

## DETEKSI EKSPRESI WAJAH MENGGUNAKAN TENSORFLOW

Mohamad Ihsan<sup>1</sup>, Ratih Kumalasari Niswatin<sup>2</sup>, Daniel Swanjaya<sup>3</sup>

Teknik Informatika, Fakultas Teknik, Universitas Nisantara PGRI Kediri

Jln. Kh. Ahmad Dahlan Gg. II No.6 Mojoroto, Kota Kediri

E-mail: <sup>1</sup>Lfihisan18@gmail.com, <sup>2</sup>ratih.workmail@gmail.com, <sup>3</sup>daniel@unpkediri.ac.id

### ABSTRAKS

Ekspresi wajah adalah merupakan perubahan bentuk raut muka wajah dalam menanggapi keadaan perasaan, niat dan komunikasi sosial seseorang. Ekspresi wajah ini sangat bagus untuk di teliti karena merupakan alat komunikasi non verbal yang biasa digunakan oleh manusia untuk menggambarkan keadaan emosi atau perasaan dan untuk menyampaikan pesan sosial di kehidupan sehari-hari. Penelitian ini menggunakan machine learning open source library Tensorflow dengan menggunakan metode Convolutional Neural Network (CNN) yang dirancang khusus untuk pengenalan dan menentukan klasifikasi terhadap 7 ekspresi dasar wajah manusia ditambah ekspresi netral, metode ini memiliki hasil paling signifikan dalam hal pengenalan citra. Pemerataan distribusi data akan dilakukan untuk meningkatkan kinerja model. Hasil dari pengujian analisis di dapatkan hasil parameter optimal batch 32, epoch 100 dan dropout 0.6 dengan tingkat akurasi training 62.24%, akurasi validasi 62,44%, training loss 4,54% dan validation loss 4,02%. Di akhir penelitian ini, penulis melakukan percobaan pendeteksian ekspresi wajah dengan video secara realtime.

Kata Kunci: Ekspresi Wajah, Tensorflow, CNN

### ABSTRACT

Facial expression is the change in facial shape in an emergency of one's emotions, intentions, or social communication. This facial expression is very good to examine because it is a non-verbal communication tool commonly used by humans to describe emotional states or feelings and to convey social messages in everyday life. This research uses machine learning open source library Tensorflow using the Convolutional Neural Network (CNN) method which is specially designed for recognition and classification of 7 basic human facial expressions plus neutral expressions, this method has the most significant results in terms of image recognition. Even distribution of data is done to improve model performance. The results of the analysis test obtained the optimal results for batch 32, epoch 100 and dropout 0.6 with 62.24% training accuracy, 62.44% validation value, 4.54% training loss and 4.02% validation loss. At the end of this study, the authors conducted an experiment in realtime detection of facial expressions with video.

Keywords : Facial Expression, Tensorflow and CNN

## 1. PENDAHULUAN

### 1.1 Latar Belakang

Ekspresi wajah adalah merupakan perubahan bentuk raut muka wajah dalam menanggapi keadaan perasaan, niat dan komunikasi sosial seseorang (Tian, Y., Kanade, dkk, 2011). Perwujudan pada wajah manusia secara tidak langsung memberi ungkapan akan suatu bentuk perasaan, keinginan, serta tujuan tertentu. Setiap mimik wajah memiliki pemakaian yang dapat terbaca oleh orang lain, maka melalui mimik wajah dapat diketahui apa yang sedang terjadi dan bagaimana perasaan seseorang (Putra, Tezar Maas, 2016).

Dalam deteksi ekspresi wajah, terdapat sejumlah tantangan yaitu posisi bentuk wajah, skala, ekspresi, wajah yang terhalang objek lain, dan kondisi pencahayaan yang kurang (Yang, S., Luo, dkk, 2016). Pendeteksian ekspresi wajah ini bertujuan untuk mencari bentuk, posisi dan juga skala wajah di sebuah citra dan mengembalikannya ke koordinat area wajah untuk setiap wajah yang telah terdeteksi. Sedangkan *face alignment* ini bertujuan untuk

menyerasikan wajah yang terdeteksi dengan cara menggunakan satu set titik referensi yang terletak di dalam citra (Trigueros, D.S., Meng, dkk, 2018). Seiring dengan berkembangnya, metode-metode pengenalan ekspresi wajah atau *face detection* dan jua meningkatnya kinerja perangkat keras, banyak penelitian yang tidak hanya melakukan pendeteksian wajah tetapi juga melakukan pengenalan wajah, termasuk di dalamnya pengenalan ekspresi wajah. Pada umumnya, aplikasi pengenalan ekspresi wajah biasanya terdiri dari 4 bagian, yaitu *face detection*, *face alignment*, *feature extraction* dan *feature matching* (Li, S. Z., & Jain, A. K., 2011).

Ekspresi wajah ini sangat bagus untuk di teliti karena merupakan alat komunikasi non verbal yang biasa digunakan oleh manusia untuk menggambarkan keadaan emosi atau perasaan dan untuk menyampaikan pesan sosial di kehidupan sehari-hari. Seseorang terkadang ingin menutup-nutupi perasaan atau emosinya, tetapi hal ini sangat sulit dilakukan karena ekspresi wajah mereka biasanya akan menunjukkan perasaan yang sedang

dialami. Misalnya, seseorang ingin menutupi perasaan bencinya terhadap orang lain, tetapi pada saat tertentu tanpa sengaja ia akan menunjukkan perasaannya tersebut melalui ekspresi wajah yang ditampilkan. Sebaliknya, banyak orang yang salah dalam membaca ekspresi seseorang karena hanya melihat sekilas saja. Hal ini tentunya memerlukan pengamatan terus menerus terhadap perubahan ekspresi wajah seseorang. Namun hal ini tidak bisa dilakukan oleh seseorang secara langsung karena pada saat tertentu, mereka akan mengalami kejenuhan yang menghasilkan ketidakteelitian.

## 1.2 Referensi

### 1.2.1 Definisi Citra

Citra merupakan salah satu bagian komponen dari multimedia yang memiliki peranan penting karena berisi informasi dalam bentuk visual. Citra ini memiliki banyak informasi yang disampaikan dibandingkan dengan informasi dalam bentuk tulisan. tetapi tidak semua citra digital memiliki kualitas yang baik, sehingga citra tersebut tidak bisa menampilkan informasi yang akurat. Hal ini umumnya terjadi pada citra yang mempunyai nilai resolusi rendah (Mustaqim Nailul, 2011).

### 1.2.2 Definisi Tensorflow

*Tensorflow* adalah *library* perangkat lunak yang dikembangkan oleh Tim Google, yang bertujuan untuk mengerjakan pembelajaran mesin dan jaringan syaraf dalam penelitiannya. *Tensorflow* menggabungkan aljabar komputasi dengan teknik optimasi kompilasi, yang memfasilitasi perhitungan banyak ekspresi matematika (Taufiq, Imam, 2018). Fitur utama yang terdapat dalam *tensorflow* adalah:

1. Pemrograman pendukung jaringan syaraf dalam dan teknik *machine learning*.
2. Mendefinisikan, mengoptimalkan, dan menghitung secara matematis ekspresi wajah yang melibatkan *array multidimension* (tensors).
3. Pemakaian GPU (*Graphics Processing Unit*) yang efisien, mengotomasi manajemen dan mengoptimalkan memori yang sama terhadap data yang dipakai. *Tensorflow* mampu menulis kode yang sama dan menjalankannya di CPU atau GPU. Lebih khususnya lagi *tensorflow* dapat mengetahui bagian mana yang harus dipindahkan ke GPU.

Skalabilitas komputasi yang tinggi pada keseluruhan mesin terhadap kumpulan data yang besar.

## 2. PEMBAHASAN

Klasifikasi ekspresi wajah dalam penelitian ini menggunakan metode CNN yang dijabarkan menjadi dua bagian, yaitu proses *training* dan proses deteksi. Dimana proses *training* ini merupakan pembuatan model yang akan dipakai sebagai acuan klasifikasi, sedangkan pada proses deteksi merupakan penerapan model terhadap data uji atau data asing yang belum pernah digunakan. Pada

proses perancangan digunakan bahasa pemrograman php dengan menggunakan library *Tensorflow*.

Pengembangan metode CNN di dalam penelitian ini menggunakan data dari *The Facial Expression Recognition 2013* (FER-2013) yang dipopulerkan pada *International Conference on Machine Learning* (ICML) 2013. FER-2013 sendiri berisi 35.887 citra grayscale bentuk wajah yang berukuran 48x48, terdiri dari 7 jenis ekspresi yang berbeda. Data tersebut telah dilabeli dan di klasifikasikan menjadi 7 kelas dengan indeks 0 sampai 6 seperti pada Tabel 1.

**Tabel 1. Kelas ekspresi pada FER2013**

Label	Jenis Ekspresi	Jumlah
0	Marah ( <i>Angry</i> )	4593
1	Jijik ( <i>Disgust</i> )	547
2	Takut ( <i>Fear</i> )	5121
3	Bahagia ( <i>Happy</i> )	8989
4	Sedih ( <i>Sad</i> )	6077
5	Terkejut ( <i>Surprise</i> )	4002
6	Netral ( <i>Neutral</i> )	6198

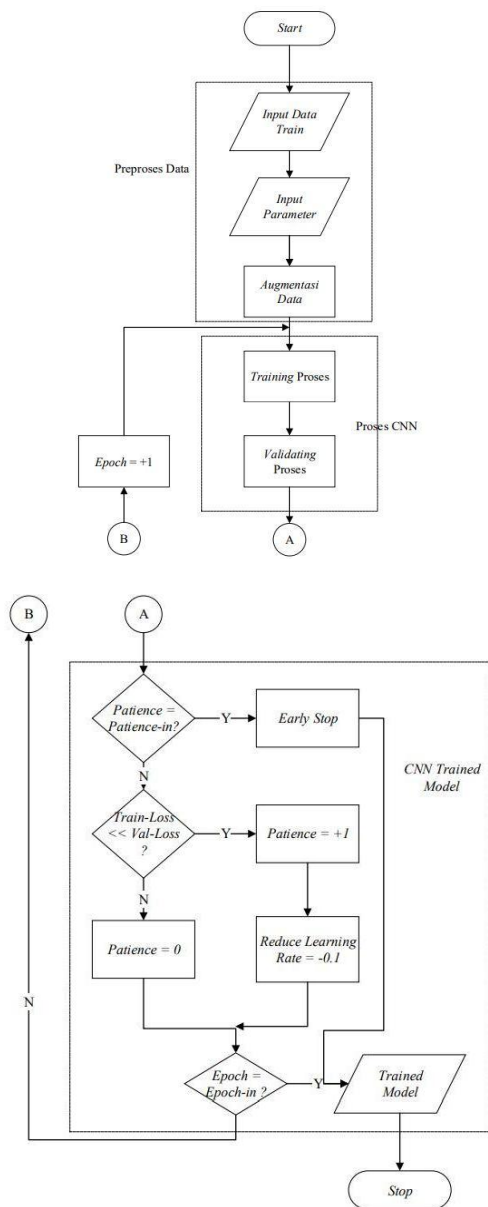
Pada Tabel 2 merupakan arsitektur CNN yang digunakan untuk melakukan proses training dengan standar *fully convolutional neural network* yang terdiri dari 10 lapisan konvolusi, *ReLU*, *Batch Normalization*, *Dropout* dan *Global Max Pooling*. Arsitektur ini mempunyai kurang lebih 600.000 parameter *training*. Nilai insialisasi *dropout* diberikan dengan nilai 0.5 sebagai titik tengah antara 0 dan 1 (Wu Haibing, Xiaodong Gu, 2015).

**Tabel 2. Arsitektur CNN**

Proses	Parameter
Konvolusi Batch Normalization	Filters = 16, kernel = (7, 7), Padding = 1
Konvolusi Batch Normalization Activation Relu	Filters = 16, kernel = (7, 7), Padding = 1
Max Pooling	Pool size = (2,2), padding = 1
Dropout	0.5
Konvolusi Batch Normalization	Filters = 32, kernel = (5, 5), Padding = 1
Konvolusi Batch Normalization Activation Relu	Filters = 32, kernel = (5, 5), Padding = 1
Max Pooling	Pool size = (2,2), padding = 1
Dropout	0.5
Konvolusi Batch Normalization	Filters = 64, kernel = (3, 3), Padding = 1
Konvolusi Batch Normalization Activation Relu	Filters = 64, kernel = (3, 3), Padding = 1
Max Pooling	Pool size = (2,2), padding = 1
Dropout	0.5
Konvolusi Batch Normalization	Filters = 128, kernel = (3, 3), Padding = 1
Konvolusi Batch Normalization Activation Relu	Filters = 128, kernel = (3, 3), Padding = 1
Max Pooling	Pool size = (2,2), padding = 1
Proses	Parameter
Dropout	0.5
Konvolusi Batch Normalization	Filters = 256, kernel = (3, 3), Padding = 1
Konvolusi Global Max Pooling Activation Softmax	Filters = 7, kernel = (3, 3), Padding = 1

Arsitektur diaplikasikan pada proses *training* beserta alur proses pada Gambar 1. Pada alur proses ini, dibagi menjadi tiga bagaian proses yaitu, *preproses* data, proses CNN dan proses pembuatan model. Pada bagian *peproses*, akan di lakukan normaliasi data agar nilai data yang diproses berada diantara -1 dan 1 dengan memakai persamaan (1), sehingga dapat lebih cepat diproses oleh *neural network*. Dimana nilai x adlah nilai normalisasi dan nila p adalah nilai awal dari data.

$$x = ((p \div 255) - 0,5) \times 2 \dots\dots\dots(1)$$



Gambar 1. Flowchart proses *training*

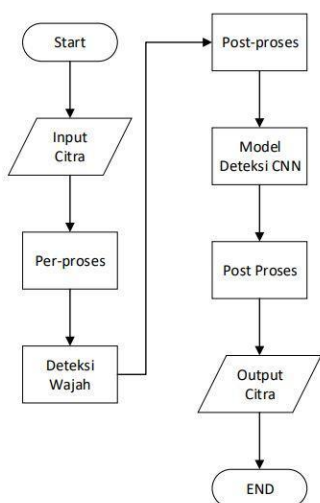
Selain proses *training*, juga akan dilakukan proses augmentasi data beserta cara alterasi berupa pergeseran citra sumbu x, pergeseran citra sumbu y, rotasi citra, memperkecil dan memperbesar citra, serta membalikan citra secara horizontal. Dalam penentuan nilai parameter awal seperti *batch*, *epoch*

dan *dropout*, akan ditentukan pada tahap *preproses* data, untuk selanjutnya akan dirubah dengan maksud menemukan parameter optimal. Pada tahap proses CNN, data akan dibagi menjadi dua bagian yaitu 28.704 data proses *training* dan 7.184 data proses validasi. Proses ini akan menghasilkan nilai akurasi dan nilai *loss*. Selanjutnya untuk proses validasi dan *training* akan dianalisa guna mencari model paling optimal dengan nilai akurasi validasi paling tinggi dan nilai *loss* paling rendah[13]. Pembentukan pada model proses CNN *trained model* ini melewati beberapa tahap untuk mencegah model akhir, demi mencapai *overfit* atau stagnansi yang optimal. Variabel *patience* ini akan digunakan untuk mencegah *overfit*, dimana nilai ini merupakan nilai maksimum *epoch* yang akan digunakan jika tidak ada perubahan nilai pada nilai *loss* validasi (Albawi Saad, Mohammed Tareq Abed, dkk, 2017).

Variabel ini akan terhubung menggunakan fungsi *early stop*, jika nilai *patience* mencapai maksimum data maka fungsi *early stop* akan memaksa menjalankan proses *training* untuk berhenti. Selain itu jika nilai *patience* mencapai seperempat nilai maksimum, maka fungsi lainnya bernama *reduce learning rate* akan dijalankan untuk mengurangi nilai *learning rate* sebesar 0,1 dari *adam optimizer*. Proses *training* akan dilakukan berulang kali dengan melakukan perubahan pada parameter berdasarkan Tabel 3. Proses ini dilakukan dengan Langkah-langkah seperti pada Gambar 2 untuk menentukan nilai parameter optimal dari model yang akan dihasilkan. Proses training dilakukan pada komputer dengan spesifikasi CPU AMD Ryzen™ 5- 3550H dengan GTX1050\_V3G, RAM 8 GB dan ROM 1 TB.



Gambar 2. Flowchart akurasi



Gambar 3. Flowchart proses deteksi

Tabel 3. Parameter Uji

Parameter uji	Parameter Tetap			Nilai Percobaan				
	Batch Time	Epoch	Drop out	1	2	3	4	5
Batch Time	-	50	0	16	32	64	128	256
Epoch	Best Batch Time	-	0	30	50	70	90	100
Drop Out	Best Batch Time	Best Epoch	-	0,2	0,4	0,5	0,6	0,8

Proses deteksi ini menggunakan model *feed forward* dengan data yang akan dimasukkan pada model yang sudah optimal, seperti pada Gambar 3. Proses ini nantinya akan terbagi menjadi dua bagian yaitu proses deteksi wajah dan proses deteksi emosi. Pada proses deteksi ekspresi wajah, menggunakan metode *haar cascade classifier* guna untuk menghasilkan posisi raut muka wajah pada data yang akan diidentifikasi jenis ekspresinya (Zufar Muhamad & Setiyono Budi, 2016).

Selanjutnya hasil dari deteksi wajah ini akan diidentifikasi dengan *output* berupa label ekspresi yang sudah ditentukan oleh *system*. Pada proses analisa nilai akurasi, *presisi* dan *recall*, akan dilakukan menggunakan metode literasi data dengan data citra asing sebanyak 3.589 yang belum dipakai pada proses *training* dan terbagi menjadi 7 variasi ekspresi seperti ditunjukkan pada Tabel 4. Hasil deteksi ekspresi pada setiap literasinya akan masukan pada tabel *confusion matix* yang selanjutnya akan dihitung nilai akurasi, *presisi*, dan *recall* menggunakan persamaan (2)(3) dan (4). Proses akuisisi data dilakukab seperti pada flowchart Gambar 4

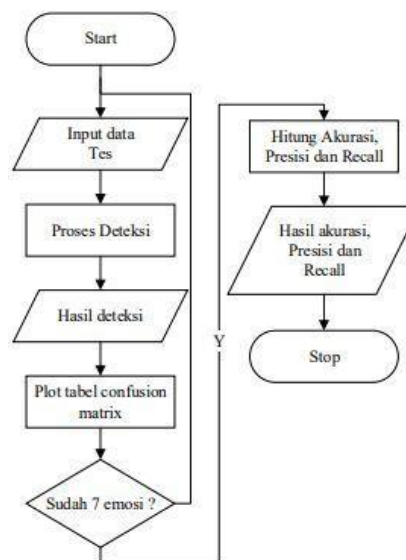
$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \dots\dots\dots(2)$$

$$Presisi = \frac{TP}{TP+FP} \dots\dots\dots(3)$$

$$Recall = \frac{TP}{TP+FN} \dots\dots\dots(4)$$

Tabel 4. Jenis ekspresi

Ekspresi	Jumlah data
Marah	467
Jijik	56
Takut	469
Senang	869
Sedih	653
Terkejut	415
Netral	607



Gambar 4. Flowchart Akurasi Deteksi

### 3. HASIL DAN PEBAHASAN

Normalisasi data pada tahap preproses dengan menggunakan Persamaan (1) ditunjukan pada Tabel 5. Nilai piksel data yang sebelumnya besar dinormalisasi agar berada pada jarak -1 sampai 1.

Tabel 5. Normalisasi nilai awal

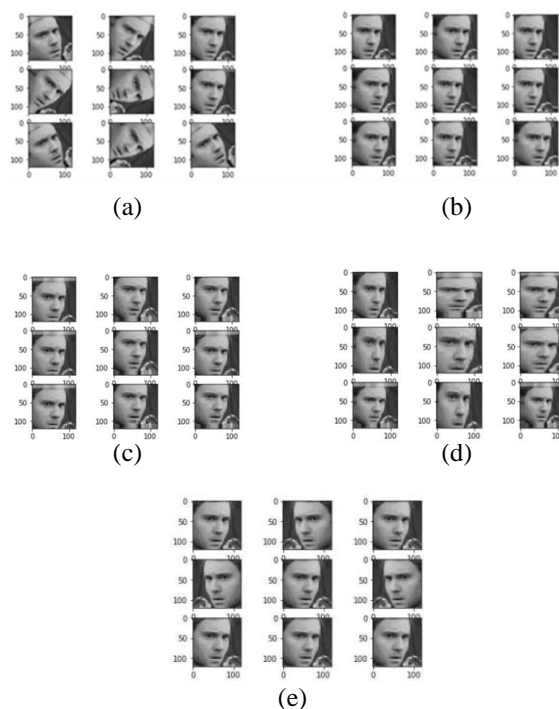
Nilai P	Persamaan	Nilai X
70	$x = ((p/255) - (0,5)) * 2$	0,45
80		-0,37
82		-0,35
72		-0,4
58		-0,54
60		-0,5
63		-0,5
54		-0,57
151		0,18
147		0,15
155		0,21

Augmentasi data memasukan nilai parameter untuk masing-masing augmentasi ditunjukan pada Tabel 6 dan hasil dari augmentasi ditunjukan pada Gambar 6. Untuk nilai rotasi dimasukan nilai 90, dimana sistem akan melakukan rotasi citra secara

acak pada data dengan rotasi antara  $-90^\circ$  sampai  $90^\circ$ . Nilai pergeseran pada sumbu x diberikan dengan nilai -10,10 dimana citra akan bergeser dengan nilai 10 piksel ke kiri dan 10 piksel ke kanan, begitu pula dengan pergeseran sumbu y dengan parameter yang sama dimana citra akan bergeser keatas dan kebawah. Nilai pembesaran citra diberikan dengan nilai 0,7 dan 1,3 dimana data akan diubah perbesarannya dengan jarak antara 70% sampai 130%. Terakhir citra akan dibalik secara horizontal, sehingga dengan menggunakan proses augmentasi didapat 5 variasi data citra input yang dihasilkan dari 1 citra input.

**Tabel 6. Parameter augmentasi**

Parameter	Nilai
<i>Rotation_range</i>	90
<i>Width_shift_rang</i>	-10,10
<i>Height_shift_range</i>	-10,10
<i>Zoom_range</i>	0,7,1,3
<i>Horizontal_flip</i>	TRUE



**Gambar 6. Hasil Augmentasi data (a) Augmentasi Rotasi; (b) Pergeseran Sumbu x; (c) Pergeseran Sumbu y; (d) Perbersaran; (e) Pembalikan Horisontal**

Setelah data disiapkan kemudian dilakukan proses training terhadap data dengan menggunakan variasi parameter uji pada Tabel 2. Hasil pelatihan terhadap beberapa parameter uji tersebut, didapatkan hasil akurasi dan *loss* yang dianggap paling baik untuk masing-masing skenario ditunjukkan pada Tabel 7 sampai dengan Tabel 9. Dari data tersebut dapat diambil parameter paling optimal dari tiga parameter *batch*, *epoch* dan *dropout* yang menghasilkan nilai paling baik dari semua skenario

yakni dengan nilai parameter *batch* 32, *epoch* 100, dan *dropout* 0,6. Parameter tersebut menghasilkan nilai akurasi *training* 62.24%, akurasi validasi 62.44%, nilai *training loss* 4.54% dan *validation loss* 4.02%.

**Tabel 7. Akumulasi Skenario I**

<i>Batch</i>	<i>Acc</i>	<i>Val_acc</i>	<i>Loss</i>	<i>Val_loss</i>
16	54,91%	60,35%	4,96%	4,38%
32	55,15%	60,59%	4,93%	4,36%
64	55,11%	59,92%	4,91%	4,40%
128	52,14%	57,91%	5,21%	4,62%
256	55,03%	59,36%	4,97%	4,49%

**Tabel 8. Akumulasi skenario ii**

<i>Epoch</i>	<i>acc</i>	<i>val_acc</i>	<i>loss</i>	<i>val_loss</i>
70	56,20%	60,42%	4,82%	4,30%
90	56,61%	62,15%	4,77%	4,20%
100	57,60%	62,80%	4,65%	4,11%
120	54,71%	60,41%	4,93%	4,31%
150	55,52%	60,46%	4,90%	4,31%

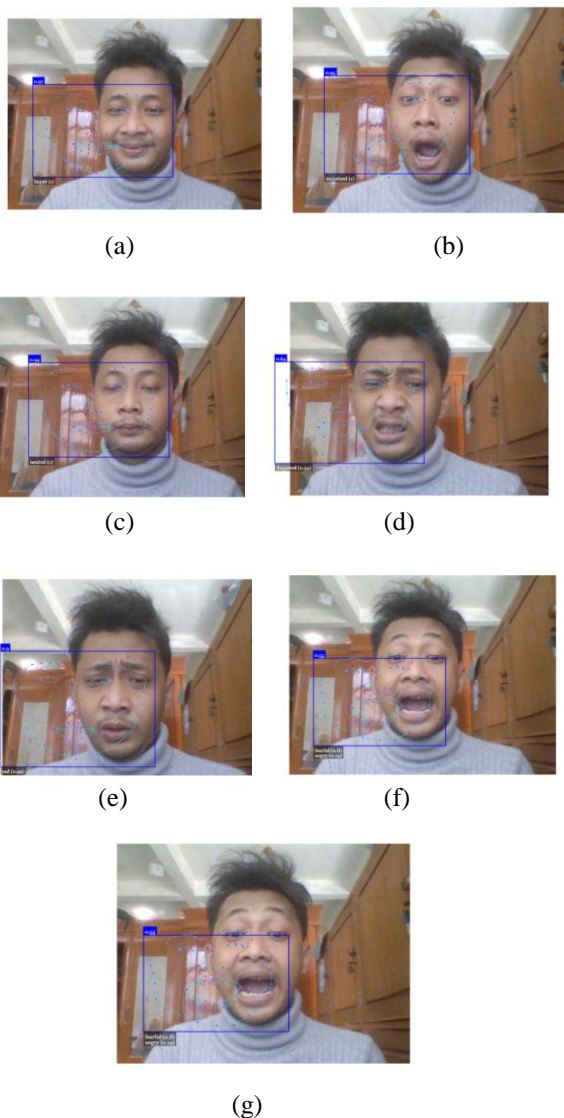
**Tabel 9. Akumulasi skenario iii**

<i>Epoch</i>	<i>acc</i>	<i>val_acc</i>	<i>loss</i>	<i>val_loss</i>
0.2	58,06%	62,78%	4,64%	4,14%
0.4	60,63%	64,22%	4,32%	4,00%
0.6	62,24%	62,44%	4,54%	4,02%
0.8	45,65%	52,01%	5,91%	5,15%
1	75,31%	66,72%	2,76%	4,12%

Hasil parameter optimal diujikan kembali pada data asing dengan hasil ditunjukkan pada Tabel 10. Nilai persentasi *cell* yang ditandai merupakan nilai *true positive* atau persentasi keberhasilan sistem dalam menebak emosi yang benar. Nilai persentasi pada baris dari tabel merupakan nilai *false positive* yakni nilai dimana sistem menebak emosi yang salah dengan benar, sedangkan pada nilai kolom merupakan nilai *false negative* yakni nilai dimana sistem menebak emosi yang salah dengan salah. Dari nilai-nilai persentasi tersebut dapat dihitung menggunakan persamaan (2), (3), dan (4) sehingga menghasilkan nilai akurasi 77%, presisi 72% dan recall 74%. Gambar 7 menunjukkan hasil dari deteksi pada 7 ekspresi wajah yang ditentukan dengan menggunakan citra input secara *realtime*.

**Tabel 10. Confussion matrix**

	<i>Marah</i>	<i>Jijik</i>	<i>Takut</i>	<i>Senang</i>	<i>Sedih</i>	<i>Terkejut</i>	<i>Netral</i>
<i>Marah</i>	70%	1%	10%	4%	2%	3%	10%
<i>Jijik</i>	26%	55%	5%	4%	5%	2%	3%
<i>Takut</i>	12%	1%	45%	5%	20%	7%	11%
<i>Senang</i>	2%	0%	2%	87%	2%	2%	5%
<i>Sedih</i>	9%	1%	10%	6%	55%	1%	19%
<i>Terkejut</i>	3%	0%	10%	5%	2%	80%	0%
<i>Netral</i>	5%	0%	5%	9%	3%	2%	75%



**Gambar 7. Hasil Deteksi 7 ekspresi : (a) Senang; (b) Terkejut; (c) Netral; (d) Jijik; (e) Sedih (f) Marah; (g) takut**

#### 4. KESIMPULAN

Berdasarkan hasil dari penelitian, klasifikasi ekspresi wajah menggunakan metode *Convolutional Neural Network* (CNN) dengan menggunakan library *TensorFlow* berhasil diimplementasikan untuk mendeteksi ketujuh ekspresi yakni marah, senang, sedih, jijik, terkejut takut dan netral. Dihilangkan nilai parameter optimal *batch* 32, *epoch* 100, dan *dropout* 0,6; dengan akurasi training 61,15%; akurasi validasi 64,54%; *training loss* 4,29%; dan *validasi loss* 3,96%; dari hasil akurasi tersebut dapat disimpulkan bahwa sistem tidak mengalami overfit karena nilai akurasi validasi lebih besar dari nilai akurasi *training*. Parameter Optimal tersebut diujikan kepada data asing yang belum pernah di deteksi sebelumnya dan menghasilkan nilai akurasi deteksi 67%, *presisi* 67% dan *recall* 66%..

#### PUSTAKA

- Tian, Y., Kanade, T., & Cohn, J. F. 2011. *Facial Expression Recognition*. In *Handbook of Face Recognition* (pp. 487–519). <https://doi.org/10.1007/978-0-85729-932-1>.
- Putra, Tezar Maas. 2016. Ekspresi Wajah Dalam Karya Lukis Surrealis. *Jurnal Fakultas Bahasa dan Seni Universitas Negeri Padang*.
- Yang, S., Luo, P., Loy, C. C., & Tang, X. (2016). WIDER FACE: A face detection benchmark. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2016-Decem, 5525–5533. <https://doi.org/10.1109/CVPR.2016.596>.
- Trigueros, D. S., Meng, L., & Hartnett, M. (2018). *Face Recognition: From Traditional to Deep Learning Methods*. *ArXiv*, (October 2018).
- Li, S. Z., & Jain, A. K. (2011). *Handbook of Face Recognition*. (S. Z. Li & A. K. Jain, Eds.), *Handbook of Face Recognition (2nd ed.)*. Springer. [https://doi.org/10.2990/29\\_1\\_103](https://doi.org/10.2990/29_1_103).
- Mustaqim Nailul (2011), Peningkatan Kualitas Citra Digital Menggunakan Metode Super Resolusi, *Universitas Syiah Banda Aceh*, Banda Aceh.
- Taufiq, Imam (2018). “Deep Learning Untuk Deteksi Tanda Nomor Kendaraan Bermotor Menggunakan Algoritma Convolutional Neural Network Dengan Python Dan Tensorflow”. Skripsi. Program Studi Sistem Informasi Sekolah Tinggi Manajemen Informatika dan Komputer AKAKOM.
- Wu Haibing, Xiaodong Gu, “Max-Pooling Dropout for Regularization of Convolutional Neural Networks”, di *ICONIP*, 2015, *Lecture Notes in Computer Science* vol. 9489.
- Goodfellow Ian, Bengio Yoshua and Courville Aaron, *Deep Learning*, 2016.
- Albawi Saad, Mohammed Tareq Abed dan Alzawi Saad, “Understanding of a Convolutional Neural Network”, *ICET*, 2017.
- Zufar Muhammad dan Setiyono Budi, “Convolutional Neural Networks untuk Pengenalan Wajah Secara Real-Time”, *Institut Teknologi Sepuluh Nopember, Jurnal Sains dan Seni ITS* Vol. 5, No.2, 2016.