

# Deteksi *Spoofing* Wajah Menggunakan *Faster R-CNN* dengan Arsitektur Resnet50 pada Video

## (*Face Spoofing Detection Using Faster R-CNN with Resnet50 Architecture on Video*)

Sunario Megawan<sup>1</sup>, \*Wulan Sri Lestari<sup>1</sup>

**Abstract**—Face detection is a main and important process in the field of face recognition that has been widely studied. The purpose of face detection is to determine the presence and mark the position of faces, in both images and videos, called bounding boxes. One important problem in face detection is to differentiate between face spoof and non-spoof which is referred to as face spoofing detection. Face spoofing detection is an important task used to ensure the security of face-based authentication and facial analysis systems. Therefore, we need a model that can detect face spoofing. In this paper, the process to build a model that can be used to detect face spoofing on video is carried out using Faster R-CNN with Resnet50 architecture. Faster R-CNN is one of the superior algorithms in solving various object detection problems. The dataset used in this paper is a Replay-Attack Database, provided by Idiap Dataset Distribution Portal. The training phase used 360 videos, consisting of 300 spoof videos and 60 non-spoof videos. The average accuracy of the training stage is 97,07% with a total of 21 epochs. The test results show that the resulting model successfully determined bounding boxes and detected face spoof and non-spoof on the video effectively.

**Intisari**—Deteksi wajah merupakan proses mendasar dan penting dalam bidang pengenalan wajah yang sudah diteliti secara luas. Tujuan deteksi wajah adalah menentukan keberadaan dan menandai posisi wajah, baik pada gambar maupun video, yang disebut dengan *bounding box*. Salah satu masalah penting dalam deteksi wajah adalah membedakan wajah *spoof* dan *non-spoof* yang disebut sebagai deteksi *spoofing* wajah. Deteksi *spoofing* wajah merupakan pekerjaan penting yang digunakan untuk menjamin keamanan autentikasi berbasis wajah dan sistem analisis wajah. Oleh karena itu, dibutuhkan sebuah model yang dapat mendeteksi *spoofing* wajah. Pada makalah ini dilakukan proses membangun model yang dapat digunakan untuk mendeteksi wajah *spoof* dan *non-spoof* pada video menggunakan algoritme *Faster R-CNN* dengan arsitektur Resnet50. *Faster R-CNN* merupakan salah satu algoritme yang unggul dalam menyelesaikan berbagai persoalan deteksi objek. *Dataset* yang digunakan adalah *Replay-Attack Database* yang disediakan oleh *Idiap Dataset Distribution Portal*. Pada tahap *training* digunakan 360 video *spoof* dan *non-spoof*. Rata-rata nilai akurasi yang dihasilkan pada tahap *training* adalah 97,07%, dengan jumlah *epoch* sebanyak 21. Hasil pengujian menunjukkan bahwa model yang dihasilkan berhasil menentukan *bounding box*

dengan akurat dan mendeteksi *spoof* dan *non-spoof* wajah pada video dengan efektif.

**Kata Kunci**—Deteksi Wajah, *Spoofing*, Video, *Faster R-CNN*, Resnet50.

### I. PENDAHULUAN

Deteksi wajah merupakan tahap penting dalam banyak aplikasi, seperti *face recognition*, *facial expression*, dan *emotion recognition* [1]. Tujuan deteksi wajah adalah menentukan keberadaan wajah dalam video dan jika ditemukan, maka posisi wajah ditandai. Ada banyak faktor yang memengaruhi deteksi wajah, yaitu kualitas video, posisi wajah manusia, pose ekstrem berbagai skala wajah, dan *occlusions* [2]. Sebagai proses penting dalam *face recognition*, algoritme untuk deteksi wajah yang baik diharapkan mampu mengidentifikasi wajah dalam kondisi dan kualitas video yang berubah-ubah.

Banyak algoritme yang telah diusulkan untuk melakukan proses deteksi wajah pada video. Referensi [3] menggunakan *Convolutional Neural Network* (CNN) untuk melakukan proses deteksi wajah. Hasil yang diperoleh menunjukkan bahwa algoritme yang digunakan mampu untuk mendeteksi wajah. Namun, penelitian tersebut hanya difokuskan pada data citra saja. Penelitian lainnya mengusulkan algoritme *Adaboost* untuk melakukan deteksi wajah pada *Youtube Faces dataset* [4]. Penelitian tersebut menggunakan video wajah *full face* dan tampak depan, sehingga mudah untuk dideteksi. Selain itu, penelitian tersebut juga hanya dapat digunakan untuk mendeteksi satu wajah saja. Penelitian lainnya menggunakan *Local Binary Pattern* untuk melakukan deteksi wajah pada video *realtime* yang ditangkap oleh kamera [5]. Namun, penelitian tersebut juga hanya dapat digunakan untuk mendeteksi satu wajah saja.

Penelitian lainnya melakukan deteksi wajah dan pejalan kaki pada video menggunakan *Region based Convolutional Neural Network* (R-CNN) [6]. R-CNN merupakan pengembangan dari CNN yang digunakan untuk menyelesaikan berbagai persoalan deteksi objek. Hasil penelitian menunjukkan bahwa algoritme yang diusulkan mampu mendeteksi banyak wajah dan para pejalan kaki [6]. Namun, penelitian tersebut memiliki kelemahan, yaitu belum dapat membedakan wajah asli (*non-spoof*) dan gambar atau video wajah (*spoof*), yang disebut dengan *spoofing* wajah. *Spoofing* wajah merupakan proses pemalsuan wajah seseorang untuk mendapatkan akses ilegal ke sistem biometrik yang dapat dilakukan dengan menampilkan video atau gambar wajah seseorang di layar monitor. Untuk

<sup>1</sup>Program Studi Teknik Informatika, STMIK Mikroskil, Jl. Thamrin No. 140 Medan 20212 INDONESIA (tlp: 061-4573767; fax: 061-4567789; e-mail: sunario@mikroskil.ac.id, wulan.lestari@mikroskil.ac.id)

mengatasi kelemahan tersebut, pada makalah ini diusulkan algoritme *Faster Region based Convolutional Neural Network* (*Faster R-CNN*). *Faster R-CNN* merupakan algoritme deteksi objek berbasis wilayah terbaru yang menunjukkan hasil luar biasa pada berbagai deteksi objek [7]. Ada banyak penelitian yang sudah menggunakan *Faster R-CNN* pada bidang deteksi objek, seperti deteksi objek pada gambar [7] dan deteksi mobil pada video [8]. Kinerja *Faster R-CNN* sangat bergantung pada kualitas data *training* yang besar. Tujuan utama makalah ini adalah menguji kinerja *Faster R-CNN* dalam mendeteksi wajah pada video yang dapat membedakan antara wajah *spoof* dan *non-spoof* pada *dataset Replay-Attack Database* yang disediakan oleh *Idiap Dataset Distribution Portal*.

## II. DETEKSI WAJAH

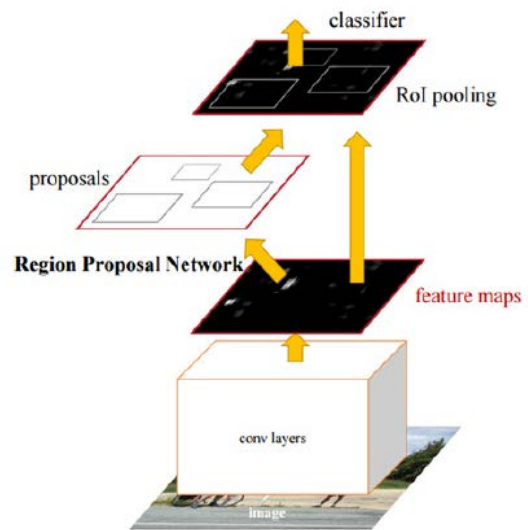
Deteksi wajah merupakan masalah mendasar dan penting dalam *computer vision* dan pengenalan pola yang telah diteliti secara luas. Deteksi wajah merupakan tahap penting dalam banyak aplikasi, seperti *face recognition*, *facial expression*, dan *emotion recognition* [1]. Tujuan deteksi wajah adalah menentukan keberadaan posisi wajah dalam video dan jika wajah ditemukan maka posisi wajah ditandai dengan *bounding box*. Ada banyak faktor yang memengaruhi deteksi wajah, yaitu kualitas video, posisi wajah manusia, pose ekstrem berbagai skala wajah dan *occlusions* [2]. Beberapa tantangan yang dihadapi pada proses deteksi wajah adalah sebagai berikut [9].

- Posisi wajah pada citra atau video dapat bervariasi seperti miring, tegak, hadap depan, hadap belakang, tunduk, dan lain-lain.
- Komponen-komponen pada wajah seperti kacamata, jenggot, dan kumis.
- Ekpresi wajah.
- Posisi wajah yang terhalang objek lain.
- Kondisi pengambilan citra atau video yang dipengaruhi oleh intensitas cahaya ruangan, lensa kamera, dan lain-lain.

## III. FASTER R-CNN

*Region based Convolutional Neural Networks* (R-CNN) merupakan algoritme deteksi objek yang masuk dalam ranah *computer vision* berbasis jaringan konvolusi. R-CNN sendiri awalnya pada tahun 2015 dibuat sebagai algoritme pendeteksian objek yang mengombinasikan algoritme *Region Proposal Network* (RPN) dan CNN [10]. Seiring berjalannya waktu, R-CNN terus dikembangkan untuk meningkatkan kinerja, baik kecepatan maupun akurasi, dalam pendeteksian objek. Akan tetapi, R-CNN maupun *Fast R-CNN* masih mempunyai kekurangan, salah satunya adalah adanya *bottleneck*, yaitu kompleksitas perhitungan *region proposal* pada RPN yang tidak dapat menyamai kecepatan komputasi pada CNN. Untuk mengatasi masalah *bottleneck*, dilakukan penelitian dengan optimisasi pada penggunaan fitur konvolusional untuk mempercepat proses RPN, sehingga dapat mengurangi terjadinya *bottleneck* menggunakan *Faster R-CNN*, yang merupakan kombinasi dari *Fast R-CNN* dan RPN [7].

*Faster R-CNN* merupakan algoritme deteksi objek berbasis wilayah terbaru yang menunjukkan hasil luar biasa pada



Gbr. 1 Arsitektur *Faster R-CNN*.

berbagai deteksi objek [7]. *Faster R-CNN* adalah algoritme yang menggunakan *Fast R-CNN* dan RPN sebagai arsitektur utamanya. Algoritme ini merupakan pengembangan dari *Fast R-CNN* dengan mengubah bagian *selective search* pada *Fast R-CNN* menjadi RPN [7]. RPN adalah sebuah *neural network* yang menggantikan peran *selective search* untuk mengajukan *region* (bagian-bagian dari sebuah gambar yang perlu “dilihat” lebih jauh). RPN menghasilkan beberapa *bounding box*, dengan setiap *box* memiliki dua skor probabilitas, yaitu pada lokasi tersebut terdapat objek atau tidak. *Region-region* yang dihasilkan tersebut akan menjadi *input* untuk arsitektur yang mirip seperti *Fast R-CNN*. Penggunaan RPN untuk menggantikan *selective search* dapat mengurangi kebutuhan komputasi yang cukup signifikan dan membuat keseluruhan model dapat dilatih secara *end-to-end*. Saat ini, *Faster R-CNN* sudah menjadi pilihan model yang umum digunakan untuk deteksi objek dengan solusi berbasis *deep learning* [7], [8], [11]. Gbr. 1 merupakan arsitektur dari *Faster R-CNN* [12]. *Faster R-CNN* dibagi menjadi dua bagian penting sebagai berikut [12].

1) *Region Proposal Network (RPN)*: RPN merupakan proses yang berguna untuk mencari kemungkinan lokasi objek pada gambar yang dimasukkan secara cepat. Lokasi objek yang ada pada gambar memiliki kemungkinan batasan objek dari wilayah yang diketahui yang disebut sebagai *Region of Interest (RoI)*. RoI diperkenalkan oleh Ross Girshick tahun 2015 [10] sebagai sebuah pendekatan deteksi objek menggunakan *deep learning*. *Input* yang digunakan pada *layer RoI* adalah *feature maps* yang merupakan *output* dari CNN dengan *multiple convolution layers* dan *max pooling layers*. Sebuah  $N \times N$  matriks dibuat dengan membagi ruang *feature map* menjadi RoI. Kolom pertama merupakan indeks dari citra dan sisa kolom lainnya merupakan koordinat dari RoI, dimulai dari koordinat paling kiri atas hingga kiri bawah. RoI yang sudah ditentukan disebut sebagai *region proposal*. Pada RPN, awalnya citra *input* diproses dalam jaringan saraf konvolusi untuk menghasilkan *feature map*. *Feature map* terdiri atas

TABEL I  
RINCIAN DATA YANG DIGUNAKAN

Jenis	Kelas		Jumlah Video	Jumlah Frame/Video
	Spoof	Real		
Training	Spoof	Fixed	150	240
		Hand	150	230
	Non-Spoof	Real	60	375
Testing	Spoof	Fixed	200	240
		Hand	200	230
	Non-Spoof	Real	80	375

enam bagian, yaitu penentuan *object* dan *non-object* dengan nilai 0–1, koordinat nilai *x* dan *y*, serta nilai *weight* dan *height* dari *bounding box*. *Sliding window* ditempatkan pada setiap *feature map* dengan ukuran  $N \times N$ , sesuai dengan setiap *anchor sliding window* yang dibentuk. Setiap *anchor* memiliki titik pusat yang sama, tetapi memiliki *aspect ratios* dan *scaling factor* yang berbeda.

2) *Classifier*: *Classifier* merupakan proses yang digunakan untuk mengklasifikasikan RoI yang sudah diidentifikasi pada RPN ke dalam kelas atau target yang sesuai. Teknik yang digunakan pada tahap ini adalah CNN.

IV. METODOLOGI

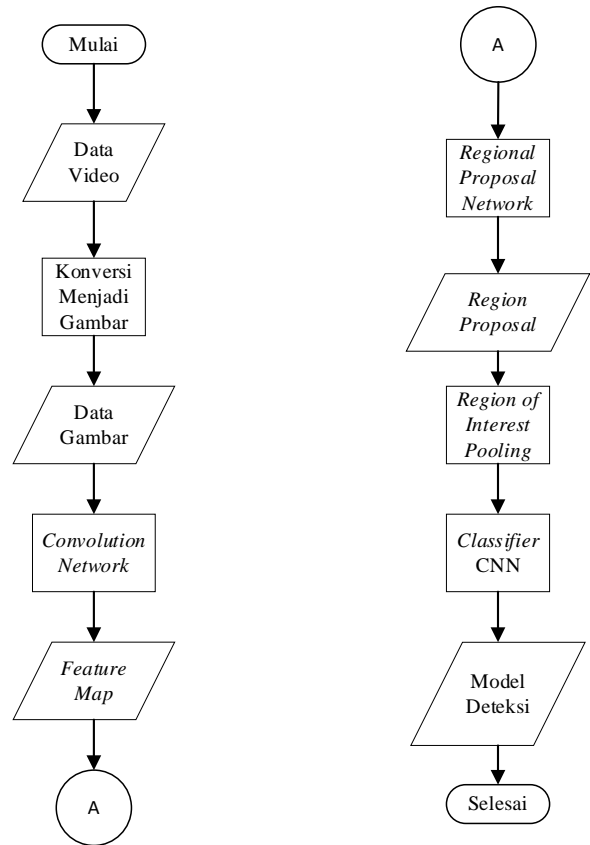
Tahapan-tahapan yang dilakukan dalam makalah ini adalah sebagai berikut.

A. Pengumpulan Data

Data yang digunakan adalah data sekunder (*dataset*), yaitu *Replay-Attack Database* yang dapat diakses pada link <https://www.idiap.ch/dataset/replayattack>. *Replay-Attack Database* merupakan *dataset* yang digunakan pada berbagai penelitian di bidang deteksi *spoofing* wajah. *Replay-Attack Database* berisi 1.300 video klip dari foto dan video yang direkam dengan berbagai *lighting* yang berbeda dari 50 klien [13]. *Dataset* dibagi menjadi empat bagian, yaitu sebagai berikut.

1. *Training data* (“train”) digunakan untuk *training* model klasifikasi *non-spoof* yang terdiri atas 60 *real-accesses* dan 300 *attacks* di bawah kondisi *lighting* yang berbeda.
2. *Development data* (“devel”) digunakan sebagai *threshold estimation* (EER – *Equal Error Rate point*), yang terdiri atas 60 *real-accesses* dan 300 *attacks* di bawah kondisi *lighting* yang berbeda.
3. *Test data* (“test”) digunakan untuk menghasilkan *error/akurasi* dari model yang diusulkan, yang terdiri atas 80 *real-accesses* dan 400 *attacks* di bawah kondisi *lighting* yang berbeda.
4. *Enrollment data* (“enroll”) digunakan untuk melakukan verifikasi sensitivitas *spoofing* pada algoritme deteksi wajah, yang terdiri atas 100 *real-accesses* di bawah kondisi *lighting* yang berbeda.

Pada makalah ini, data yang digunakan untuk membangun model deteksi *spoofing* wajah pada video hanya menggunakan



Gbr. 2 Flowchart tahap *training*.

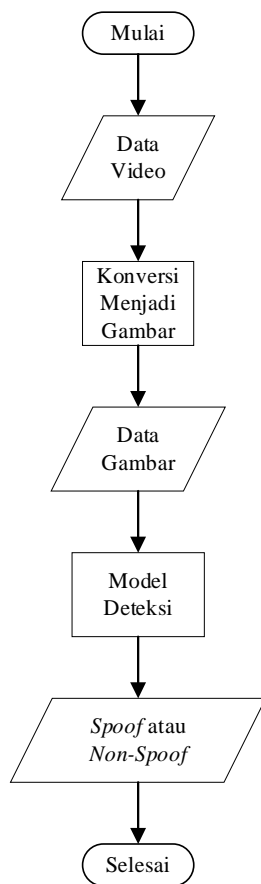
*training data* dan *testing data*. Tabel I merupakan rincian data yang digunakan.

B. Implementasi *Faster R-CNN*

Pada tahap ini dilakukan proses analisis dan desain terhadap algoritme *Faster R-CNN* untuk diimplementasikan pada sistem deteksi wajah manusia pada video, yang dapat digunakan untuk membedakan wajah *spoof* dan *non-spoof* berdasarkan *dataset* yang digunakan. Alat bantu yang digunakan adalah bahasa pemrograman Python, Tensorflow, Keras, dan OpenCV. Penelitian terdiri atas dua proses penting, yaitu tahap *training* dan tahap *testing*.

1) *Tahap Training*: Tahap *training* digunakan untuk membangun model berdasarkan algoritme *Faster R-CNN* dan *dataset Replay-Attack*. Gbr. 2 merupakan langkah-langkah yang dilakukan pada tahap *training*. Berikut adalah penjelasan masing-masing tahap pada Gbr. 2.

- Masukan berupa data video.
- Seluruh data video dikonversi menjadi data gambar menggunakan OpenCV. Kemudian, masing-masing gambar diberi label sesuai dengan jenis kelasnya masing-masing (*spoof* dan *non-spoof*) beserta ukuran dari *bounding box* yang akan digunakan untuk proses deteksi dan disimpan ke dalam *file* teks untuk mempermudah proses pemanggilan data.



Gbr. 3 Flowchart tahap testing.

- Seluruh data diproses ke dalam *layer convolution network* dengan nilai *weight* yang diinisialisasi menggunakan Resnet50 dan menghasilkan *feature map* berdasarkan jumlah *anchor*.
- Selanjutnya, *feature map* diproses ke *regional proposal network* untuk mendapatkan *region proposal*.
- *Region proposal* selanjutnya akan diproses ke dalam ROI dan *classifier CNN* untuk menentukan jenis kelas (label) serta akurasi dalam pembuatan model berdasarkan jumlah *epoch* yang digunakan.

2) *Tahap Testing*: Tahap *testing* digunakan untuk menguji model yang dihasilkan pada tahap *training*. Gbr. 3 merupakan *flowchart* tahap *testing*, dengan penjelasan masing-masing tahap adalah sebagai berikut.

- Masukan berupa data video.
- Seluruh data video dikonversi menjadi data gambar menggunakan OpenCV.
- Seluruh data gambar dimasukkan ke dalam model yang sudah dibangun pada tahap *training*.
- Kemudian video akan dideteksi, termasuk wajah *spoof* atau *non-spoof*, beserta dengan nilai akurasi yang dihasilkan.

### C. Confusion Matrix

*Confusion matrix* digunakan untuk mengukur kinerja algoritme yang diimplementasikan. Kinerja yang diukur dilihat

TABEL II  
CONFUSION MATRIX

		Correct Result/Classification	
		E1	E2
Obtained Result/Classification	E1	TP (True Positive)	FP (False Positive)
	E2	FN (False Negative)	TN (True Negative)

TABEL III  
SPESIFIKASI HARDWARE DAN SOFTWARE

Kriteria	Spesifikasi
Sistem Operasi	Windows 10 64-Bit
Prosesor	Amd Ryzen 3700x
Memori	32Gb DDR4 PC 3200
Graphics Card	GTX 1660 TI 6GB DDR6
GPU Engine	NVIDIA Cuda Core 1536
Bahasa Pemrograman	Python
IDE Pemrograman	IDLE
Library	Tensorflow, Keras, dan OpenCV

dari sisi tingkat akurasi. Tabel II merupakan *confusion matrix* yang sering digunakan.

$$Akurasi = \frac{TP+TN}{TP+TN+FN+FP} \times 100\% \quad (1)$$

dengan *TP* adalah *True Positive*, yaitu jumlah data positif yang terklasifikasi dengan benar oleh sistem; *TN* adalah *True Negative*, yaitu jumlah data negatif yang terklasifikasi dengan benar oleh sistem; *FN* adalah *False Negative*, yaitu jumlah data negatif tetapi terklasifikasi salah oleh sistem; dan *FP* adalah *False Positive*, yaitu jumlah data positif tetapi terklasifikasi salah oleh sistem.

## V. HASIL DAN PEMBAHASAN

Untuk mengimplementasi teknik *Faster R-CNN* dengan data video, dibutuhkan spesifikasi perangkat yang mendukung agar proses *training* dan *testing* dapat berjalan dengan baik dan lebih efisien. Tabel III menyajikan spesifikasi *hardware* dan *software* yang digunakan.

Pada makalah ini, untuk membangun model deteksi menggunakan *Faster R-CNN*, digunakan 360 video (300 *spoof* dan 60 *non-spoof*) data *training*. Kemudian, video tersebut dikonversi menjadi gambar dengan total 92.998, yang terdiri atas 22.461 gambar *non-spoof* dan 69.996 gambar *spoof*. Sebelum proses *training* dilakukan, disediakan sebuah file teks yang berisi informasi terkait nama gambar, ukuran *bounding box*, dan *class* (label). Data *training* tersebut dibagi menjadi dua, yaitu data *train* sebanyak 76.990 gambar dan data *validation* sebanyak 15.467 gambar. Arsitektur CNN yang digunakan adalah Resnet50 yang merupakan model yang sudah dilatih menggunakan *ImageNet Dataset* agar dapat menghasilkan *feature extraction* yang baik. Jumlah *anchor* yang digunakan adalah sembilan, sesuai dengan *default anchor Faster R-CNN*. *Anchor* merupakan bagian penting yang digunakan untuk menentukan bagian penting dari gambar

TABEL IV  
HASIL PROSES TRAINING

Epoch ke-	Loss RPN Classifier (mse)	Loss RPN Regression (mse)	Loss Detector Classifier (mse)	Loss Detector Regression (mse)	Akurasi	Waktu (H:M:S)
1	0,308	0,21	0,13	0,06	94,92	07:25:00
2	0,213	0,009	0,085	0,026	96,55	07:14:57
3	0,197	0,007	0,084	0,022	96,62	07:13:25
4	0,202	0,005	0,080	0,020	96,84	07:18:11
5	0,113	0,005	0,082	0,019	96,76	07:15:10
6	0,011	0,004	0,082	0,018	96,80	07:15:21
7	0,009	0,004	0,078	0,017	96,97	07:22:05
8	0,007	0,004	0,075	0,016	97,04	07:17:07
9	0,008	0,003	0,074	0,015	97,09	07:16:29
10	0,006	0,003	0,071	0,014	97,25	07:21:27
11	0,007	0,003	0,072	0,014	97,17	07:19:45
12	0,006	0,003	0,071	0,014	97,22	12:07:49
13	0,007	0,003	0,068	0,013	97,38	07:20:22
14	0,006	0,003	0,069	0,013	97,33	07:19:13
15	0,006	0,002	0,067	0,013	97,39	07:17:37
16	0,005	0,002	0,066	0,013	97,44	07:19:16
17	0,004	0,002	0,067	0,012	97,40	07:18:50
18	0,004	0,002	0,064	0,012	97,54	07:17:48
19	0,003	0,002	0,062	0,012	97,58	07:21:57
20	0,002	0,002	0,062	0,012	97,58	07:22:08
21	0,003	0,002	0,062	0,011	<b>97,63</b>	07:21:22



Gbr. 4 Hasil deteksi wajah tahap testing.

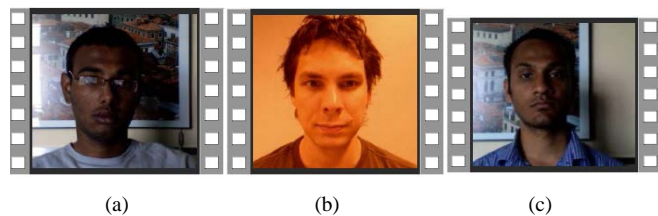
(region proposal) yang akan dimasukkan ke RPN. Optimizer yang digunakan adalah Adam, dengan nilai learning rate sebesar 0,00001. Selain itu, Stochastic Gradient Distance digunakan untuk mengoptimasi convolution layer, RPN weight, dan fully connected layer. Epoch length yang digunakan adalah 50.000, dengan jumlah epoch sebesar 21.

Tabel IV menunjukkan hasil yang diperoleh pada tahap training. Berdasarkan tabel tersebut, nilai akurasi tertinggi diperoleh pada epoch ke-21 dengan nilai 97,63% , nilai loss (tingkat kesalahan) yang semakin rendah, dan waktu eksekusi 7 jam 21 menit 22 detik. Pada saat pembuatan model di tahap training, nilai weight akan diperbarui setiap kali nilai akurasi pada epoch berjalan lebih besar daripada nilai akurasi pada epoch sebelumnya. Sementara itu, Gbr. 4 merupakan hasil deteksi wajah yang dilakukan pada tahap testing. Lalu, untuk menguji model training yang dihasilkan, dilakukan beberapa skenario pengujian sebagai berikut.

1) Pengaruh Lighting Video Terhadap Akurasi Deteksi yang Dihasilkan: Untuk mengetahui pengaruh lighting



Gbr. 5 Video uji spoof.



Gbr. 6 Video uji non-spoof.

terhadap akurasi model deteksi, pada pengujian ini digunakan enam data video yang terdiri atas tiga video spoof dan tiga video non-spoof. Gbr. 5 merupakan video spoof sedangkan Gbr. 6 merupakan video non-spoof.

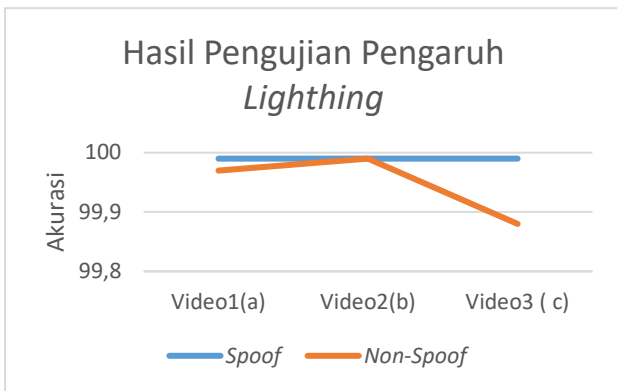
Hasil pengujian untuk melihat pengaruh lighting terhadap hasil akurasi deteksi spoof dan non-spoof diperlihatkan pada Tabel V. Berdasarkan tabel, dibuat grafik yang diperlihatkan pada Gbr. 7. Dapat dilihat bahwa lighthing dapat memengaruhi keakuratan model dalam mendeteksi wajah spoof dan non-spoof. Video1 (a) Spoof dengan jumlah frame 230 mendapatkan rata-rata akurasi 89,04% dengan lighthing video gelap. Sementara untuk video1 (a) non-spoof dengan jumlah frame 375, dihasilkan nilai rata-rata akurasi yang lebih rendah, yaitu 56,48% dengan lighthing video yang diberikan juga gelap. Tabel V juga menunjukkan bahwa waktu yang dibutuhkan

TABEL V  
HASIL PENGUJIAN PENGARUH LIGHTING

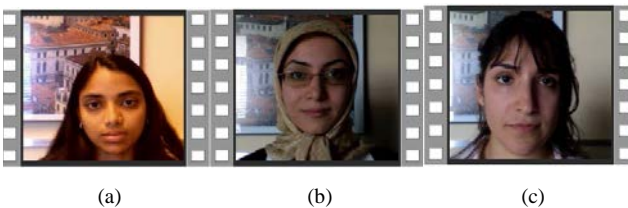
Kelas	Jumlah Frame	Hasil Deteksi	Rata-rata Akurasi (%)	Waktu (Menit:Detik)
<b>Spoof</b>				
Video1 (a)	230	Spoof	89,04	36:23
Video2 (b)	230	Spoof	99,99	38:47
Video3 (c)	230	Spoof	99,99	37:09
<b>Non-Spoof</b>				
Video1 (a)	375	Non-Spoof	56,48	56:17
Video2 (b)	375	Non-Spoof	99,92	58:09
Video3 (c)	375	Non-Spoof	99,94	59:16

TABEL VI  
HASIL PENGUJIAN PENGARUH GENDER

Kelas	Jumlah Frame	Hasil Deteksi	Rata-rata Akurasi	Waktu (Jam:Menit:Detik)
<b>Spoof</b>				
Video1 (a)	230	Spoof	99,99	00:35:00
Video2 (b)	230	Spoof	99,99	00:38:13
Video3 (c)	240	Spoof	99,99	00:37:04
<b>Non-Spoof</b>				
Video1 (a)	375	Spoof	99,97	01:01:27
Video2 (b)	375	Non-Spoof	99,99	01:32:16
Video3 (c)	375	Spoof	99,88	01:01:52



Gbr. 7 Grafik hasil pengujian pengaruh lighting.



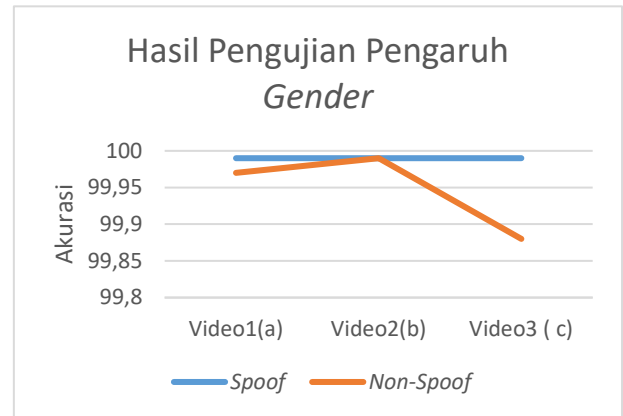
Gbr. 8 Data video wanita spoof.



Gbr. 9 Data video wanita non-spoof.

untuk menguji masing-masing video berbeda-beda, meskipun jumlah frame yang diberikan sama.

2) Pengaruh Gender Terhadap Akurasi Deteksi yang Dihasilkan: Pengujian ini dilakukan dengan tujuan mengetahui gender memengaruhi hasil deteksi atau tidak. Pada tahap training, data yang digunakan untuk membangun model merupakan 94% data video dengan gender laki-laki, sehingga perlu diuji dengan data video wanita untuk melihat keakuratan model dalam melakukan deteksi spoof dan non-spoof wajah.



Gbr. 10 Grafik hasil pengujian pengaruh gender.

Gbr. 8 menunjukkan tiga data video wanita spoof dan Gbr. 9 menunjukkan data video wanita non-spoof yang digunakan untuk pengujian. Hasil pengujian untuk melihat pengaruh gender terhadap hasil akurasi deteksi spoof dan non-spoof diperlihatkan pada Tabel VI.

Berdasarkan Tabel VI, dibuat grafik yang diperlihatkan pada Gbr. 10. Dapat dilihat bahwa video wanita wajah spoof dapat terdeteksi benar sesuai jenis kelasnya dengan rata-rata nilai akurasi 99,99%. Sementara, data video uji wanita untuk wajah non-spoof hanya berhasil dideteksi dengan benar pada video1 (a) dengan nilai akurasi 99,97%, sedangkan video2 (b) dan video3 (b) non-spoof hanya dapat dideteksi bagian wajahnya, tetapi hasil klasifikasi kelasnya tidak sesuai. Oleh karena itu, untuk pengujian pengaruh gender dari enam video data uji yang diberikan, model hanya dapat mendeteksi empat video dengan benar. Hal tersebut terjadi karena pada saat melakukan training, jumlah video wanita yang diuji tidak termasuk ke dalam data training.

VI. KESIMPULAN

Berdasarkan hasil pengujian yang didapatkan dengan berbagai skenario, diketahui bahwa lighting dapat memengaruhi hasil akurasi deteksi spoofing wajah, tetapi hasil deteksi yang dilakukan benar sesuai dengan jenis kelasnya. Model yang dihasilkan juga belum mampu mendeteksi dengan baik ketika data uji yang diberikan merupakan video wajah wanita. Hal tersebut dapat dilihat pada pengujian pengaruh

gender, yaitu model hanya dapat mendeteksi empat video dengan baik dari enam video yang diberikan. Namun, secara keseluruhan, *Faster R-CNN* dapat mendeteksi wajah dan membedakan wajah *spoof* dan *non-spoof* dengan rata-rata akurasi *training* sebesar 97,07%. Untuk penelitian selanjutnya, model yang dihasilkan perlu diuji dengan beberapa skenario lain, seperti *dataset* yang lebih kompleks, pengaruh *noise*, *dataset* lainnya yang dapat menguji model yang dihasilkan dapat mendeteksi *spoofing* dan *non-spoofing* wajah dengan akurat dari sisi samping dan jarak wajah yang lebih jauh atau tidak.

#### UCAPAN TERIMA KASIH

Terima kasih disampaikan kepada STMIK Mikroskil yang telah memberikan bantuan hibah dan dukungan pada penelitian ini melalui skema Hibah Kompetisi Internal Kreativitas dan Inovasi Dosen Semester Ganjil TA. 2019/2020.

#### REFERENSI

- [1] E. Owusu, J.D. Abdulai, dan Y. Zhan, "Face Detection Based on Multilayer Feed-forward Neural Network and Haar Features," *Softw. Pract. Exper.*, Vol. 49, No. 1, hal. 120-129, 2019.
- [2] Y. Zhou, D. Liu, dan T. Huang, "Survey of Face Detection on Low-quality Images," *13<sup>th</sup> IEEE International Conference on Automatic Face & Gesture Recognition (FG)*, 2018, hal. 769-773.
- [3] G. Guo, H Wang, Y. Yan, J Zheng, dan B. Li, "A Fast Face Detection Method via Convolutional Neural Network," *Neurocomputing*, Vol. 395, hal, 128-137, 2020.
- [4] X. Huang, "A Hybrid Framework for Human Face Detection and Recognition in Videos," *IOP Conf. Series: Earth and Environmental Science*, Vol. 252, No. 5, hal. 1-5, 2018.
- [5] N.A. Othman dan I. Aydin, "A Face Recognition Method in the Internet of Things for Security Applications in Smart Homes and Cities," *6th International Istanbul Smart Grids and Cities Congress and Fair (ICSG)*, 2018, hal 20-24.
- [6] S.Y. Jin, A. RoyChowdhury, H. Jiang, A. Singh, A. Prasad, D. Chakraborty, dan E.L. Miller, "Unsupervised Hard Example Mining from Videos for Improved Object Detection," *European Conference on Computer Vision*, 2018, hal. 1-18.
- [7] S. Ren, K. He, R.B. Girshick, dan J. Sun, "Faster R-CNN: Towards Real-time Object Detection with Region Proposal Networks," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 39, No. 6, hal. 1137-1149, 2017.
- [8] B. Benjdira, T. Khursheed, A. Koubaa, A. Ammar, dan K. Ouni, "Car Detection using Unmanned Aerial Vehicles: Comparison between Faster R-CNN and YOLOv3," *1st International Conference on Unmanned Vehicle Systems-Oman (UVS)*, 2019, hal. 1-6.
- [9] M.D. Putro, T.B. Adji, dan B. Winduratna, "Sistem Deteksi Wajah dengan Menggunakan Metode Viola-Jones," *Sem. Nas. Sci. Eng. Technol.*, 2012, hal. 1-5.
- [10] R. Girshick, J. Donahue, T. Darrell, dan J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *IEEE Conf. on Computer Vision and Pattern Recognition*, 2014, hal. 580-587.
- [11] Y. Liu, Z. Ma, X. Liu, S. Ma, dan K. Ren, "Privacy-Preserving Object Detection for Medical Images with Faster R-CNN," *IEEE Transactions on Information Forensics and Security*, hal. 1-16, 2019.
- [12] S.M. Abbas dan S.N. Singh, "Region-based Object Detection and Classification Using Faster R-CNN," *Int. Conf. on Computational Intelligence and Communication Technology*, 2018, hal. 1-6.
- [13] I. Chingovska, A. Anjos, dan S., Marcel, "On the Effectiveness of Local Binary Patterns in Face Non-spoofing," *Proc. of the Int. Conf. of Biometrics Special Interest Group (BIOSIG)*, 2012, hal. 1-7.