

# Deteksi Glaukoma Menggunakan Metode Convolutional Neural Network dan Grabcut Segmentation

1<sup>st</sup> Dicky Hidayat  
Fakultas Informatika  
Universitas Telkom  
Bandung, Indonesia  
dickyhdyt@students.telkomuniversity.ac.id

2<sup>nd</sup> Wikky Fawwaz Al Maki  
Fakultas Informatika  
Universitas Telkom  
Bandung, Indonesia  
wikkyfawwaz@telkomuniversity.ac.id

## Abstrak

Glaukoma adalah jenis gangguan yang menyerang penglihatan. Glaukoma terjadi karena kerusakan saraf optik yang dapat menyebabkan kebutaan. Cara yang dapat dilakukan untuk mendeteksi glaukoma adalah melalui gambar retina. Ada banyak cara untuk memproses gambar retina sebelum dapat mendeteksi glaukoma. Proses ini sangat penting karena dapat mempengaruhi tingkat keberhasilan sistem pendeteksian glaukoma. Dalam penelitian ini, kami menggunakan Convolution Neural Network sebagai metode klasifikasi. Gambaran retina dibagi menjadi dua kelas, yaitu glaukoma positif dan negatif. kemudian kami menerapkan metode *Contrast Limited Adaptive Histogram Equalization* (CLAHE) dan segmentasi *Grabcut*. Hasil akurasi tertinggi dengan menggunakan data uji pada percobaan ini adalah 75.71%, *precision* sebesar 75.47%, *recall* sebesar 76.19%, dan *F1-score* sebesar 75.82% untuk arsitektur CNN InceptionV3.

Kata kunci: *convolutional neural network*, CLAHE, pemrosesan gambar, segmentasi grabcut

## Abstract

*Glaucoma is a type of disorder that attacks vision. Glaucoma occurs due to damage to the optic nerve which can lead to blindness. The way that can be done to detect glaucoma is through retinal images. There are many ways to process retinal images before being able to detect glaucoma. This process is very important because it can affect the success rate of glaucoma detection system. In this study, we used the Convolution Neural Network as a classification method. Retinal images are divided into two classes, namely positive and negative glaucoma. then we apply the Contrast Limited Adaptive Histogram Equalization (CLAHE) method and Grabcut segmentation. The results of the highest accuracy using test data in this experiment were 75.71%, precision was 75.47%, recall was 76.19%, and F1-score was 75.82% for InceptionV3 architecture.*

*Keywords: convolutional neural network, CLAHE, image processing, grabcut segmentation*

## I. PENDAHULUAN

### A. Latar Belakang

Glaukoma adalah jenis gangguan yang menyerang penglihatan. Glaukoma terjadi karena kerusakan saraf optik yang dapat menyebabkan kebutaan. Pada tahun 2020 76 juta orang di dunia menderita glaukoma dan diperkirakan pada tahun 2040 akan meningkat menjadi 111,8 juta orang [1]. Di Indonesia, 65.774 orang menderita glaukoma pada tahun

2015 dan meningkat menjadi 427.091 orang pada tahun 2017 [2]. Pesatnya perkembangan teknologi dapat membantu dalam bidang kesehatan. Gambar retina mata dapat digunakan untuk mendeteksi glaukoma.

Citra retina memiliki informasi yang sangat penting karena citra retina juga dapat mendeteksi berbagai penyakit seperti Retinopati Diabetik, Katarak, dan Glaukoma. Pengolahan citra diperlukan agar sistem dapat dengan mudah mendeteksi penyakit berdasarkan karakteristik tertentu pada citra retina. Pengolahan citra retina dapat berupa transformasi citra. *Gray Level Co-occurrence Matrix* (GLCM) digunakan dalam ekstraksi fitur tekstur yang mencakup enam fitur yaitu homogenitas, kontras, energi, korelasi, entropi, dan standar deviasi yang diekstraksi dari citra retina. Teknik ini memperoleh akurasi sebesar 72% [3]. Segmentasi citra juga dapat digunakan untuk mendeteksi glaukoma. Contoh segmentasi citra retina adalah segmentasi *Optic Cup*, *Optic Disk*, atau *Blood Vessel*. Teknik ini memperoleh akurasi sebesar 80% [4].

Tujuan penulisan ini adalah untuk mendeteksi glaukoma menggunakan citra retina menggunakan metode klasifikasi *Convolutional Neural Network* (CNN) menggunakan teknik pengolahan citra *Contrast Limited Adaptive Histogram Equalization* (CLAHE) dan segmentasi gambar *Grabcut*.

### B. Topik dan Batasannya

Berdasarkan latar belakang yang sudah dijabarkan diatas, permasalahan yang diangkat dalam tugas akhir ini, yaitu membangun sistem yang dapat mendeteksi Glaukoma dengan menggunakan gambar hasil scan *Optical Coherence Tomography* (OCT) retina mata menggunakan *Contrast Limited Adaptive Histogram Equalization* (CLAHE) untuk meningkatkan kontras gambar, *Grabcut* untuk segmentasi gambar, dan *Convolutional Neural Network* (CNN) sebagai metode klasifikasi. Adapun batasan-batasan yang diterapkan dalam permasalahan ini antara lain adalah

1. Dataset yang digunakan adalah dataset *open source* yang dapat ditemukan di situs Kaggle, dataset ini Bernama "Glaucoma Detection".
2. Dataset yang digunakan terdiri dari 141 data latih untuk kelas "*Glaucoma Positive*", 404 data latih untuk kelas "*Glaucoma Negative*",

27 data uji untuk kelas “*Glaucoma Positive*”, dan 78 data uji untuk kelas “*Glaucoma Negative*”.

3. Dataset merupakan hasil scan mata orang berumur diatas 60 tahun.
4. Hanya menggunakan dua akelas klasifikasi, yaitu “*Glaucoma Negative*” dan *Glaucoma Positive*”.
5. Hanya menggunakan 4 arsitektur *Convolutional Neural Network* (CNN), yaitu *EfficientNetB1*, *EfficientNetB2*, *InceptionV3*, dan *ResNet50v*.
6. Khusus untuk arsitektur *EfficientNetB1* dan *EfficientNetB2* dilakukan dua percobaan tambahan, yaitu menggunakan bobot (*weight*) Imagenet dan noisy-student.

### C. Tujuan

Tujuan dari penelitian ini adalah untuk membangun sistem yang dapat mendeteksi Glaukoma menggunakan *Contrast Limited Adaptive Histogram Equalization* (CLAHE) dan segmentasi gambar *Grabcut* sebagai teknik yang digunakan dalam pra-pemrosesan gambar dan *Convolutional Neural Network* (CNN) sebagai metode klasifikasi.

### D. Organisasi Tulisan

Laporan tugas akhir ini terdiri dari lima bagian pembahasan. Pada bagian pertama menjelaskan pendahuluan, bagian kedua menjelaskan studi terkait, bagian ketiga menjelaskan sistem yang dibangun, bagian keempat menjelaskan hasil dan evaluasi, dan bagian kelima menjelaskan kesimpulan yang diambil dalam tugas akhir ini.

## II. KAJIAN TEORI

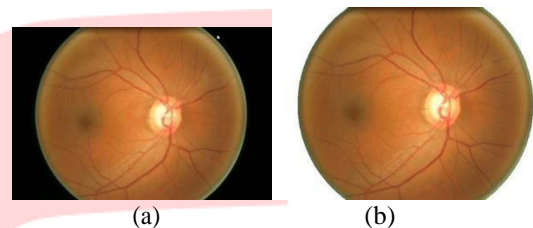
Kasus deteksi atau klasifikasi citra memerlukan pemrosesan tambahan. Pra-pemrosesan citra (*Image preprocessing*) merupakan langkah dasar yang harus dilakukan sebelum mengklasifikasikan citra atau citra. Operasi yang dilakukan pada image preprocessing adalah mengurangi *noise*, meningkatkan kontras, menghaluskan gambar, atau segmentasi gambar [5]. Proses pengklasifikasian citra atau image adalah dengan menyiapkan data latih dan data uji. Kemudian pra-pemrosesan citra (*Image preprocessing*) adalah proses mengubah data mentah menjadi format yang dapat digunakan untuk melatih model. Untuk data citra, transportasi dasar yang dapat dilakukan antara lain cropping, filtering, memutar, atau membalik [6].

*Convolutional Neural Network* (CNN) merupakan pengembangan dari metode sebelumnya yaitu *Artificial Neural Networks* (ANN). *Convolutional Neural Network* (CNNs) mirip dengan *Neural Network* tradisional karena terdiri dari neuron yang mengoptimalkan diri melalui pembelajaran [7]. Namun, kedua metode ini memiliki mekanisme proses yang hampir sama. Bentuk arsitektur *Convolutional Neural Network* (CNN) mirip dengan neuron atau sel otak manusia. CNN dilatih untuk dapat mengolah

informasi citra yang detail untuk proses deteksi atau klasifikasi [8]. Dalam makalah ini, CNN digunakan untuk mendeteksi glaukoma menggunakan citra retina. Arsitektur yang digunakan adalah *ResNet50V2*, *InceptionV3*, *EfficientNetB1* dan *EfficientNetB2*.

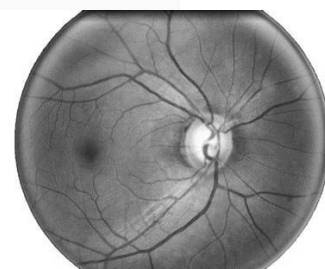
### A. Preprocessing

Langkah-langkah yang dilakukan adalah mengubah background citra menjadi putih, CLAHE, segmentasi *Grabcut*, augmentasi data, dan *data splitting*. Tahap pertama yang dilakukan adalah merubah warna *background* menjadi warna putih. *Background* gambar diubah menjadi putih untuk menghindari peningkatan kontras CLAHE pada *background* yang dapat menyebabkan timbulnya *noise* pada gambar.



Gambar 1. (a) Gambar Asli (b) Gambar setelah diubah menggunakan background putih

CLAHE adalah metode peningkatan citra yang digunakan untuk memperbesar citra [9]. Dalam metode ini, fungsi peningkatan kontras diterapkan pada semua piksel tetangga dan fungsi transformasi [10]. CLAHE adalah proses meningkatkan kontras dalam gambar dan mengubahnya menjadi gambar skala abu-abu dengan kontras yang lebih baik. Metode ini dapat diterapkan pada citra grayscale dan RGB dengan output berupa citra grayscale.

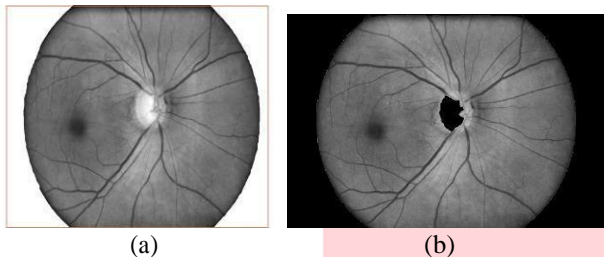


Gambar 2. Gambar setelah CLAHE

*Grabcut* adalah metode untuk mengelompokkan bagian gambar yang termasuk kedalam kelompok *foreground* gambar agar terpisah dari *background* [11]. Sederhananya *Grabcut* memisahkan *foreground* dari *background*. Langkah algoritma *Grabcut*:

1. Memperkirakan distribusi warna antara *foreground* dan *background* dengan *Gaussian Mixture Model* (GMM).
2. Membuat kotak pada area gambar yang akan disegmentasi.
3. Membangun bidang pada piksel gambar dan memberi label apakah termasuk *foreground* atau *background* pada kotak yang telah dibuat.
4. Menerapkan *Grabcut* berdasarkan label *foreground* dan *background*.

Ekstrasi ciri citra retina mata pada segmentasi grabcut dalam penelitian ini dibagi menjadi dua kelompok yaitu gambar yang menjadi latar depan (*foreground*) dan latar belakang (*background*). Ekstrasi ciri meliputi gambar fundus retina sebagai latar depan (*foreground*) dan gambar bagian optic cup retina sebagai latar belakang (*background*). Inisiasi kotak yang digunakan sebagai daerah segmentasi memiliki parameter [100, 10, 2872, 2028] dengan penjelasan [koordinat x, koordinat y, lebar kotak, tinggi kotak].



Gambar 3 (a) Ilustrasi penerapan kotak untuk segmentasi grabcut. (b) Gambar setelah Grabcut

Setelah inisiasi kotak, tahap selanjutnya adalah melakukan distribusi piksel dalam distribusi piksel akan menghasilkan dua node yaitu *source node* dan *sink node*. Setiap piksel yang termasuk latar depan (*foreground*) akan dihubungkan dengan *source node* dan Setiap piksel yang termasuk latar belakang (*background*) akan dihubungkan dengan *sink node*. Tahap terakhir adalah melakukan *cutting* berdasarkan *source node* dan *sink node*.

*Data splitting* digunakan untuk memisahkan data dalam proses pelatihan. Secara umum dataset dipisahkan menjadi tiga bagian, yaitu data latih, data validasi, dan data uji. karena dataset sudah menyediakan data latih dan data uji, maka data validasi dapat diambil dari data latih dengan konfigurasi 20% dari data latih. Hasil akhir pemisahan data adalah 437 citra untuk data latih, 108 citra untuk data validasi, dan 105 citra untuk data uji.

Augmentasi data adalah proses memanipulasi bentuk gambar. Augmentasi data adalah cara untuk menambahkan variasi ke gambar tanpa harus menambahkan data baru. dalam penelitian ini, kami menggunakan library TensorFlow yaitu ImageDataGenerator, dan menggunakan berbagai augmentasi data seperti pergeseran lebar, pergeseran ketinggian, zoom, rotasi, flip horizontal, dan flip vertical.

Tabel 1 – Deskripsi Augmentasi Data

No	Data Augmentation	Value
1	Width_shift_range	0.2
2	Height_shift_range	0.2
3	Zoom_range	0.2
4	Rotation_range	30
5	Horizontal_flip	True
6	Vertical_flip	True

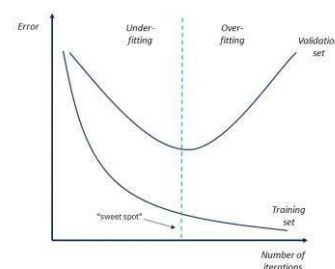
## B. Model CNN

*Convolutional Neural Network* (CNN) memiliki tiga lapisan utama dalam arsitekturnya, yaitu *convolution layer*, *pooling layer*, dan *fully-connected layer* [12]. Dalam penelitian ini, kami menggunakan 4 model arsitektur CNN seperti InceptionV3, ResNet50V2, EfficientNetB1, dan EfficientNetB2. 4 model arsitektur CNN ini diuji menggunakan parameter yang sama: bobot (*weight*), *custom top layer*, dan *optimizer*.

Dalam membangun model CNN ada kemungkinan model yang dibuat menyebabkan *overfitting*. *Overfitting* adalah salah satu masalah yang sering muncul dalam kasus machine learning. Tanda *overfitting* adalah ketika akurasi pelatihan memiliki nilai tinggi tetapi akurasi pengujian rendah. Ada beberapa metode yang dapat mengantisipasi terjadinya *overfitting*. Ada tiga metode yang digunakan dalam penelitian ini yang pertama adalah menggunakan *L2-regularization*. *L2-regularization* mengatasi *overfitting* dengan menghukum parameter bobot terhadap fungsi kerugian.  $\lambda w_i$  ditambahkan ke *loss function* untuk setiap bobot  $w_i$  dalam arsitektur CNN. *L2-regularization* disebut juga dengan peluruhan bobot karena *L2-regularization* dapat mencegah pertumbuhan nilai bobot sehingga tidak bertambah terlalu besar [13].

$$\Omega(W^{(p)})_{l2} \equiv \|W\|_2^2 \quad (1)$$

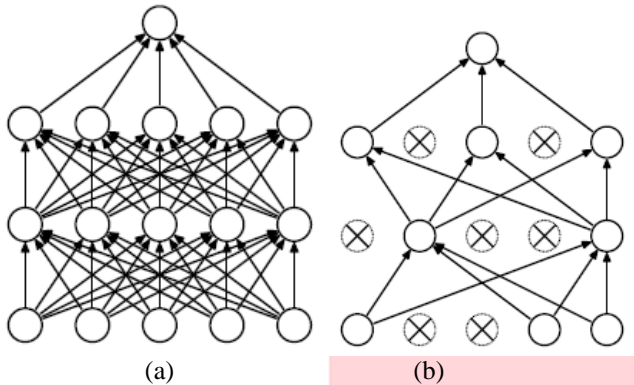
Metode yang kedua adalah menggunakan *early stopping*. *Early stopping* adalah teknik yang digunakan untuk menghentikan sistem dalam proses pelatihan ketika mengalami kondisi tertentu. Misalnya Ketika nilai loss secara tiba-tiba naik secara signifikan, dengan *early stopping* maka proses *training* akan langsung dihentikan. *Epoch* merupakan salah satu parameter penting dalam proses pelatihan. *Epoch* adalah parameter yang digunakan untuk menentukan berapa kali model akan dilatih. Melakukan iterasi yang terlalu banyak dalam pelatihan dapat menyebabkan *overfitting* dan pelatihan yang terlalu sedikit akan menyebabkan *underfitting* sehingga pemilihan jumlah epoch harus diperhatikan. Berhenti lebih awal bisa menjadi solusi untuk masalah ini. Keras mendukung penghentian awal yang disebut *callback*. Dalam penelitian ini, parameter yang digunakan sebagai pemantau penghentian awal adalah *validasi\_akurasi*. sistem akan menghentikan pelatihan ketika nilai *validasi\_akurasi* tidak meningkat setelah 10 iterasi dari *akurasi\_validasi* tertinggi.



Gambar 4. Ilustrasi grafik overfitting dan underfitting

Metode yang ketiga adalah menggunakan *Dropout layer*. *Dropout* adalah teknik regularisasi dalam jaringan saraf dengan menghilangkan neuron secara acak di lapisan jaringan yang tersembunyi atau terlihat. Menghapus beberapa

neuron berarti memutuskan jaringan yang masuk dan keluar dengan neuron tetangga [14]. Selain mencegah overfitting, dropout dapat mempercepat proses pelatihan, karena dropout mengurangi penggunaan neuron di jaringan saraf, sehingga membuat pelatihan lebih cepat. Nilai dropout yang digunakan berkisar antara 0 sampai 1. *L2-regularization*, *early stopping*, dan *dropout* merupakan teknik yang digunakan dalam penelitian ini untuk mencegah overfitting.



Gambar 5. (a) neural network standar (b) neural network setelah menggunakan Dropout

### III. METODE

Pada sistem deteksi glaucoma ini dibagi menjadi tiga tahap, yaitu tahap persiapan, tahap pelatihan dan tahap evaluasi. Pada tahap persiapan merupakan tahap awal untuk mempersiapkan dataset untuk proses selanjutnya yaitu proses pelatihan. Proses pada tahap persiapan adalah proses *preprocessing* gambar yang digunakan untuk memodifikasi gambar asli. Proses pelatihan merupakan proses dimana model sistem yang telah dibuat akan dilatih menggunakan data latih. Proses evaluasi merupakan proses untuk menilai seberapa baik pelatihan yang telah dilakukan.



Gambar 6. Flowchart Sistem

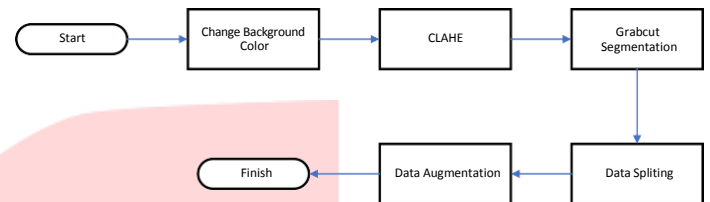
#### A. Dataset

Dataset adalah kumpulan data yang berisi informasi yang akan digunakan untuk tujuan tertentu, misalnya untuk memprediksi, mendeteksi, atau mengklasifikasikan. Data dalam dataset dapat berupa teks, gambar atau video. Dataset tersebut akan diproses sehingga dapat dikenali oleh komputer. Pada penelitian ini dataset yang digunakan adalah data citra dengan tujuan untuk mendeteksi glaukoma.

Gambar retina yang digunakan diambil dari situs Kaggle [15]. Citra retina merupakan hasil scan OCT (Optical Coherence Tomography) yang berisi 650 citra dengan 2 kelas yaitu glaukoma positif dan glaukoma negatif. Dataset memiliki ukuran 3072 x 2048 piksel dengan format jpg, memiliki basis warna RGB, dan memiliki latar belakang hitam, latar belakang hitam akan diubah menjadi putih sebelum dilakukan pra-pemrosesan. dataset dipisahkan menjadi data latih dan data uji, dengan total 545 gambar untuk data latih dan 105 gambar untuk data uji.

#### B. Preprocessing

Langkah-langkah yang dilakukan adalah mengubah background citra menjadi putih, CLAHE, segmentasi *Grabcut*, augmentasi data, dan *data splitting*. Tahap pertama yang dilakukan adalah merubah warna *background* menjadi warna putih. *Background* gambar diubah menjadi putih untuk menghindari peningkatan kontras CLAHE pada *background* yang dapat menyebabkan timbulnya *noise* pada gambar. Tahap kedua adalah CLAHE. Tahap ketiga adalah segmentasi *Grabcut*. Tahap keempat adalah *data splitting*. Tahap terakhir adalah augmentasi data.



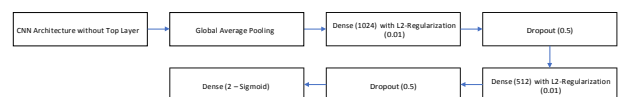
Gambar 7. Flowchart Preprocessing

#### C. Pelatihan

Pelatihan dilakukan dengan menggunakan 4 model arsitektur CNN, yaitu *ResNet50v2*, *InceptionV3*, *EfficientNetB1*, dan *EfficientNetB2*. Model *EfficientNetB1* dan *EfficientNetB2* dilatih dua kali menggunakan bobot yang berbeda, yaitu *imagenet* dan *noise-students*. *Noisy Student Training* termasuk dalam pembelajaran semi terawasi. *Noisy Student* merupakan perluasan dari ide pelatihan diri dan penyulingan dengan menggunakan model siswa yang sama atau lebih besar dan menambahkan *noise* pada *students* selama proses pembelajaran [16]. *Noisy Student* digunakan untuk model *EfficientNetB0* hingga *EfficientNetB7*.

Parameter yang digunakan dalam proses pelatihan adalah epoch maksimal 50 dan 10 batch size. Komputasi pelatihan memerlukan waktu antara 85 – 100 detik untuk 1 epoch. Optimizer yang digunakan adalah Adam Optimizer dengan learning rate sebesar 0,00001. Spesifikasi hardware yang digunakan dalam pelatihan ini adalah Inter Core I7 7700hq, GTX 1050 mobile (4 Gb), dan RAM 16 Gb. Spesifikasi software yang digunakan adalah Anaconda Navigator, Python 3.7.9, Tensorflow GPU versi 2.5.0, dan Keras versi 2.4.3. Software pendukung Tensorflow GPU versi 2.4.0 adalah CUDA Toolkit versi 11.0.3 dan NVIDIA cuDNN versi 8.2.1.

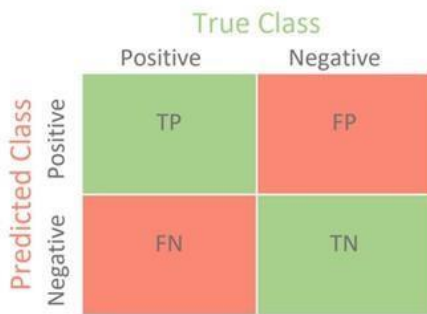
Untuk mempercepat pelatihan, penelitian ini menggunakan komputasi dengan Graphical Processing Unit (GPU), karena core di GPU lebih banyak daripada core di Central Processing Unit (CPU) sehingga komputasi dengan GPU akan lebih cepat. Jumlah core pada Intel Core I7 7700hq memiliki 4 core sedangkan jumlah core CUDA pada mobile GTX 1050 memiliki 640 core.



Gambar 8. Custom Top Layer

IV. HASIL DAN PEMBAHASAN

Setelah pelatihan model, langkah selanjutnya adalah mengevaluasi metrik. Secara umum, metrik yang dievaluasi adalah Accuracy, Precision, Recall, F1, Score, dan Loss. Nilai akurasi menjadi masalah utama karena dapat menentukan akurasi klasifikasi yang telah dibangun pada model sebelumnya. Semakin tinggi nilai akurasi, semakin akurat prediksi klasifikasi. Confusion matrix digunakan untuk pengukuran nilai akurasi, *precision*, *recall*, dan *F1-score*. Berikut persamaan perhitungan nilai akurasi, *precision*, *recall*, dan *F1-score*.



Gambar 9. Confusion Matrix

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (2)$$

$$Precision = \frac{TP}{TP + FP} \quad (3)$$

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

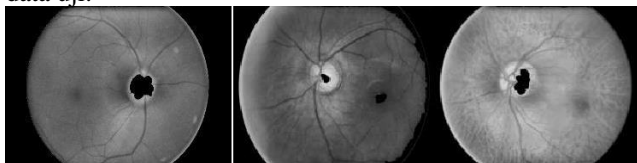
$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (5)$$

A. Hasil Pengujian

Pengukuran performa sistem yang dibangun menggunakan metrik akurasi, precision, recall, dan F1-score. Pengujian yang ditampilkan merupakan hasil performa dari *training* dan *testing*.

1. Preprocessing

Pada tahap ini, proses preprocessing yang pertama kali dilakukan adalah merubah warna *background* menggunakan warna putih. Perubahan *background* menggunakan aplikasi “Adobe Photoshop”. Tahap kedua adalah melakukan peningkatan citra dengan teknik CLAHE. Tahap ketiga adalah melakukan *data splitting*. *Data splitting* membagi data menjadi tiga, yaitu data latih, data validasi dan data uji dengan pembagian 437 data latih, 108 data validasi, dan 105 data uji.



Gambar 10. Contoh hasil grabcut

B. Analisis Hasil Pengujian

Pada tahap ini peneliti akan memberikan hasil dari penelitian yang telah dilakukan. Hasil penelitian akan

menampilkan nilai akurasi, *precision*, *recall*, *loss*, dan *F1-score*. Untuk penghitungan nilai *loss* menggunakan fungsi tambahan tergantung dengan masalah dan jumlah kelas dalam *dataset*. Fungsi *loss* yang digunakan pada kasus ini adalah *binary crossentropy*, karena *dataset* memiliki dua kelas *positive* dan *negative*.

Tabel 2 – Hasil *Training*

Model	Weight	Acc	Loss	Precision	Recall	F1
EffNet 1	Imagene t	82.84 %	14.9 3	82.1%	83.98 %	83.02 %
EffNet 1	Noisy	78.49 %	13.7 8	78.75%	78.03 %	78.38 %
EffNet 2	Imagene t	81.12 %	14.4 5	80.22%	82.61 %	81.39 %
EffNet 2	Noisy	77.69 %	14.6 9	77.63%	77.8%	77.71 %
IncV3	Imagene t	81.12 %	18.2 9	81.05%	81.24 %	81.14 %
ResNet	Imagene t	83.41 %	14.5 5	83.49%	83.30 %	83.39 %

Informasi tambahan untuk Tabel II dan Tabel III: EffNet1 adalah model EfficientNetB1, EffNet2 adalah model EfficientNetB1, IncV3 adalah model InceptionV3, noise adalah bobot siswa yang berisik, Acc adalah nilai Accuracy, dan F1 adalah nilai F1-score.

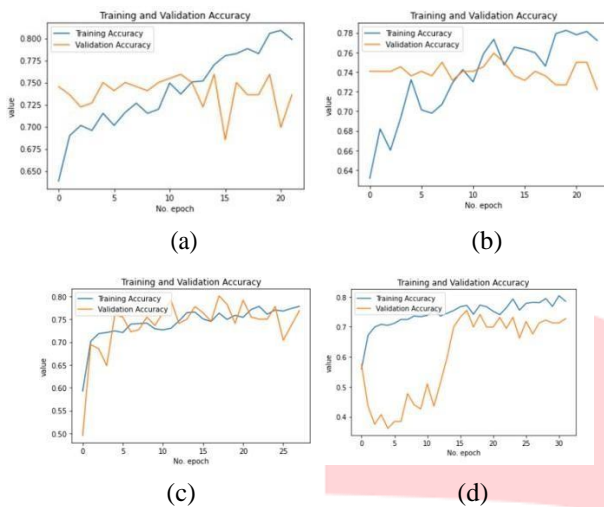
Berdasarkan Tabel 2, dapat dilihat bahwa model arsitektur CNN yang memiliki nilai akurasi tertinggi pada proses *training* adalah model ResNet50v2 dengan nilai akurasi sebesar 83.41%. Namun model yang dihasilkan oleh model arsitektur CNN lainnya tidak jauh berbeda, yaitu sekitar 77% - 83% dengan akurasi tertinggi didapat oleh model ResNet50v2 dan akurasi terendah didapat oleh model EfficientNetB2 dengan bobot *noisy-student*.

Tabel 3 – Hasil *Testing*

Model	Weight	Acc	Loss	Precision	Recall	F1
EffNet 1	Imagene t	74.76 %	15.0 4	74.07%	76.19 %	75.11 %
EffNet 1	Noisy	73.81 %	13.8 5	74.04%	73.33 %	73.68 %
EffNet 2	Imagene t	73.81 %	14.5 8	73.58%	74.29 %	73.93 %
EffNet 2	Noisy	71.9%	14.7 1	72.12%	71.43 %	71.77 %
IncV3	Imagene t	75.71 %	18.4	74.47%	76.19 %	75.82 %
ResNet	Imagene t	75.24 %	17.6 8	74.77%	76.19 %	75.47 %

Berdasarkan Tabel 3, dapat dilihat bahwa model arsitektur CNN yang memiliki nilai akurasi tertinggi pada proses *testing* adalah model InceptionV3 dengan nilai akurasi sebesar 75.71%. Namun model yang dihasilkan oleh model arsitektur CNN lainnya tidak jauh berbeda, yaitu sekitar 71% - 75% dengan akurasi tertinggi didapat oleh model InceptionV3 dan akurasi terendah didapat oleh model EfficientNetB2 dengan bobot *noisy-student*. Meskipun model InceptionV3 memiliki akurasi tertinggi, namun model ini menghasilkan nilai *loss* yang paling tinggi diantara model lainnya. Untuk model EfficientNetB1 dan EfficientNetB2 dengan penggunaan bobot yang berbeda memperoleh hasil

yang tidak jauh berbeda. Pada Tabel 2 dan Tabel 3, dapat dilihat bahwa penggunaan bobot imagenet untuk model EfficientNetB1 dan EfficientNetB2 memperoleh hasil yang lebih baik dibandingkan bobot *noisy-student*.



Gambar 11 – Perbandingan akurasi pada proses training (a) ResNet50v2

(b) InceptionV3 (c) EfficientNetB1 (d) EfficientNetB2

Berdasarkan Gambar 11, pada model ResNet50v2 *validation accuracy* mengalami penurunan yang signifikan pada iterasi ke 14. Model ResNet50v2 mengalami penurunan dan kenaikan yang cukup tinggi yaitu sekitar 10% pada iterasi diatas 14. Pada model InceptionV3 model mengalami performa yang stabil, namun proses pembelajaran pada sistem tidak mengalami kemajuan. Pada model EfficientNetB1, model mengalami performa yang stabil dan pembelajaran sistem mengalami kemajuan yang ditandai naiknya nilai *validation accuracy* yang awalnya sekitar 50% pada iterasi 1, kemudian terus naik hingga iterasi 5 dan stabil setelah iterasi 5. Pada model EfficientNetB2, model mengalami penurunan pada awal pembelajaran dan kemudian naik pada iterasi ke 12.

## V. KESIMPULAN

Berdasarkan hasil penelitian yang telah dilakukan, performa sistem yang dibuat memperoleh akurasi terbaik pada proses *testing* adalah 75.71%, *loss* sebesar 18.4, *precision* sebesar 75.47%, *recall* sebesar 75.47% dan *F1-score* sebesar 75.82%. Pada kasus ini model yang dibangun memiliki performa yang tidak terlalu baik, karena hanya memiliki akurasi tertinggi sebesar 75.82% pada proses *testing*. Penggunaan bobot yang berbeda pada model EfficientNetB1 dan EfficientNetB2 tidak memberikan perbedaan yang signifikan dengan bobot terbaik pada kasus ini adalah Imagenet.

Saran untuk penelitian selanjutnya, yaitu dapat memperbanyak jumlah data yang digunakan karena 650 gambar yang digunakan pada penelitian ini sangat sedikit untuk kasus yang memerlukan gambar yang sangat detail. Detail gambar sangat penting untuk kasus yang berkaitan dengan medis, karena detail gambar dapat mempengaruhi hasil deteksi penyakit. Selain itu, mungkin dapat menggunakan segmentasi gambar yang berbeda seperti *watershed* segmentasi atau *UNET segmentation*.

## REFERENSI

- [1]. Tham. Y. C., Li. X, Wong. T. Y, Quigley. H. A, Aung. T, dan Cheng. C. Y. 2014. Global prevalence of glaucoma and projections of glaucoma burden through 2040. a systematic review and meta-analysis. *Ophthalmology*. 2014 Nov, 121(11), 2081-90. Doi: 10.1016/j.ophtha.2014.05.013.
- [2]. Indonesia, 2019. INFODATIN Pusat Data dan Informasi Kementerian Kesehatan RI Situasi Glaukoma di Indonesia. Kementerian Kesehatan RI, Jakarta, Indonesia.
- [3]. D. Yadav, M. P. Sarathi dan M. K. Dutta. 2014. "Classification of glaucoma based on texture features using neural networks," 2014 Seventh International Conference on Contemporary Computing (IC3), 2014, pp. 109-112, doi: 10.1109/IC3.2014.6897157.
- [4]. Ahmad, M., M. P., Sarathi,, dan M. K., Dutta. (2016). Deteksi Penyakit Glaukoma pada Citra Fundus Retina Mata Menggunakan Adaptive Thresholding dan Support vector machine. *JURNAL TEKNIK ITS Vol.5, No. 2*. Doi: 10.12962/j23373539.v5i2.18929.
- [5]. S. Adatrao dan M. Mittal. 2016. "An analysis of different image preprocessing techniques for determining the centroids of circular marks using hough transform." 2016 2nd International Conference on Frontiers of Signal Processing (ICFSP). pp. 110-115. doi: 10.1109/ICFSP.2016.7802966.
- [6]. Tran. N. M., Mathieu. Sinn, Hoang. T. L, dan Martin. W. 2018. Automated Image Data Preprocessing with Deep Reinforcement Learning. [Online] Available at: <https://arxiv.org/abs/1806.05886> (Accessed on December 25, 2021).
- [7]. Keiron. O., dan Ryan, N. 2015. An Introduction to Convolutional Neural Networks. . [Online] Available at: <https://arxiv.org/abs/1511.08458>(Accessed on December 21, 2021).
- [8]. F. Iraldi and W. F. Al Maki. 2021 "Damage Classification on Roads Using Machine Learning," 2021 International Conference on Data Science and Its Applications (ICoDSA). pp. 151-156. doi: 10.1109/ICoDSA53588.2021.9617520.
- [9]. Purnawarman, M., F. Al Rafi, dan M. Lamsani. 2018. A Review: Contrast-Limited Adaptive Histogram Equalization (CLAHE) methods to help the application of face recognition. Third International Conference on Informatics and

- Computing (ICIC), pp. 1-6. Doi: 10.1109/IAC.2018.8780492.
- [10]. Garima. Y., S. Maheshwari, dan A. Agarwal. (2014). Contrast limited adaptive histogram equalization based enhancement for real time video system. 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI), pp. 2392-2397. Doi: 10.1109/ICACCI.2014.6968381.
- [11]. Rosebrock, A. 2020. OpenCV GrabCut: Foreground Segmentation and Extraction. [Online] Available at: <https://www.pyimagesearch.com/2020/07/27/open-cv-grabcut-foreground-segmentation-and-extraction/> [Accessed 28 Januari 2022].
- [12]. Tianmei. G, J. Dong, H. Li dan Y. Gao. 2017. Simple convolutional neural network on image classification. 2017 IEEE 2nd International Conference on Big Data Analysis (ICBDA). pp. 721-724' doi: 10.1109/ICBDA.2017.8078730.
- [13]. ASHWINI. R dan Shintal. . 2019. Deep Neural Network Regularization for Feature Selection in Learning-to-Rank. in IEEE Access, vol. 7, pp. 53988-54006. doi: 10.1109/ACCESS.2019.2902640.
- [14]. Nitish. S., Geoffrey. H, Alex. K, Ilya. S, dan Ruslan. S. 2014. Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, pp 1929-1958. [Online] Available at: <https://dl.acm.org/doi/10.5555/2627435.2670313> (Accessed on January 1, 2022).
- [15]. Kaggle, 2021 . Glaucoma Detection OCT Scans Retinal. [Online] Available at: [https://www.kaggle.com/sshikamaru/glaucoma-detection?select=Fundus\\_Train\\_Val\\_Data](https://www.kaggle.com/sshikamaru/glaucoma-detection?select=Fundus_Train_Val_Data). (Accessed on October 19, 2021).
- [16]. Qizhe, X., Minh-Thang. L, E. Hovy, dan Q. V. Le. 2020. Self-Training With Noisy Student Improves