.989	
			7.8	92	1	0.989	
		Tas Kuning	7.4	47	46	0.505	
			7.6	47	46	0.505	
			7.8	47	46	0.505	
		Galon	7.4	74	19	0.796	
			7.6	74	19	0.796	
			7.8	74	19	0.796	
2	Ex_2_2 (93)	Kotak Merah	7.4	91	2	0.979	
			7.6	93	0	1	
			7.8	93	0	1	
		Tas Kuning	7.4	Tidak Terdeteksi			
			7.6				
			7.8				
		Galon	7.4	89	4	0.957	

No	Nama Video	Nama Objek	Threshold	Frame Terdeteksi		Akurasi
				Benar	Salah	
3	Ex_2_3 (92)	Kotak Merah	7.6	89	4	0.957
			7.8	89	4	0.957
			7.4	92	0	1
		Tas Kuning	7.6	92	0	1
			7.8	92	0	1
			7.4	92	0	1
		Sepatu Hitam	7.6	92	0	1
			7.8	92	0	1
			7.4	75	17	0.815
		7.6	75	17	0.815	
		7.8	75	17	0.815	

Tabel 13 menampilkan hasil akurasi dari pengujian menggunakan metode RCNN Siamese *Euclidean*. Pengujian tersebut dilakukan menggunakan tiga *threshold* yaitu 7.4, 7.6, dan 7.8. Model ini diuji menggunakan 3 video yaitu Ex_2_1, Ex_2_2, dan Ex_2_3. Selain itu, terdapat 4 objek target yang akan dideteksi yaitu kotak merah, tas kuning, galon, dan sepatu hitam.

Pengujian pada video Ex_2_1 terdapat tiga objek yang dideteksi yaitu kotak merah, galon, dan tas kuning. Kotak merah mendapatkan akurasi terbesar yaitu mencapai akurasi 0.989 dengan menggunakan *threshold* 7.6 dan 7.8. Galon mendapatkan akurasi sebesar 0.796 dengan menggunakan semua *threshold*. Sedangkan tas kuning mendapatkan akurasi yang paling terkecil diantara objek target lainnya yaitu sebesar 0.505 dengan menggunakan semua *threshold*.

Pengujian pada video Ex_2_2 dilakukan deteksi dengan tiga objek yang sama dengan video Ex_2_1 yaitu kotak merah, galon, dan tas kuning. Kotak merah mendapatkan akurasi yaitu mencapai 1 dengan menggunakan *threshold* 7.6 dan 7.8. Galon mendapatkan akurasi yaitu sebesar 0.957 pada semua *threshold*. Sedangkan

tas kuning yang terdapat pada video ini tidak terdeteksi sehingga tidak dapat dihitung akurasinya.

Pengujian pada video Ex_2_3 terdapat tiga objek yang dideteksi yaitu kotak merah, sepatu hitam, dan tas kuning. Kotak merah mendapatkan akurasi yaitu mencapai 1 dengan menggunakan semua *threshold*. Sepatu hitam mendapatkan akurasi yaitu sebesar 0.815 dengan menggunakan semua *threshold*. Sedangkan tas kuning mendapatkan akurasi yaitu sebesar 1 dengan menggunakan semua *threshold*.

2) Skenario Nonideal

Pada pengujian ini, pengujian akan dilakukan dengan skenario dengan kondisi nonideal yang sudah dijelaskan pada subbab pengumpulan data. Pengujian dilakukan menggunakan model yang sama yang dipakai untuk pengujian pada skenario ideal. Skenario ini memiliki satu video yaitu video Ex_2_4. Pengujian dilakukan menggunakan satu *threshold* dengan mAP dan akurasi terbaik yang didapatkan dari hasil pengujian yang dilakukan menggunakan skenario ideal. Hasil pengujian mAP dan akurasi dengan kedua model yang dibuat dapat dilihat pada Tabel 14 dan Tabel 15.

Tabel 14. Hasil mAP dengan pengujian skenario nonideal

Nama Video	Nama Objek	Nama Model	<i>Threshold</i>	mAP	<i>Precision</i>	<i>Recall</i>
Ex_2_4	Kotak Merah	RCNN Siamese CNN	0.30	0.66	0.763	0.630
	Tas Kuning		0.30	0.54	1.0	0.111
	Galon		0.30	0.60	0.6	1.0
	Sepatu Hitam		0.30	0.71	0.727	0.645
	Kotak Merah	RCNN Siamese	7.8	0.80	0.968	0.652

Nama Video	Nama Objek	Nama Model	<i>Threshold</i>	mAP	<i>Precision</i>	<i>Recall</i>
	Tas Kuning	CNN <i>Euclidean</i>	7.8	0.76	0.933	0.622
	Galon		7.8	0.90	1.0	0.849
	Sepatu Hitam		7.8	0.74	1.0	0.226

Tabel 14 menampilkan hasil mAP dari pengujian dengan skenario nonideal menggunakan kedua metode yang dibuat. Pada pengujian tersebut terdapat 4 objek yaitu kotak merah, tas kuning, galon, dan sepatu hitam.

Pada pengujian menggunakan model RCNN Siamese CNN, kotak merah mendapatkan mAP sebesar 0.66. Tas kuning mendapatkan mAP sebesar 0.54. Galon mendapatkan mAP sebesar 0.60. Sedangkan sepatu hitam mendapatkan mAP sebesar 0.74.

Pada pengujian menggunakan model RCNN Siamese CNN *Euclidean*, kotak merah mendapatkan mAP sebesar 0.80. Tas kuning mendapatkan mAP sebesar 0.76. Galon mendapatkan mAP sebesar 0.90. Sedangkan sepatu hitam mendapatkan mAP sebesar 0.71.

Tabel 15. Hasil akurasi dengan pengujian skenario nonideal

Nama video	Nama Objek	Nama Model	<i>Threshold</i>	<i>Frame</i> terdeteksi		Akurasi
				Benar	Salah	
Ex_2_4	Kotak Merah	RCNN Siamese CNN	0.30	67	26	0.720
	Tas Kuning		0.30	53	40	0.570
	Galon		0.30	71	22	0.763
	Sepatu Hitam		0.30	56	37	0.602

Nama video	Nama Objek	Nama Model	<i>Threshold</i>	<i>Frame</i> terdeteksi		Akurasi
				Benar	Salah	
	Kotak Merah	RCNN Siamese CNN <i>Euclidean</i>	7.8	76	17	0.817
	Tas Kuning		7.8	74	19	0.796
	Galon		7.8	88	5	0.946
	Sepatu Hitam		7.8	45	48	0.484

Tabel 15 menampilkan hasil akurasi dari pengujian dengan skenario nonideal menggunakan kedua metode yang dibuat. Pada pengujian tersebut terdapat 4 objek yaitu kotak merah, tas kuning, galon, dan sepatu hitam.

Pada pengujian menggunakan model RCNN Siamese CNN, kotak merah mendapatkan akurasi sebesar 0.720. Tas kuning mendapatkan akurasi sebesar 0.570. Galon mendapatkan akurasi sebesar 0.763. Sedangkan sepatu hitam mendapatkan akurasi sebesar 0.602.

Pada pengujian menggunakan model RCNN Siamese CNN *Euclidean*, kotak merah mendapatkan akurasi sebesar 0.817. Tas kuning mendapatkan akurasi sebesar 0.796. Galon mendapatkan akurasi sebesar 0.946. Sedangkan sepatu hitam mendapatkan akurasi sebesar 0.484.

4.3 Analisis dan Evaluasi Hasil Eksperimen

Berdasarkan hasil pengujian yang telah dilakukan, hasil yang didapatkan pada setiap model yang dibangun dapat dilihat pada subbab sebelumnya. Pada subbab ini akan dilakukan analisis serta evaluasi dari hasil yang telah didapatkan.

Akan dibuat tabel perbandingan hasil mAP dan akurasi yang didapatkan pada setiap modelnya. Pada tabel ini akan digunakan satu *threshold* terbaik berdasarkan pengujian yang telah dilakukan yaitu 0.30 untuk model RCNN Siamese CNN dan 7.8 untuk model RCNN Siamese CNN *Euclidean*. Perbandingan hasil mAP dan akurasi akan ditampilkan pada Tabel 16.

Tabel 16. Perbandingan hasil mAP dan akurasi pada skenario ideal

No	Nama Video	Nama Objek	Nama Model	TH*	mAP	<i>Precision</i>	<i>Recall</i>	Akurasi
1	Ex_2_1	Kotak Merah	RCNN Siamese CNN	0.30	0.99	1.0	0.953	0.968
		Tas Kuning		0.30	0.32	0.0	0.0	0.527
		Galon		0.30	0.88	0.953	0.683	0.774
2	Ex_2_2	Kotak Merah		0.30	1.00	1.0	1.0	1
		Tas Kuning		0.30	-	-	-	-
		Galon		0.30	0.92	1.0	0.875	0.957
3	Ex_2_3	Kotak Merah		0.30	0.98	0.912	0.969	0.957
		Tas Kuning		0.30	0.44	1.0	0.166	0.728
		Sepatu Hitam		0.30	0.89	0.625	0.834	0.783
4	Ex_2_1	Kotak Merah	RCNN Siamese CNN <i>Euclidean</i>	7.8	1.00	1.0	0.984	0.989
		Tas Kuning		7.8	0.32	0.0	0.0	0.505
		Galon		7.8	0.89	1.0	0.683	0.796
5	Ex_2_2	Kotak Merah		7.8	1.00	1.0	1.0	1
		Tas Kuning		7.8	-	-	-	-
		Galon		7.8	0.92	1.0	0.875	0.957
6	Ex_2_3	Kotak Merah		7.8	1.00	1.0	1.0	1
		Tas Kuning		7.8	1.00	1.0	1.0	1
		Sepatu Hitam		7.8	0.62	1.0	0.433	0.8152

**threshold*

Berdasarkan Tabel 16, terlihat pada pendeteksian objek berupa kotak merah, model RCNN Siamese CNN *Euclidean* memiliki mAP dan akurasi yang lebih tinggi yaitu sekitar 0.1-0.2 dan 0.215-0.435 pada semua video dibandingkan dengan model RCNN Siamese CNN.

Pada pendeteksian objek berupa tas kuning, model RCNN Siamese CNN *Euclidean* memiliki mAP dan akurasi yang lebih tinggi sekitar 0.56 dan 0.272 pada video Ex_2_3 dibandingkan dengan model RCNN Siamese CNN. Sedangkan pada video Ex_2_1 dan Ex_2_2 hasil pengujian kurang tepat pada semua model. Pada video Ex_2_1, objek yang berupa tas kuning masih terdeteksi pada beberapa *frame* dan mendapatkan perbedaan akurasi sebesar 0.215 dengan model RCNN Siamese CNN mendapatkan akurasi yang lebih tinggi. Sedangkan pada video Ex_2_2, tas kuning tidak terdeteksi sama sekali sehingga tidak dapat dievaluasi mAP dan akurasinya.

Pada pendeteksian objek berupa galon, model RCNN Siamese CNN *Euclidean* dan RCNN Siamese CNN memiliki nilai mAP yang relatif sama. Hanya saja model RCNN Siamese CNN *Euclidean* memiliki nilai *precision* yang lebih tinggi yaitu sekitar 0.04651. Selain itu, model RCNN Siamese *Euclidean* juga mendapatkan akurasi yang lebih tinggi yaitu 0.215 dibandingkan dengan model RCNN Siamese CNN.

Pada pendeteksian objek berupa sepatu hitam, model RCNN Siamese CNN memiliki nilai mAP yang lebih tinggi sekitar 0.27 dibandingkan dengan model RCNN Siamese CNN *Euclidean*. Tetapi model RCNN Siamese CNN *Euclidean* dapat mendeteksi lebih tepat dibandingkan dengan model RCNN Siamese CNN, hal ini terbukti bahwa nilai *precision* untuk deteksi sepatu hitam lebih tinggi sekitar 0.375. Selain itu, model RCNN Siamese CNN *Euclidean* juga mendapatkan akurasi yang lebih tinggi yaitu 0.326 dibandingkan dengan model RCNN Siamese CNN.

Model RCNN Siamese CNN *Euclidean* dapat mendeteksi objek di mana objek yang ditemukan sudah sesuai dengan objek yang menjadi target deteksi. Sedangkan model RCNN Siamese CNN masih melakukan kesalahan dalam mendeteksi objek yang sesuai dengan objek target. Berikut contoh hasil pengujian objek yang diuji menggunakan model RCNN Siamese CNN *Euclidean* yang terlihat

pada Gambar 34 dan hasil pengujian objek yang diuji menggunakan model RCNN Siamese CNN terlihat pada Gambar 35.



Gambar 34. Beberapa contoh *frame* yang diuji menggunakan metode RCNN Siamese CNN *Euclidean*



Gambar 35. Beberapa contoh *frame* yang diuji menggunakan metode RCNN Siamese CNN

Pada Gambar 34 dan Gambar 35, objek yang terdeteksi ditandai dengan bentuk kotak berwarna putih, sedangkan objek yang tidak dapat terdeteksi ditandai dengan bentuk bulat berwarna kuning.

Berdasarkan hasil pengujian yang dilakukan pada skenario ideal, model RCNN Siamese CNN *Euclidean* mendapatkan hasil yang lebih baik dibandingkan dengan model RCNN Siamese CNN. Berikut hasil pengujian pada skenario nonideal menggunakan model dan *threshold* terbaik yang terlihat pada Tabel 17.

Tabel 17. Perbandingan hasil mAP dan akurasi pada skenario nonideal

Nama Video	Nama Objek	Nama Model	TH*	mAP	<i>Precision</i>	<i>Recall</i>	Akurasi
Ex_2_4	Kotak Merah	RCNN Siamese CNN	0.30	0.66	0.763	0.630	0.720
	Tas Kuning		0.30	0.54	1.0	0.111	0.570
	Galon		0.30	0.60	0.6	1.0	0.763
	Sepatu Hitam		0.30	0.71	0.727	0.645	0.602
	Kotak Merah	RCNN Siamese CNN <i>Euclidean</i>	7.8	0.80	0.968	0.652	0.817
	Tas Kuning		7.8	0.76	0.933	0.622	0.796
	Galon		7.8	0.90	1.0	0.849	0.946
	Sepatu Hitam		7.8	0.74	1.0	0.226	0.484

**threshold*

Pada Tabel 17 terlihat hasil pengujian skenario nonideal menggunakan model RCNN Siamese CNN *Euclidean* pada pendeteksian objek berupa kotak merah mendapatkan hasil yang lebih tinggi dengan mAP 0.80 dan akurasi 0.817 dibandingkan dengan model RCNN Siamese CNN.






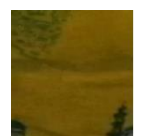




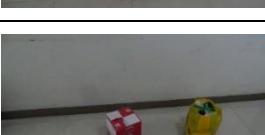


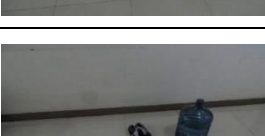
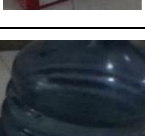

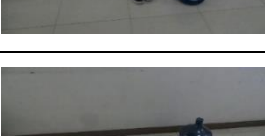



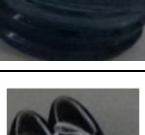
Pada pendeteksian objek tas kuning, model RCNN Siamese CNN *Euclidean* mendapatkan hasil yang lebih tinggi dengan mAP 0.76 dan akurasi 0.796 dibandingkan dengan model RCNN Siamese CNN.


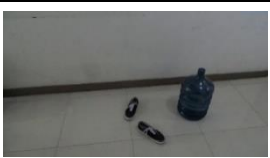
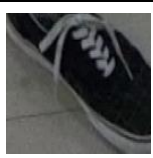

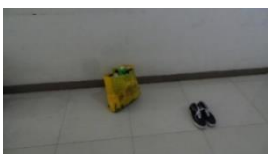


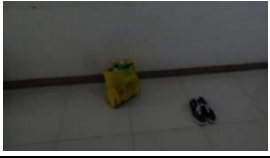

Pada pendeteksian objek galon, model RCNN Siamese CNN *Euclidean* mendapatkan hasil yang lebih tinggi dengan mAP 0.90 dan akurasi 0.946 dibandingkan dengan model RCNN Siamese CNN.

Pada pendeteksian objek sepatu hitam, model RCNN Siamese CNN *Euclidean* mendapatkan hasil dengan mAP 0.74 dan akurasi 0.484. Sedangkan pada model RCNN Siamese CNN didapatkan mAP 0.71 dan akurasi sebesar 0.602.

Model RCNN Siamese CNN *Euclidean* mendapatkan hasil yang cukup memuaskan pada kedua skenario yang telah dilakukan. Akan tetapi berdasarkan hasil pengujian yang telah dilakukan pada skenario nonideal, model RCNN Siamese CNN *Euclidean* masih belum mampu mendeteksi pada beberapa *frame* tertentu. Berikut beberapa sampel hasil pendeteksian menggunakan RCNN Siamese CNN *Euclidean* dengan *threshold* 7.8 ditunjukkan pada tabel sekian.

Tabel 18. Sampel hasil pengujian pada skenario nonideal dengan model terbaik

No	Gambar target	Frame	Gambar objek terdeteksi	Skor	Ket
1				7.58274	Terdeteksi
2				8.18134	Tidak terdeteksi
3				6.02195	Terdeteksi
4				7.18990	Terdeteksi
5				6.07862	Terdeteksi
6				6.49917	Terdeteksi
7				7.57301	Terdeteksi

8				11.4024	Tidak terdeteksi
9				9.89636	Tidak terdeteksi
10				9.50071	Tidak terdeteksi

Pada Tabel 18, skor prediksi yang ditandai warna biru merupakan prediksi yang tepat dengan pernyataan aslinya. Sedangkan, yang ditandai warna merah merupakan prediksi yang tidak tepat dengan pernyataan aslinya. Dari data tersebut terlihat bahwa, model yang dibangun masih melakukan kesalahan dalam mendeteksi objek pada video yang memiliki perubahan yang signifikan terhadap bentuk objek yang ditangkap oleh kamera dibandingkan dengan gambar objek yang menjadi target. Hal tersebut terlihat pada gambar 8 hingga 9 pada Tabel 18. Selain itu, video yang memiliki pencahayaan yang rendah pun juga dapat mempengaruhi proses pendeteksian yang terlihat pada gambar 10 pada Tabel 18.

Berdasarkan penjabaran keseluruhan hasil pengujian kedua model diatas, dapat disimpulkan bahwa model RCNN Siamese CNN *Euclidean* mendapatkan mAP dan akurasi yang lebih tinggi dibandingkan dengan model RCNN Siamese CNN, di mana hasil tersebut dapat memperlihatkan kinerja model yang dapat mendeteksi lebih tepat dan akurat. Model RCNN Siamese CNN *Euclidean* juga mampu melakukan pelatihan dengan waktu yang lebih cepat, sehingga model ini cocok digunakan pada platform yang membutuhkan efisiensi waktu lebih tinggi dalam penggunaannya. Namun walaupun sudah lebih bagus, model RCNN Siamese CNN *Euclidean* masih melakukan kesalahan jika terdapat perubahan yang signifikan terhadap bentuk objek yang ditangkap oleh kamera dibandingkan dengan gambar objek target. Model ini juga dapat melakukan kesalahan jika pencahayaan

pada video rendah. Selain itu, proses awal pendeteksian menggunakan metode RPN juga mempengaruhi hasil dari pendeteksian objek.