

Estimasi Arah Tatapan Mata Menggunakan Ensemble Convolutional Neural Network

William Sugiarto
Program Studi Magister
Teknologi Informasi
Sekolah Tinggi Teknik Surabaya
williamsugiarto2011@gmail.com

Yosi Kristian
Program Studi Magister
Teknologi Informasi
Sekolah Tinggi Teknik Surabaya
yosi@stts.edu

Eka Rahayu Setyaningsih
Program Studi Magister
Teknologi Informasi
Sekolah Tinggi Teknik Surabaya
eka@stts.edu

Abstrak - Studi arah tatapan mata adalah salah satu masalah dalam bidang *computer vision*. Pengetahuan akan arah tatapan mata dapat memberikan informasi berharga yang dapat dimanfaatkan untuk berbagai macam keperluan dalam bidang lainnya, khususnya dalam bidang interaksi manusia dengan komputer. Dalam *paper* ini nantinya akan meneliti arah tatapan mata menggunakan *Ensemble Convolutional Neural Network* dengan menggunakan *dataset CAVE (Columbia Gaze Dataset)*. *Convolutional Neural Network (CNN)* merupakan sebuah bidang keilmuan dalam bidang *machine learning* yang berkembang cukup pesat khususnya untuk mengklasifikasi citra. Nantinya, *paper* ini akan menganalisa dan membandingkan hasil *F1 score* dan *weighted kappa (w-kappa) score* serta *error* dari klasifikasi dengan menggunakan 3, 9, dan 21 kelas. Dengan sama-sama menggunakan kanal RGB sebagai gambar input, maka dapat dibandingkan dan disimpulkan bahwa dengan menggunakan metode *Ensemble Convolutional Neural Network* dengan koefisien 1 untuk mata kiri, 1 untuk mata kanan, dan 3 untuk kedua mata untuk klasifikasi dengan 3 dan 9 kelas, serta dengan koefisien 1 untuk mata kiri, 1 untuk mata kanan, dan 5 untuk kedua mata untuk klasifikasi dengan 21 kelas dapat menghasilkan hasil *F1 score* dan *w-kappa* yang lebih baik, serta tingkat *error* yang lebih rendah daripada menggunakan koefisien dengan nilai lainnya.

Kata Kunci: *Convolutional Neural Network, Eye Gaze, Gaze Estimation.*

mata sebagai kontrol input. Sedangkan tujuan dari penelitian yang dilakukan adalah sebagai alat bantu teknologi (*assistive technology*) yang berfungsi sebagai alat input yang dapat digunakan pada bidang interaksi manusia dengan komputer, *user experience*, sosiologi, edukasi, *marketing*, dalam bidang perindustrian, dan juga bidang-bidang lainnya yang dapat membantu para penyandang disabilitas.

Ada beberapa macam pendekatan yang dapat digunakan untuk menentukan arah tatapan mata. Pada penelitian ini akan menggunakan metode pendekatan berbasis CNN (*Convolutional Neural Network*). CNN merupakan metode yang terbilang masih relatif baru dan diharapkan dapat memperoleh tingkat kinerja yang jauh lebih baik dibandingkan metode-metode sebelumnya. CNN yang merupakan kategori dari *neural network* ini terbukti sangat baik dalam hal klasifikasi citra. CNN mempunyai *layer* untuk *convolutional* dan *pooling*, yang dimana MLP (*Multilayer Perceptron*) tidak memilikinya.

Convolutional Neural Network pertama kali dikembangkan oleh seorang peneliti bernama Kunihiko Fukushima dari NHK *Broadcasting Science Research Laboratories*, Kinuta, Setagaya, Tokyo, Jepang dengan nama NeoCognitron [2]. *Convolution Neural Network* bermula dari Yann LeCun dan teman-temannya berhasil melakukan klasifikasi citra kode zip menggunakan kasus khusus dari *Feed Forward Neural Network* Pada tahun 1989 [3]. Mulai pada tahun 2012, dimana perangkat keras *Graphical Processing Unit (GPU)* berkembang, DNN juga ikut berkembang dan bahkan sebuah CNN dapat melakukan pengenalan citra dengan akurasi yang sangat baik dan dapat menyaingi manusia pada *dataset* tertentu [4].

I. PENDAHULUAN

Mata merupakan fitur untuk berkomunikasi dengan tampilan antarmuka yang memiliki peran yang sangat vital. Perhatian seseorang dapat terlihat dari arah tatapan matanya. Mata dan pergerakannya dapat menunjukkan perhatian seseorang dan memiliki peran yang penting dalam mengkomunikasikan suatu informasi sosial maupun emosional [1].

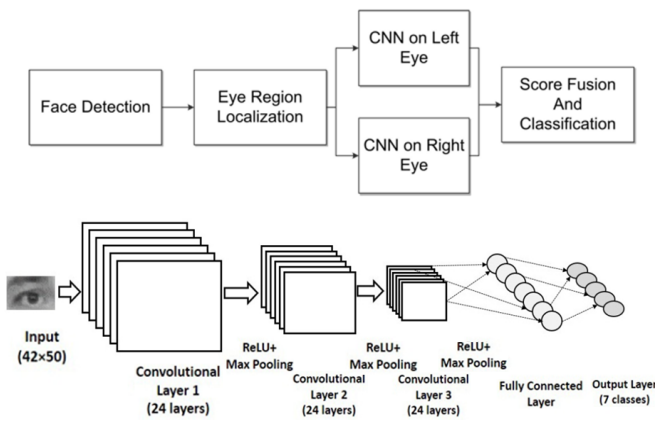
Penelitian tentang pelacakan arah tatapan mata hingga saat ini masih berfokus pada menginterpretasikan posisi mata pada layar secara lebih akurat. Latar belakang dari penelitian yang dilakukan adalah sebagai studi kognitif yang menggunakan

II. PENELITIAN TERKAIT

Ada beberapa jenis pendekatan pelacakan arah tatapan mata. Ada pendekatan pelacakan arah tatapan mata yang berbasis fitur [5]–[9] dan juga ada yang berbasis tampilan [10]–[11] dan kedua jenis tersebut merupakan metode yang sudah lama dan sudah digantikan dengan menggunakan metode berbasis CNN. Kedua pendekatan tersebut pada umumnya memerlukan perangkat keras yang cukup mahal dan tidak praktis. Berbeda dengan menggunakan pendekatan berbasis

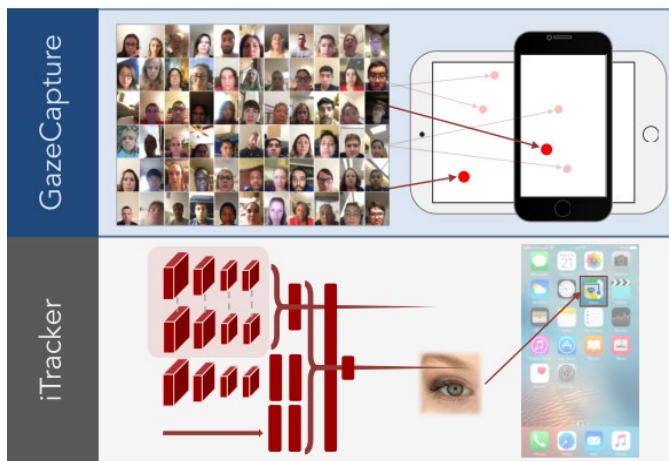
CNN yang pada umumnya hanya memerlukan sebuah *web camera*.

Telah ada beberapa karya terkait yang meneliti arah tatapan mata dengan menggunakan metode CNN, salah satunya adalah [12]. Pada *paper* tersebut menggunakan CNN untuk mengelolah mata kanan dan mata kiri secara independen dan membandingkan hasil *recognition rate* antara 3 dan 7 kelas. Pada penelitian tersebut, dua jaringan CNN dilatih secara independen untuk mata kanan dan mata kiri seperti yang ditunjukkan pada Gambar 1 dimana kelasnya merupakan hasil maksimal dari rata-rata hasil CNN mata kanan dan mata kiri. Penelitian tersebut meneliti untuk arah tatapan mata tiga dan tujuh kelas. Penelitian tersebut dilakukan dengan dua metode yang berbeda, yaitu menggunakan ERT (*Ensemble of Randomized Tree*) [13] dan menggunakan *geometric* (ROI).

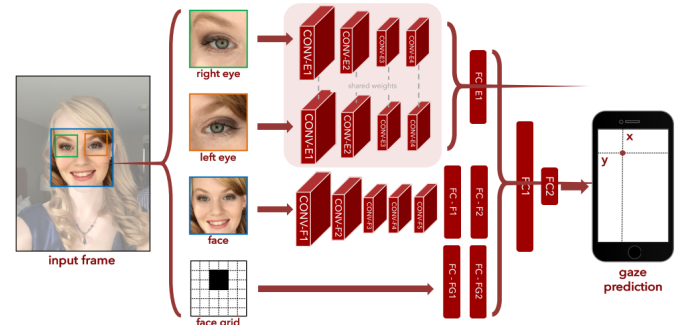


Gambar 1. Skema dan Arsitektur yang Digunakan pada “Real-time Eye Gaze Direction Classification Using Convolutional Neural Network” [12].

Ada juga penelitian yang meneliti arah tatapan mata ini untuk layar *mobile phone* maupun *tablet* [14]. Menggunakan *Gaze capture* sebagai pengumpul data dan *iTracker* sebagai *deep convolutional neural network* untuk memprediksi arah tatapan mata seperti pada Gambar 2. Skema dan arsitektur yang dipakai digambarkan pada Gambar 3.



Gambar 2. Gaze Capture yang Digunakan [14].



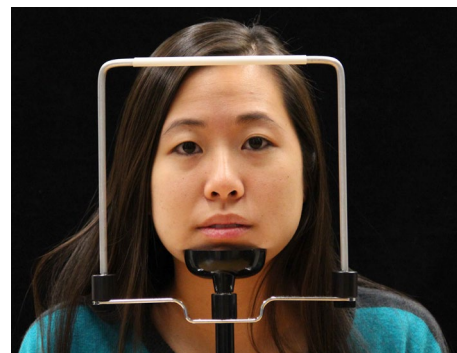
Gambar 3. Skema dan Arsitektur yang Digunakan pada “Eye Tracking for Everyone” [14].

III. METODE PENELITIAN

Pada penelitian ini, *dataset* yang digunakan bersumber dari *Columbia Gaze Dataset* (CAVE) [15]. *Dataset* ini berisi 5.880 gambar dengan total 56 subyek (32 subyek pria dan 24 subyek wanita) dengan rentang usia 18 hingga 36 tahun dengan resolusi gambar sebesar 5.184 x 3.456.

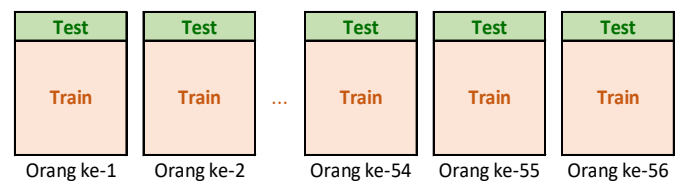
Untuk setiap subyek, diambil gambar dengan lima *horizontal head pose* (0°, ±15°, ±30°) dengan tujuh arah tatapan secara horizontal (0°, ±5°, ±10°, ±15°) dan tiga arah tatapan secara vertikal (0°, ±10°).

Dataset ini menyuguhkan gambar yang sangat variatif, dimana ada yang menggunakan kacamata maupun tanpa kacamata. Dalam *dataset* ini juga terdiri dari berbagai macam warna kulit dan memiliki rentang usia antara 18 hingga 36 tahun. Gambar 4 merupakan salah satu contoh *dataset*-nya.



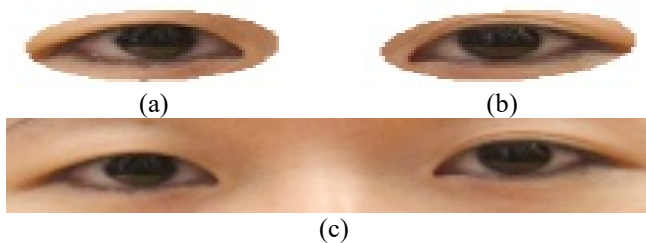
Gambar 4. Contoh Columbia Gaze Dataset.

Sebesar 20% dari *dataset* akan digunakan sebagai *test set* dan 80% sisanya akan digunakan sebagai *train set*. Metode untuk *train* dan *test split* menggunakan *seen respondent test set* seperti pada Gambar 5.



Gambar 5. Seen Respondent Test Set.

Berikut ini adalah skema yang akan digunakan dalam penelitian ini. Proses awal yang dilakukan adalah dengan memanipulasi gambar asli dengan menggunakan *gamma correction* sebesar 0,75. Kemudian memecah atau memotong gambar *dataset* kedalam tiga bagian terlebih dahulu, yaitu mata kanan, mata kiri, dan kedua mata. Untuk pemotongannya dilakukan dengan melakukan *facial landmark detection* [16]-[17] untuk mengambil titik-titik pada area mata kiri dan mata kanan. Untuk mata kanan dan kiri diambil secara oval dan memiliki latar belakang berwarna putih seperti yang dicontohkan pada Gambar 6. Dengan menggunakan metode tersebut, dari 5.880 gambar hanya 5.833 gambar yang dapat diproses dan sisanya diabaikan.



Gambar 6. Hasil *Eye Region Localization* untuk (a) Mata Kiri, (b) Mata Kanan, (c) Kedua Mata.

Adapun kaidah untuk menentukan pembagian kelas berdasarkan sudut arah tatapan mata sesuai dengan yang tertera pada Gambar 7.

3 KELAS		-15	-10	-5	0	5	10	15
	10	A		B			C	
	0	A		B			C	
	-10	A		B			C	
9 KELAS		-15	-10	-5	0	5	10	15
	10	A		B			C	
	0	D		E			F	
	-10	G		H			I	
21 KELAS		-15	-10	-5	0	5	10	15
	10	A	B	C	D	E	F	G
	0	H	I	J	K	L	M	N
	-10	O	P	Q	R	S	T	U

Gambar 7. Pembagian Nama Kelas Berdasarkan Derajat Sudut Tatapan.

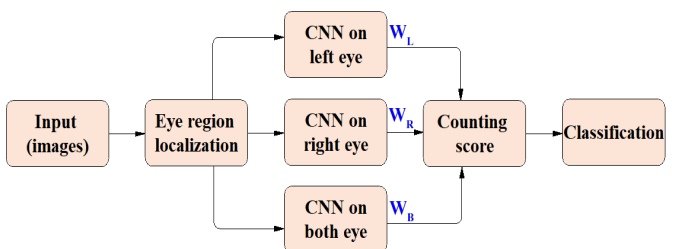
Proses CNN akan dilakukan sebanyak 500 *epoch* pada masing-masing gambar secara independen. Persamaan yang digunakan untuk menghitung *score* akhir yaitu:

$$score = \left(\frac{L \text{ Score} \times WL + R \text{ score} \times WR + B \text{ score} \times WB}{WL + WR + WB} \right) \dots\dots\dots(1)$$

Score L merupakan hasil dari perhitungan *softmax* untuk mata kiri, *score* R untuk mata kanan, dan *score* B untuk kedua mata. Koefisien untuk mata kiri dilabelkan dengan WL, dan untuk mata kanan dilabelkan dengan WR, serta untuk kedua mata dilabelkan dengan WB. Kelas yang dipilih akan ditentukan dengan probabilitas maksimum yaitu:

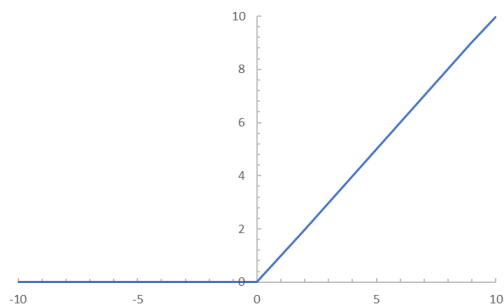
$$class = \arg \max(score) \dots\dots\dots(2)$$

Pada Gambar 8, menunjukkan skema rangkaian kerja yang akan digunakan pada penelitian ini. Pada gambar ini juga dapat menjelaskan dengan lebih mudah maksud dari persamaan 1 dan 2.



Gambar 8. Skema yang Digunakan

Dalam CNN terdapat beberapa macam fungsi aktivasi yang dapat digunakan diantaranya adalah *Rectified Linear Unit* (ReLU) dan *softmax*. ReLU merupakan fungsi aktivasi yang akan menghasilkan nilai nol apabila $x < 0$ dan kemudian linier dengan kemiringan 1 ketika $x > 0$



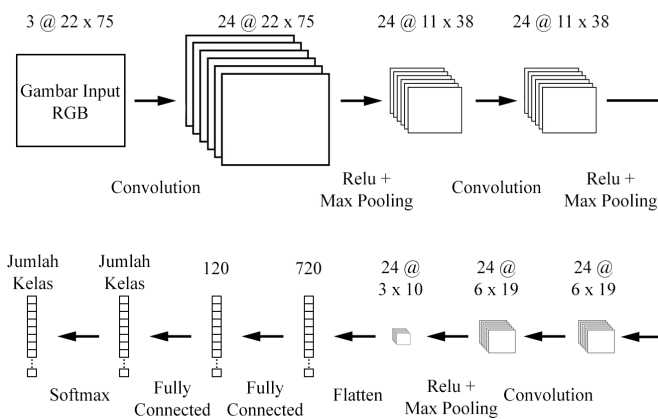
Gambar 9. Fungsi Aktivasi *Rectified Linear Unit* (ReLU).

Aktivasi *softmax* dipilih karena output dari lapisan terakhir pada sebuah jaringan saraf akan diubah menjadi distribusi probabilitas dasarnya dan memiliki rentang antara 0 hingga 1, dan juga jumlah semua probabilitasnya adalah satu. Adapun persamaan untuk menghitung *softmax* yaitu:

$$f_i(\vec{x}) = \frac{e^{x_i}}{\sum_{j=1}^k e^{x_j}} \text{ for } i = 1, 2, 3, \dots, k \dots\dots\dots(3)$$

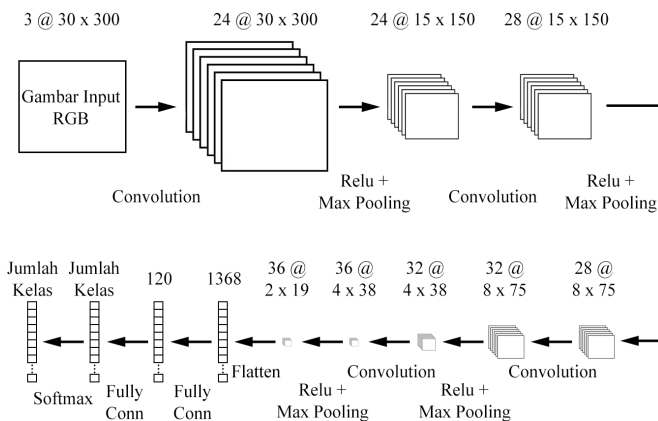
Pada penelitian ini menggunakan *Adam Optimizer* [18]. Nama Adam berasal dari *adaptive moment estimation*. Adam adalah algoritma optimasi yang dapat digunakan sebagai pengganti prosedur *stochastic gradient descent* klasik untuk memperbaharui bobot jaringan yang berbasis pada data pelatihan.

Untuk proses CNN pada mata kiri dan mata kanan akan menggunakan arsitektur seperti tertera pada Gambar 10, yang dimana *convolutional layer* tahap pertama menggunakan 24 filter dengan dimensi 7 x 7 dan diikuti dengan *rectified linear unit* serta 2 x 2 *max pooling layer*. Sedangkan untuk *convolutional layer* tahap kedua dan ketiga, keduanya menggunakan 24 filter dengan dimensi 5 x 5 dan juga 3 x 3, serta proses ReLU dan *max pooling* yang sama seperti pada tahap sebelumnya.



Gambar 10. Arsitektur CNN untuk Mata Kanan dan Mata Kiri.

Untuk proses CNN dengan menggunakan kedua mata akan menggunakan arsitektur yang berbeda. Arsitektur yang digunakan untuk CNN dengan kedua mata ditunjukkan pada Gambar 11 dengan dimensi filter yang digunakan yaitu 24 @ 7 x 7, 28 @ 7 x 7, 32 @ 5 x 5, dan 36 @ 3 x 3 dan masing-masingnya dilakukan proses ReLU dan *max pooling* yang sama seperti pada proses CNN untuk mata kanan dan mata kiri.



Gambar 11. Arsitektur CNN untuk Kedua Mata.

Gambar yang digunakan sebagai input merupakan gambar dengan 3 kanal (gambar RGB). Hasil outputnya sejumlah banyaknya kelas itu sendiri dan merupakan hasil *softmax* dari 3, 9, atau 21 kelas tergantung dari apa yang sedang diujicobakan.

Untuk menghitung kinerja sistem dapat menggunakan *F1 score*. *F1-score* adalah ukuran suatu kinerja yang mempertimbangkan presisi *p* maupun *recall* *r* untuk menghitung skor. Nilai *F1-score* memiliki nilai terbaiknya yaitu 1 dan terburuk adalah 0.

$$F = 2 \frac{PR}{P+R} \dots\dots\dots(4)$$

Weighted kappa (*w-kappa*) juga dapat digunakan untuk mengukur kinerja sistem. Dengan menggunakan *w-kappa*, maka jarak jauh dekatnya *error* akan mempengaruhi nilai yang dihasilkan. Khusus untuk penelitian ini, persamaan *w-kappa* yang digunakan sedikit diubah agar rentang nilai yang dihasilkan berapa pada 0 untuk nilai terendahnya, dan 1 untuk nilai tertinggi. Persamaan *w-kappa* yang digunakan adalah sebagai berikut:

$$\kappa_w = \max(0, 1 - \frac{\sum_{i,j} w_{ij} p_{ij}}{\sum_{i,j} w_{ij} e_{ij}}) \dots\dots\dots(5)$$

Dimana untuk menetapkan nilai koefisiennya, maka digunakan teorema pythagoras. Sebagai contoh kasus, jika kelas yang sebenarnya adalah kelas "A" dan ditebak sebagai kelas "P" pada klasifikasi dengan menggunakan 21 kelas, maka akan memiliki nilai koefisien sebesar 2,236 seperti pada Gambar 12.

A	B	C	D	E	F	G
H	I	J	K	L	M	N
O	P	Q	R	S	T	U

Gambar 12. Perhitungan Koefisien W-Kappa untuk Klasifikasi dengan 21 Kelas.

Hasil perhitungan tersebut akan dibulatkan menjadi tiga angka dibelakang koma. Pada gambar 13, telah dicontohkan nilai koefisien untuk klasifikasi dengan menggunakan 9 kelas. Sebagai catatan tambahan, bahwa hasil tabel tersebut simetris pada diagonalnya.

Sedangkan *error* yang dihasilkan memiliki nilai dengan rentang mulai dari 0 hingga 1. Hasil bernilai 0 apabila semua tebakannya benar, dan bernilai 1 apabila semua tebakannya salah. Untuk menghitung *error* yang dihasilkan, dapat digunakan persamaan sebagai berikut:

$$Error = \frac{\text{number of images} - \text{number of correct predictions}}{\text{number of images}} \dots\dots\dots(6)$$

	A	B	C	D	E	F	G	H	I
A	0	1	2	1	1,414	2,236	2	2,236	2,828
B	1	0	1	1,414	1	1,414	2,236	2	2,236
C	2	1	0	2,236	1,414	1	2,828	2,236	2
D	1	1,414	2,236	0	1	2	1	1,414	2,236
E	1,414	1	1,414	1	0	1	1,414	1	1,414
F	2,236	1,414	1	2	1	0	2,236	1,414	1
G	2	2,236	2,828	1	1,414	2,236	0	1	2
H	2,236	2	2,236	1,414	1	1,414	1	0	1
I	2,828	2,236	2	2,236	1,414	1	2	1	0

Gambar 13. Koefisien W-Kappa untuk Klasifikasi dengan Sembilan Kelas.

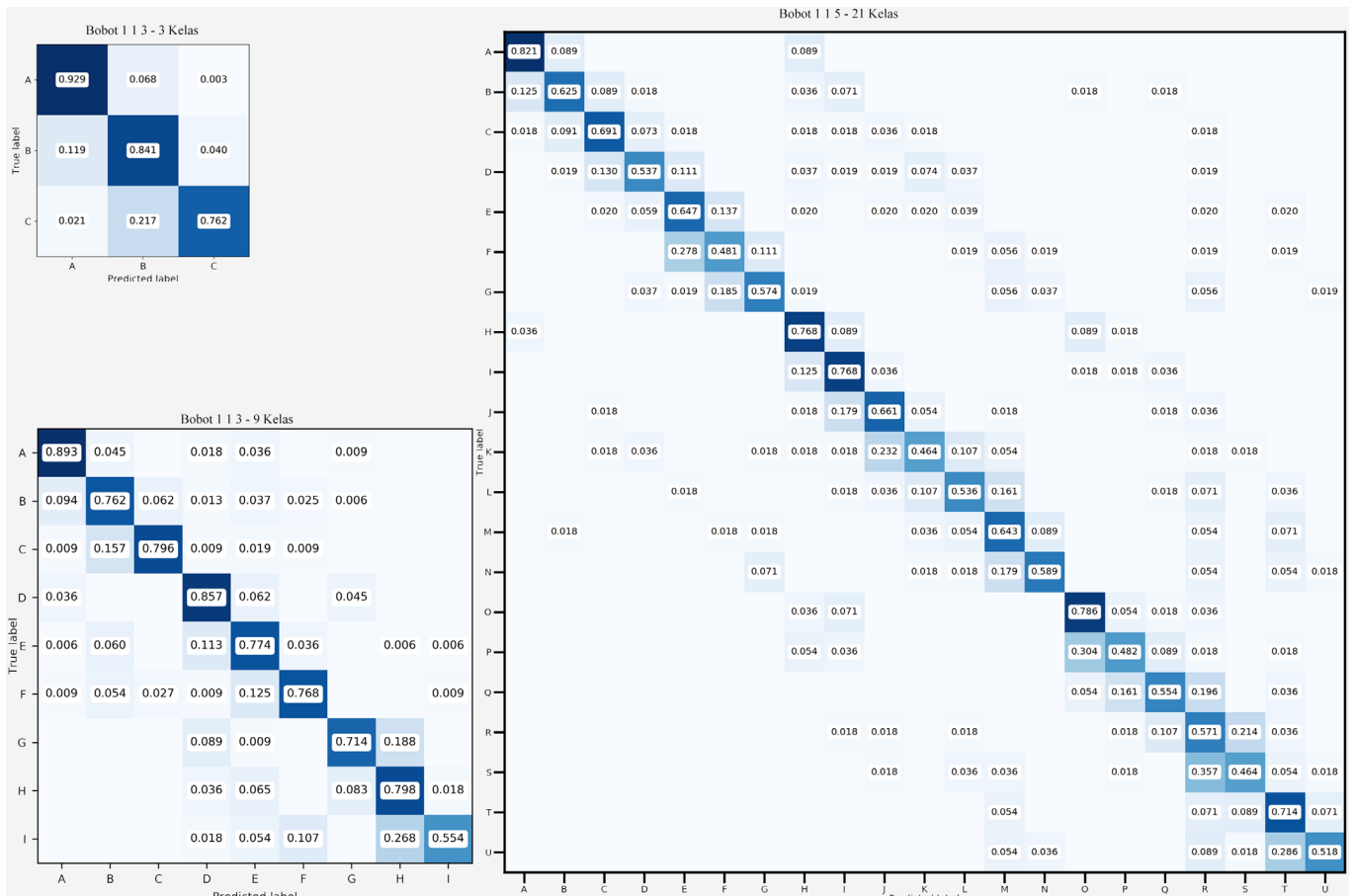
IV. UJI COBA

Uji coba ini dilakukan dengan menggunakan koefisien yang berbeda-beda. Untuk mempermudah penyebutan, maka penamaan koefisien dimulai dari WL, lalu WR, dan WB.

Misalkan koefisien 1 1 3 yang berarti koefisien 1 untuk WL (mata kiri), 1 untuk WR (mata kanan), dan 3 untuk WB (kedua mata).

Tabel 1. Hasil Training

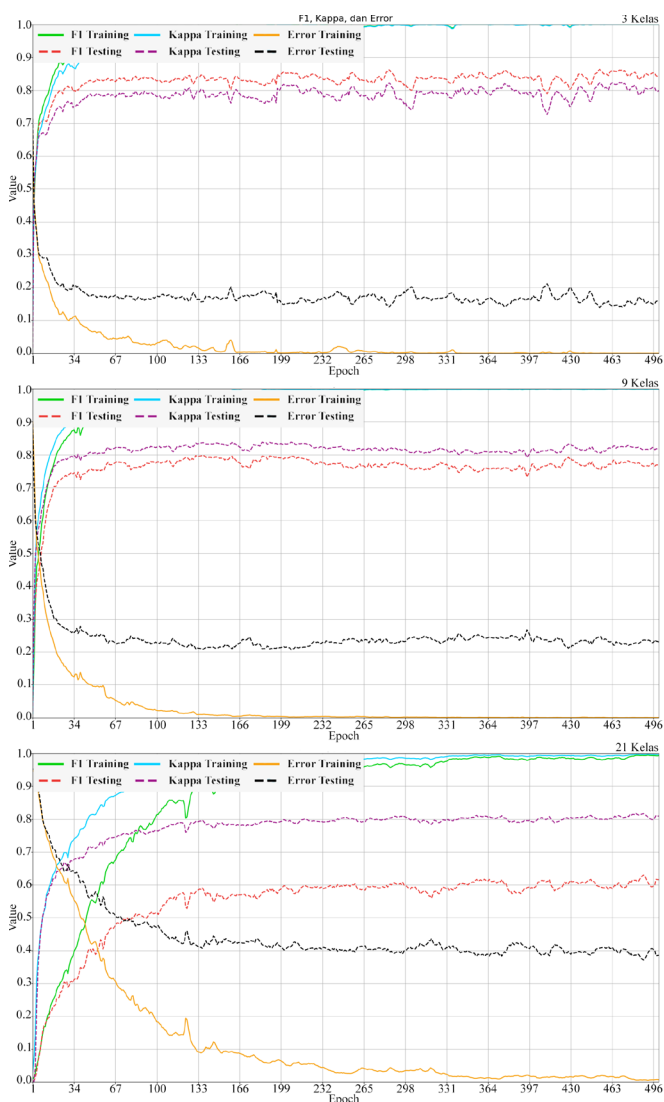
Koefisien	Metode Pengukuran	3 Kelas		9 Kelas		21 Kelas	
		Hasil Training	Hasil Testing	Hasil Training	Hasil Testing	Hasil Training	Hasil Testing
1 1 1	F1 Score	1.000	0.819	0.924	0.643	0.987	0.491
	W-Kappa	1.000	0.766	0.929	0.630	0.990	0.729
	Error	0.000	0.185	0.080	0.387	0.013	0.507
1 1 2	F1 Score	1.000	0.828	1.000	0.758	0.987	0.562
	W-Kappa	1.000	0.779	1.000	0.809	0.993	0.776
	Error	0.000	0.175	0.000	0.241	0.013	0.442
1 1 3	F1 Score	1.000	0.845	1.000	0.771	0.989	0.575
	W-Kappa	1.000	0.799	0.999	0.819	0.994	0.778
	Error	0.000	0.156	0.000	0.230	0.011	0.424
1 1 4	F1 Score	1.000	0.841	1.000	0.759	0.994	0.598
	W-Kappa	1.000	0.794	1.000	0.811	0.997	0.797
	Error	0.000	0.162	0.000	0.242	0.006	0.401
1 1 5	F1 Score	1.000	0.839	1.000	0.761	0.993	0.615
	W-Kappa	1.000	0.793	1.000	0.812	0.996	0.809
	Error	0.000	0.164	0.000	0.241	0.007	0.386
1 1 7	F1 Score	1.000	0.834	0.999	0.763	0.994	0.583
	W-Kappa	1.000	0.783	0.999	0.811	0.997	0.795
	Error	0.000	0.168	0.001	0.240	0.006	0.414



Gambar 14. Arsitektur CNN untuk Kedua Mata.

Uji coba ini dilakukan dengan menggunakan 6 macam koefisien yang berbeda. Keenam koefisien tersebut adalah 1 1 1, 1 1 2, 1 1 3, 1 1 4, 1 1 5, dan 1 1 7 dan diujicobakan untuk 3, 9, dan 21 kelas. Untuk semua percobaan yang diujikan menggunakan gambar input RGB dan dilakukan *training* sebanyak 500 *epoch* dengan ukuran *batch* sebesar 64 dan *learning rate* sebesar $1e-4$. Kompleksitas perhitungan yang dipakai termasuk cukup kompleks. Dengan menggunakan Core i5 - 6200U @ 2.30 GHz (8GB RAM) pada Windows 10 dan Nvidia GeForce 930MX dengan *dedicated GPU memory* 2GB memerlukan waktu *training* sekitar 9 hingga 10 jam.

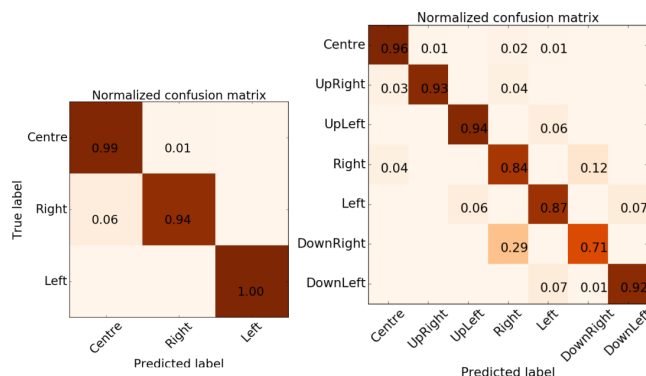
Pada Tabel 1, klasifikasi menggunakan 3 kelas pada koefisien 1 1 3 menghasilkan hasil *testing* paling baik, dimana *F1 score* mencapai 0,845, *w-kappa* sebesar 0,799, dan *error* hanya 0.156. Hasil untuk klasifikasi 9 kelas yang didapat serupa dengan hasil klasifikasi menggunakan 3 kelas dimana nilai terbaiknya berada pada koefisien 1 1 3 dengan *F1 score* sebesar 0,771, *w-kappa* sebesar 0,819, dan *error* hanya 0,23.



Gambar 15. Grafik *Training* dan *Testing* dengan Menggunakan Koefisien Terbaik.

Namun, untuk klasifikasi dengan menggunakan 21 kelas, performa optimalnya didapat pada koefisien 1 1 5 dimana *F1 score* yang dihasilkan yaitu 0,615, dan *w-kappa* sebesar 0,809, serta *error* yang didapat yaitu 0,386. Pada gambar 14 merupakan hasil *confusion matrix* dengan menggunakan koefisien terbaik. Dan pada gambar 15 menunjukkan grafik *training* dan *testing* dengan menggunakan koefisien terbaik untuk 3, 9, dan 21 kelas.

Sebagai hasil perbandingan, akan digunakan *paper* yang ditulis oleh A. George bersama dengan A. Routray yang berjudul “*Real-time Eye Gaze Direction Classification Using Convolutional Neural Network*” [12]. Pada penelitian tersebut *dataset* yang digunakan adalah *Eye Chimera database* [19], [20] yang mengandung tujuh kelas EAC (*Eye Accessing Cues*). *Dataset* tersebut berisi 40 subyek dengan total 1170 gambar.



Gambar 16. *Normalized Confusion Matrix* pada *Paper* [12].

Pada Gambar 16 merupakan *normalized confusion matrix* untuk 3 dan 7 kelas pada *paper* [12]. Agar hasil yang dibandingkan dapat mendekati kemiripan, meskipun *dataset* yang digunakan berbeda dikarenakan tidak berhasil untuk mendapatkan *dataset Eye Chimera database*, maka hasil dari percobaan 9 kelas pada penelitian ini akan dikonversi menjadi 7 kelas arah tatapan mata seperti yang ditunjukkan pada Gambar 17.

	-15	-10	-5	0	5	10	15
10	Up Left			Centre			Up Right
0	Left			Centre			Right
-10	Down Left			Centre			Down Right

Gambar 17. Konversi Dari 9 Kelas Menjadi 7 Kelas Arah Tatapan Mata.

Tabel 2 merupakan tabel hasil yang dicapai pada 3 dan 7 kelas arah tatapan mata melengkapi tabel perbandingan yang ditulis pada *paper* [12]. Pada tabel tersebut menunjukkan bahwa hasil yang didapat pada penelitian ini sedikit dibawah penelitian yang dilakukan oleh [12]. Hal itu dimungkinkan karena adanya perbedaan *dataset* yang digunakan dimana *Eye Chimera* hanya memiliki 1.170 gambar, sedangkan pada *Columbia Gaze Dataset* memiliki gambar yang lebih bervariasi dan lebih banyak (pada penelitian ini menggunakan 5.833 gambar).

Tabel 2. Perbandingan Akurasi

Lokalisasi Mata	Klasifikasi Arah Mata	Jumlah Gambar Dataset	3 Kelas (%)	7 Kelas (%)
BoRMaN [21]	Valenti [22]	1.170	55.73	39.83
Zhu [23]	Zhu [23]	1.170	63.01	43.29
Vrănceanu [20]	Vrănceanu [20]	1.170	95.21	83.08
George (ROI) [12]	George [12]	1.170	85.58	97.65
George (ERT) [12]	George [12]	1.170	98.32	89.81
Proposed [16], [17]	Proposed	5.833	84.36	79.36

V. KESIMPULAN DAN SARAN

Dari hasil uji coba, dengan menggunakan *dataset* dengan karakteristik *seen responden test set* dan gambar input RGB, maka diperoleh nilai koefisien yang baik adalah 1 1 3 untuk klasifikasi dengan 3 dan 9 kelas, dan 1 1 5 untuk klasifikasi dengan 21 kelas. Dengan menggunakan koefisien yang tepat, dapat meningkatkan F1 *score* maupun w-kappa serta menurunkan *error*. Akan tetapi koefisien yang tidak pas atau terlalu besar/kecil terutama koefisien pada kedua mata, malah dapat menurunkan performa.

Terdapat beberapa saran untuk penelitian selanjutnya, yaitu mencoba dengan menggunakan gambar *greyscale* ataupun dengan gambar input yang memiliki *gamma correction* yang berbeda dengan *paper* ini. Serta perlunya mencari nilai koefisien yang lain yang dapat meningkatkan performa yang ada. Bahkan juga dapat merubah arsitektur untuk mata kanan dan kiri, serta arsitektur untuk kedua mata misalnya pada ukuran gambar input, jumlah dan ukuran filter, ataupun memodifikasi layer yang digunakan.

REFERENSI

- [1] Coates, A., Arbores, A., & Ng, A.Y. (2011). *An Analysis of Single-Layer Networks in Unsupervised Feature Learning*. *Aistats 2011*, 15, pp. 215–223. <https://doi.org/10.1109/ICDAR.2011.95>
- [2] Florea, L., & Vertan, C. (2013). *Can Your Eyes Tell Me How You Think? A Gaze Directed Estimation of the Mental Activity*. *Bmvc2013*, pp. 1–11. <https://doi.org/10.5244/C.27.60>
- [3] Fukushima, K. (1980). *Neocognitron: A Self-organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position*. *Biological Cybernetics*, 36(4), pp. 193–202. <https://doi.org/10.1007/BF00344251>
- [4] George, A., & Routray, A. (2016). *Real-time Eye Gaze Direction Classification Using Convolutional Neural Network*.
- [5] Kazemi, V., & Sullivan, J. (2014). One Millisecond Face Alignment with an Ensemble of Regression Trees. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 1867–1874. <https://doi.org/10.1109/CVPR.2014.241>
- [6] Kingma, D.P., & Ba, J. (2014). *Adam: A Method for Stochastic Optimization*, pp. 1–15. <https://doi.org/10.1145/1830483.1830503>
- [7] Kraffka, K., Khosla, A., & Kellnhofer, P. (n.d.). *Eye Tracking for Everyone*.
- [8] Kristian, Y., Takahashi, H., Purnama, I.K.E., Yoshimoto, K., Setiawan, E.I., Hanindito, E., & Purnomo, M.H. (2017). A Novel Approach on Infant Facial Pain Classification Using Multi Stage Classifier and Geometrical-textural Features Combination. *IAENG International Journal of Computer Science*, 44(1), pp. 112–121.
- [9] Meyer, A., Böhme, M., Martinetz, T., & Barth, E. (2006). *A Single-Camera Remote Eye Tracker* (pp. 208–211). Berlin: Springer, Heidelberg. https://doi.org/10.1007/11768029_25
- [10] Morimoto, C.H., Amir, A., & Flickner, M. (n.d.). Detecting Eye Position and Gaze from a Single Camera and 2 Light Sources. In *Object Recognition Supported by User Interaction for Service Robots*. *IEEE Comput. Soc.*, Vol. 4, pp. 314–317. <https://doi.org/10.1109/ICPR.2002.1047459>
- [11] Setyati, E., Kristian, Y., Pranoto, Y.M., & Alexandre, D. (2013). Augmented Reality 3D Eyeglasses Frame Simulator Using Active Shape Model and Real Time Face Tracking. *2013 Eleventh International Conference on ICT and Knowledge Engineering*, pp. 1–5. <https://doi.org/10.1109/ICTKE.2013.6756272>
- [12] Sheela, S.V, Vijaya, P.A., & College, B.M.S. (2011). *An Appearance Based Method for Eye Gaze Tracking*, 7(8), pp. 1194–1203.
- [13] Shih, S.W., & Liu, J. (2004). A Novel Approach to 3-D Gaze Tracking Using Stereo Cameras. *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, 34(1), pp. 234–245. <https://doi.org/10.1109/TSMCB.2003.811128>
- [14] Smith, B.A., Yin, Q., Feiner, S.K., & Nayar, S. K. (2013). Gaze Locking: Passive Eye Contact Detection for Human-Object Interaction. *ACM Symposium on User Interface Software and Technology*, pp. 271–280. <https://doi.org/10.1145/2501988.2501994>
- [15] Stephenson, G.M., & Rutter, D.R. (1970). Eye-Contact, Distance and Affiliation. *British Journal of Psychology*, 61(3), 385–393. <https://doi.org/10.1111/j.2044-8295.1970.tb01257.x>
- [16] Valenti, R., & Gevers, T. (2008). Accurate Eye Center Location and Tracking Using Isophote Curvature. *26th IEEE Conference on Computer Vision and Pattern Recognition, CVPR*. <https://doi.org/10.1109/CVPR.2008.4587529>
- [17] Valstar, M., Martinez, B., Binfa, X., & Pantic, M. (2010). Facial Point Detection Using Boosted Regression and Graph Models. *Proceedings of the IEEE Computer*

- Society Conference on Computer Vision and Pattern Recognition*, pp. 2729–2736. <https://doi.org/10.1109/CVPR.2010.5539996>
- [18] Villanueva, A., Cabeza, R., & Porta, S. (2006). Eye Tracking: Pupil Orientation Geometrical Modeling. *Image and Vision Computing*, 24(7), pp. 663–679. <https://doi.org/10.1016/j.imavis.2005.06.001>
- [19] Villanueva, A., Cabeza, R., & Porta, S. (2007). Gaze Tracking System Model Based On Physical Parameters. *International Journal of Pattern Recognition and Artificial Intelligence*, 21(05), pp. 855–877. <https://doi.org/10.1142/S0218001407005697>
- [20] Vrânceanu, R., Florea, C., Florea, L., & Vertan, C. (2013). NLP EAC recognition by component separation in the eye region. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 8048 LNCS(PART 2), pp. 225–232. https://doi.org/10.1007/978-3-642-40246-3_28
- [21] LeCun, Y. (1990). *Handwritten Digit Recognition with a Back-Propagation Network*.
- [22] Zhang, X., Yusuke, Y., Fritz, M., & Bulling, A. (2015). *Appearance-Based Gaze Estimation in the Wild*, 1–10.
- [23] Zhu, X., & Ramanan, D. (2012). Face Detection, Pose Estimation, and Landmark Localization in the Wild. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2879–2886. <https://doi.org/10.1109/CVPR.2012.6248014>