

# Penerapan Metode *Generative Adversarial Network* dalam Peningkatan Kinerja Model *EfficientNet* untuk Mendeteksi *Age-related Macular Degeneration* pada Citra Fundus Retina Mata

---

Herardita Cahyaning Wulan | 2006568632

## Dosen Pembimbing

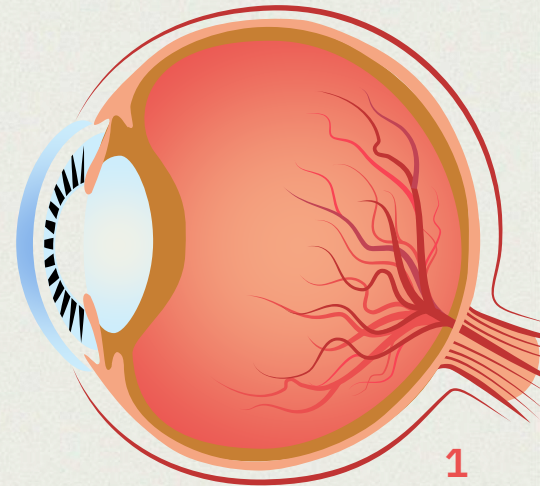
Prof. Alhadi Bustamam, S.Si., M.Si., Ph.D.

Devvi Sarwinda, S.Si., M.Kom.

## Dosen Penguji

Dr. Denny Riama Silaban, M.Kom.

Dra. Siti Aminah, S.Si., M.Kom.



# OUTLINE

**01**

Pendahuluan

**02**

Landasan Teori

**03**

Penerapan GAN  
pada EfficientNet

**04**

Simulasi &  
Analisis Hasil

**05**

Penutup



# 01

## Pendahuluan

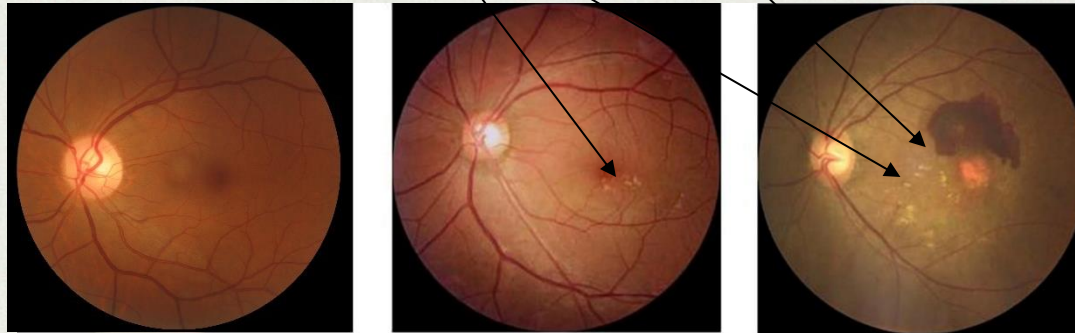


Latar Belakang  
Rumusan Masalah  
Tujuan Penelitian  
Metodologi Penelitian  
Batasan Masalah

# Latar Belakang

*Age-related macular degeneration* (AMD) adalah penyakit degeneratif pada makula yang menyebabkan gangguan penglihatan sentral pada orang lanjut usia.

AMD ditandai oleh adanya *drusen*, atrofi geografis, dan pelepasan epitel pigmen retina yang berkembang secara progresif (Gheorghe, Mahdi, & Musat, 2015).



(a) Normal

(b) Dry AMD

(c) Wet AMD

Sumber : Wu et al., 2022



# Latar Belakang

Secara **global**, prevalensi AMD mencapai **170 juta orang**, sementara di **Amerika Serikat** terdapat **sekitar 11 juta orang** yang didiagnosis mengalami AMD (Flaxel et al., 2020). Angka ini diperkirakan akan meningkat menjadi hampir **288 juta orang** pada tahun 2040. (Wong et al., 2014).



Menurut Kemenkes RI tahun 2018, AMD menjadi penyebab kebutaan terbesar ketiga, setelah katarak dan gangguan refraksi.

# Latar Belakang

Diagnosis AMD diperoleh melalui oftalmoskop, salah satunya dengan fotografi fundus retina. Dalam mendiagnosis AMD diperlukan keahlian dari dokter spesialis mata atau oftalmologis (Kadhim, Khan, & Mishra, 2022)



**Waktu lama, Risiko *human error*, subjektif**



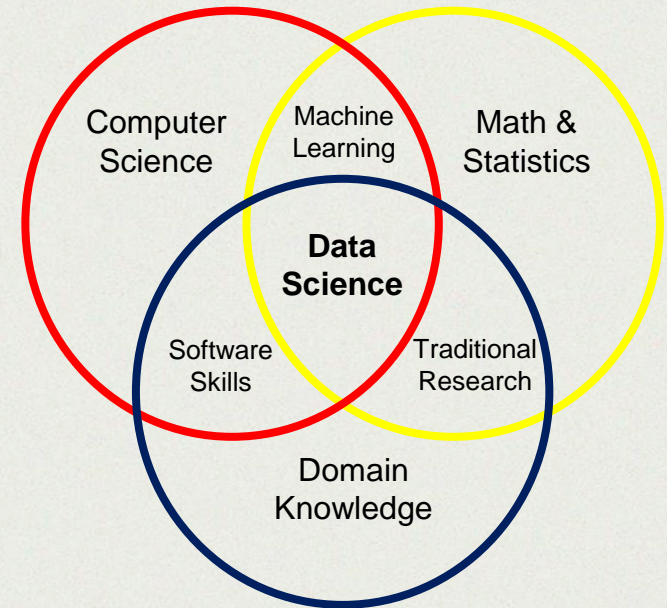
Sumber: storyset.com

→ Diperlukan pendekatan teknologi yang dapat membantu proses diagnosis penyakit secara lebih efisien dalam hal waktu dan menghasilkan akurasi yang baik



# Latar Belakang

Salah satu pendekatan yang menjanjikan adalah menggunakan sains data dan **artificial intelligence** (AI), secara khusus **machine learning** untuk mendeteksi dan mendiagnosis penyakit mata. Metode yang dapat diterapkan adalah **Generative Adversarial Network** (GAN) dan **Convolutional Neural Network** (CNN), arsitektur **EfficientNet** merupakan salah satu yang paling efektif pada tugas klasifikasi gambar serta efisien secara komputasional.



Sumber: Ruas *et al*, 2020

# Latar Belakang (Penelitian Terdahulu)

*Deep Convolutional GAN (DCGAN)* yang menggunakan CNN dalam arsitekturnya. Penggunaan CNN membantu menangkap informasi spasial gambar dengan lebih baik (Radford, Metz, dan Chintala, 2015)

*EfficientNet* untuk mendeteksi *diabetic retinopathy* melalui citra fundus mata didapat akurasi 94,02% (Maswood et al., 2020)

2014

2015

2019

2020

Generative Adversarial Network (GAN) untuk menghasilkan data baru yang menyerupai data asli (Goodfellow et al., 2014)

Arsitektur *EfficientNet* dengan konsep *compound scaling* yang mencoba menyelesaikan masalah peningkatan skala model dengan tetap meningkatkan akurasi pada model yang dibuat (Tan & Le, 2019)



# Latar Belakang (Penelitian Terdahulu)

DCGAN untuk memperoleh citra sintetis fundus untuk retina normal, glaukoma, miopia, dan *diabetic retinopathy* (Smaida, Yaroshchak, & El, 2021)

*EfficientNet* untuk mendeteksi AMD melalui citra OCT didapat akurasi 90,60% (Anam, Novamizanti, & Rizal, 2021)

2021

2023

*EfficientNet* untuk mendeteksi tumor otak melalui citra MRI dengan akurasi 98,80% (Razi, Bustamam, & Latifah, 2023)

DCGAN untuk memperoleh citra sintetis fundus untuk kelas retina normal dan stroke (Sasiyanto, Bustamam, & Sarwinda, 2023)

# Latar Belakang (Metode)

## ***Generative Adversarial Network (GAN)***

Menghasilkan data sintetis yang realistis dan menyerupai data asli untuk menyeimbangkan jumlah data pada masing-masing kelas.

## ***Convolutional Neural Network (CNN) – EfficientNet***

Melakukan klasifikasi citra fundus retina mata. Ide utama adalah menciptakan keseimbangan antara *width*, *depth*, dan *resolution* untuk mencapai kinerja yang lebih baik dengan parameter dan komputasi yang lebih sedikit.

### Keterbaruan Penelitian

Penggunaan GAN dalam meningkatkan kinerja *EfficientNet* untuk mendeteksi AMD melalui citra fundus retina mata



# Rumusan Masalah

Berdasarkan latar belakang di atas, berikut rumusan masalah pada skripsi ini:

1. Bagaimana menerapkan metode *Generative Adversarial Network* dalam augmentasi data citra fundus retina mata?
2. Bagaimana menerapkan model *EfficientNet* dalam mendeteksi *Age-related Macular Degeneration* pada citra fundus retina mata?
3. Bagaimana pengaruh penerapan metode *Generative Adversarial Network* dalam meningkatkan kinerja model *EfficientNet* untuk mendeteksi *Age-related Macular Degeneration* pada citra fundus retina mata berdasarkan nilai akurasi, *sensitivity*, dan *specificity*?

# Tujuan Penelitian

Berdasarkan rumusan masalah di atas, berikut tujuan penelitian skripsi ini:

1. Menerapkan metode *Generative Adversarial Network* untuk augmentasi data citra fundus retina mata.
2. Menerapkan model *EfficientNet* dalam mendeteksi penyakit *Age-related Macular Degeneration* pada citra fundus retina mata.
3. Menganalisis pengaruh penerapan metode *Generative Adversarial Network* terhadap kinerja model *EfficientNet* dalam mendeteksi penyakit *Age-related Macular Degeneration* pada citra fundus retina mata berdasarkan nilai akurasi, *sensitivity*, dan *specifity*.



# Batasan Permasalahan

1. Terdapat dua kelas yang dideteksi, yaitu normal dan AMD.
2. Data yang digunakan adalah data citra fundus retina mata yang diperoleh dari beberapa situs *opensource database*:
  - *Joint Shantou International Eye Center* (JSIEC), kelas normal dan AMD
  - *Structured Analysis of the Retina* (STARE), kelas AMD
  - *Retinal Fundus Multi-Disease Image Dataset* (RFMID 2.0), kelas AMD
  - *Ocular Disease Intelligent Recognition* (ODIR), kelas normal dan AMD
3. Tidak ada pemisahan citra fundus mata kiri dan mata kanan pada data dan diperumum sebagai citra fundus mata.
4. Augmentasi data citra fundus menggunakan metode *Deep Convolutional GAN* (DCGAN).

# Metodologi Penelitian

Jenis metodologi penelitian adalah **metodologi kuantitatif** dengan tahapan sebagai berikut:

## 1. Studi Literatur

Mengenai *age-related macular degeneration* (AMD), CNN, GAN, EfficientNet, dan teori pendukung lainnya

## 2. Pengumpulan Data

Mengumpulkan data sekunder berupa data citra fundus yang diperoleh dari *opensource database*. Terdapat 2 kelas, yaitu normal dan AMD berjumlah 324 dan 281 data.

## 3. Implementasi&Simulasi Model

Melakukan *preprocessing* dan augmentasi data dengan GAN. Merancang model CNN arsitektur *EfficientNet* dan melakukan simulasi menggunakan bahasa Python

## 4. Evaluasi Model

Mengevaluasi hasil kinerja model CNN arsitektur *EfficientNet* dalam mengklasifikasi penyakit AMD. Evaluasi berdasarkan nilai akurasi, *sensitivity*, dan *specifity*.



# 02

## Landasan Teori



Citra Digital  
*Preprocessing Data*  
*Generative Adversarial Network*  
*Convolutional Neural Network*  
Evaluasi Kinerja Model

# Citra Digital

Citra digital merupakan citra dua dimensi yang terbatas dan diskrit, di mana  $x$  dan  $y$  adalah koordinat bidang dan  $f(x, y)$  adalah tingkat keabuan citra pada setiap pasangan  $(x, y)$ . Setiap titik  $f(x, y)$  disebut piksel (Gonzalez & Wood, 2018).

Citra dapat direpresentasikan dalam matriks dengan  $a_{i,j} = f(i, j)$

$$A = \begin{bmatrix} a_{0,0} & a_{0,1} & \cdots & a_{0,N-1} \\ a_{1,0} & a_{1,1} & \cdots & a_{1,N-1} \\ \vdots & \vdots & & \vdots \\ a_{M-1,0} & a_{M-1,1} & \cdots & a_{M-1,N-1} \end{bmatrix}$$

Terdapat empat jenis citra, yaitu citra biner, **grayscale**, *black-and-white*, dan warna sejati (RGB) (Xia *et al.*, 2020)



# *Preprocessing Data*

## ***Centered Crop***

Citra akan dipotong sedemikian sehingga objek yang diobservasi berada tepat di tengah dan memiliki rasio 1:1.

## ***Resize***

Proses mengubah citra digital agar mendapatkan ukuran dimensi citra yang diinginkan. Pada penelitian ini, citra diubah ke ukuran 296x296 piksel.

## ***Contrast Limited Adaptive Histogram Equalization (CLAHE)***

- CLAHE merupakan teknik pemrosesan untuk meningkatkan kontras gambar.
- CLAHE membagi citra menjadi area kecil (*tile*). Setiap *tile* menerapkan *histogram equalization* secara terpisah yang memastikan penguatan kontras bersifat lokal (Musa, Rafi, & Lamsani, 2018).

# Preprocessing Data

## Contrast Limited Adaptive Histogram Equalization (CLAHE)

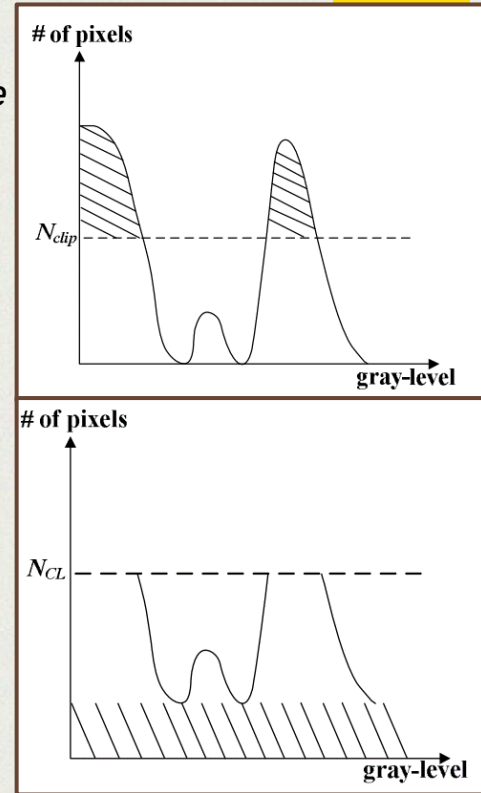
1. Membagi gambar menjadi area kecil (*tile*) dengan ukuran  $M \times N$ . Setiap *tile* diproses secara terpisah sehingga penguatan kontras bersifat lokal.
2. Menghitung histogram setiap *tile* untuk melihat distribusi piksel pada *tile*.
3. Memangkas histogram pada nilai tertentu ( $N_{clip}$ ). Piksel di atas  $N_{clip}$  akan dipangkas dan didistribusikan level keabuannya. Nilai rata-rata piksel untuk mendistribusikan level keabuan:

$$N_{aver} = \frac{N_{CR-XP} \times N_{CR-YP}}{N_{gray}}$$

Kemudian hitung nilai aktual *clip-limit* ( $N_{CL}$ ) dengan formula:

$$N_{CL} = N_{clip} \times N_{aver}$$

Jika jumlah piksel pada suatu level keabuan melebihi  $N_{CL}$ , maka piksel tersebut akan dipangkas.



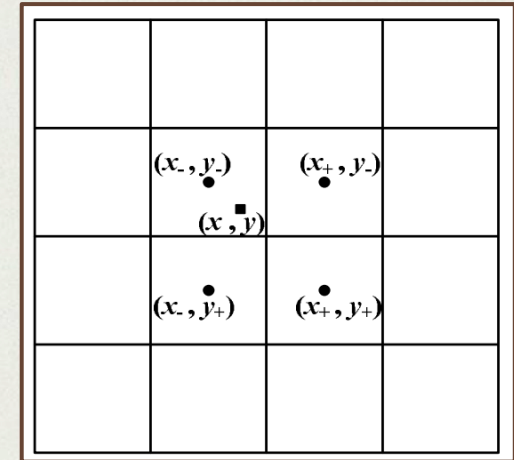
Sumber: Xu, Liu, & Chen, 2009 **18**



# Preprocessing Data

## *Contrast Limited Adaptive Histogram Equalization (CLAHE)*

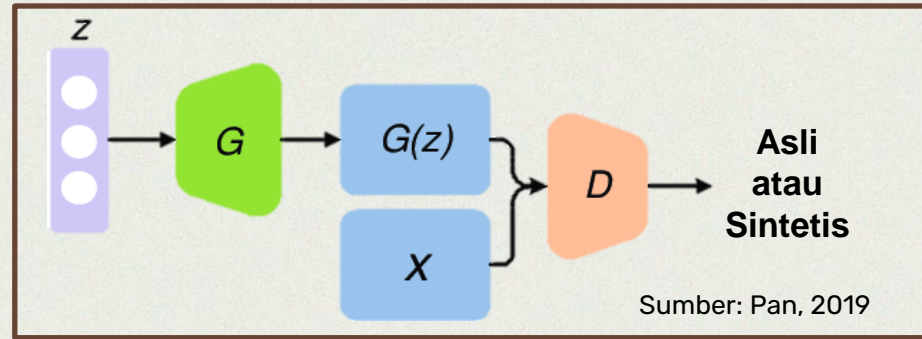
4. Memproses **histogram equalization** untuk setiap **tile** sehingga skala keabuan pada histogram lebih seimbang
5. Menganggap titik tengah pada **tile** sebagai titik sample
6. Menginterpolasi intensitas piksel dari titik sample ke empat titik di sekitar titik sampel



Sumber: Xu, Liu, & Chen, 2009

# *Generative Adversarial Network (GAN)*

Model generatif yang dirancang untuk menghasilkan sampel data baru yang menyerupai *dataset* yang diberikan.



- **Generator**  $G$  menangkap distribusi vektor *noise*  $z$  kemudian menghasilkan data baru.
- **Diskriminator**  $D$  memperkirakan probabilitas sampel yang berasal dari data asli ( $x$ ) dan data sintetis ( $G(z)$ ).

Terdapat beberapa variasi GAN yang telah dikembangkan, beberapa diantaranya adalah cGAN, DCGAN, dan WGAN.



# *Generative Adversarial Network (GAN)*

$$\min_G \max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log (1 - D(G(z)))]$$

$x$  = data asli

$z$  = vektor *noise*

$D(x)$  = probabilitas  $x$  merupakan data asli

$G(z)$  = data sintesis oleh generator

$p_{data}(x)$  = distribusi probabilitas data asli

$p_z(z)$  = distribusi *noise*

- **Generator**

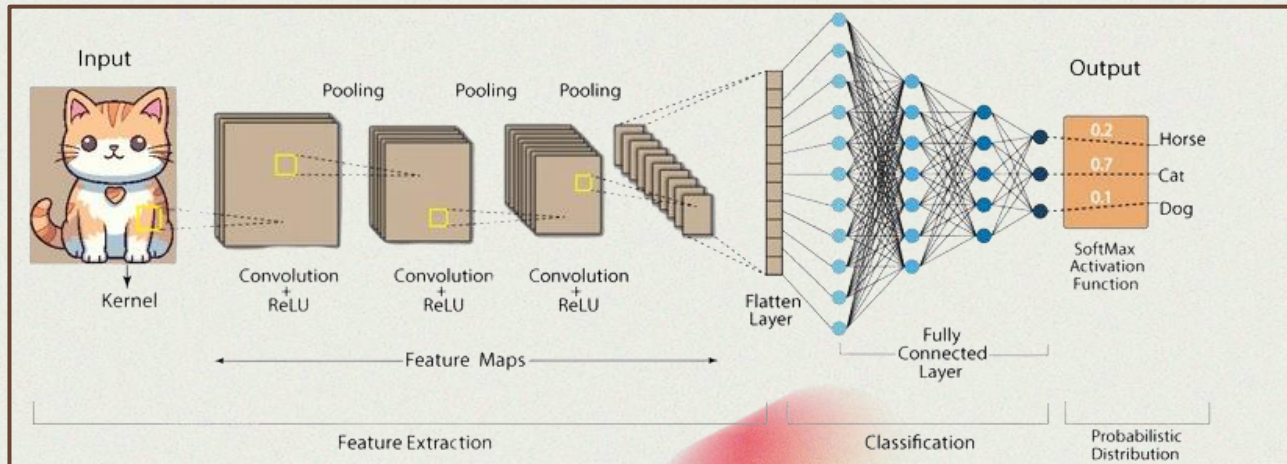
$$\min_G V(G) = \mathbb{E}_{z \sim p_z(z)} [\log (1 - D(G(z)))]$$

- **Diskriminator**

$$\max_D V(D, G) = \mathbb{E}_{x \sim p_{data}(x)} [\log D(x)] + \mathbb{E}_{z \sim p_z(z)} [\log (1 - D(G(z)))]$$

# *Convolutional Neural Network (CNN)*

- CNN merupakan *neural network* yang memiliki kemampuan untuk mendeteksi dan mengenali objek pada citra, teks, suara, dan jenis data lain.
- CNN terdiri dari beberapa *layer*, yaitu *convolutional layer*, *pooling layer*, fungsi aktivasi, dan *fully-connected layer* (Zhang et al., 2021).



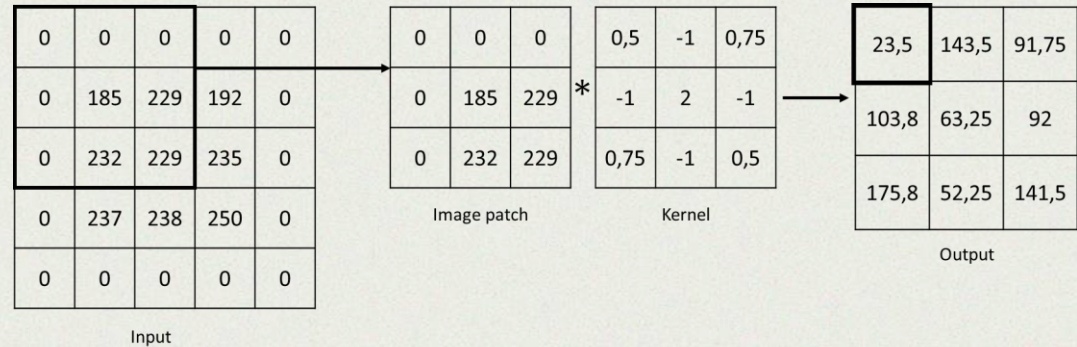


# Convolutional Layer

Lapisan yang berfungsi untuk **mengeksktraksi fitur data**. Proses ekstraksi fitur dilakukan melalui operasi konvolusi berikut

$$S(i, j) = \sum_m \sum_n I(i + m, j + n) * K(m, n)$$

- $I$  : *input*
- $K$  : kernel atau filter
- $S(i, j)$  : *feature map*



Operasi Konvolusi dengan Input 5x5, Kernel 3x3, dan Stride 1

Stride, besarnya pergeseran jendela kernel pada *input* saat melakukan operasi konvolusi

# Fungsi Aktivasi

Menentukan apakah neuron akan diaktifkan atau tidak berdasarkan nilai *inputnya*.  
(Goodfellow, Bengio, & Courville, 2016).

## Sigmoid

$$f(x) = \sigma(x) = \frac{1}{1 + e^{-x}}$$

## Swish

$$f(x) = x(\sigma(x))$$

## RELU

$$f(x) = \begin{cases} 0, & \text{for } x < 0 \\ x, & \text{for } x \geq 0 \end{cases}$$

## Leaky RELU

$$f(x) = \begin{cases} \alpha x, & \text{for } x < 0 \\ x, & \text{for } x \geq 0 \end{cases}$$

## Tanh

$$f(x) = \frac{2}{1 + e^{-2x}} - 1$$



# *Global Average Pooling (GAP)*

Fungsi *pooling* yang menghitung nilai rata-rata dengan ukuran kernel sama dengan ukuran *feature maps* (Lin, Chen, & Yan, 2013). Misal sebuah *feature maps* berukuran  $H \times W$  akan direpresentasikan menjadi nilai berupa rata-rata yang mewakili seluruh nilai dari *feature maps* dengan  $U_c(i, j)$  adalah nilai unit pada *feature maps*:

$$GlobalAveragePooling = \frac{1}{H \times W} \sum_{m=1}^H \sum_{n=1}^W U_c(i, j)$$

# ***Batch Normalization***

- Membantu meningkatkan stabilitas dan kecepatan pelatihan model. Tujuan utamanya untuk mengurangi masalah *internal covariate shift*, yang terjadi akibat perubahan distribusi input pada setiap layer model.
- Dihitung nilai mean ( $\mu_B$ ) dan variansi ( $\sigma_B^2$ ) dari aktivasi atau *output* neuron. Kemudian, dilakukan normalisasi sedemikian sehingga rata-rata nilainya mendekati 0 dan nilai standar mendekati 1.

$$\mu_B = \frac{1}{m} \sum_{i=1}^m x_i$$

$$\sigma_B^2 = \frac{1}{m} \sum_{i=1}^m (x_i - \mu_B)^2$$

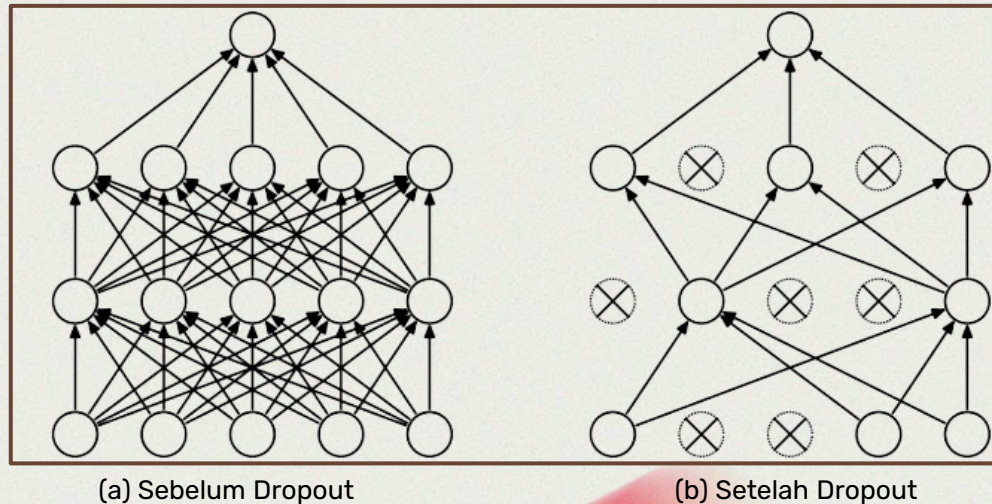
$$\hat{x}_i = \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}}$$

dengan  $\epsilon$  bernilai 0,001



# Dropout

Mencegah **overfitting** pada training model CNN. Pada setiap iterasi, *dropout* secara acak menonaktifkan neuron di suatu *layer* sebelum masuk ke *layer* selanjutnya (Srivastava et al, 2014).



Sumber : Srivastava et al, 2014

# *Binary Cross Entropy*

*Binary Cross Entropy* merupakan *loss function* yang digunakan pada masalah klasifikasi biner untuk menghitung selisih antara hasil prediksi model dengan label asli dari dataset.

$$L = -[y_1 \log(p(y_1)) + (1 - y_1) \log(p(y_2))]$$

- $y_1$ : label asli kelas 0
- $y_2$ : label asli kelas 1
- $p(y_1)$ : probabilitas prediksi kelas 0
- $p(y_2)$ : probabilitas prediksi kelas 1



# Fungsi Optimasi Adam

*Adaptive movement estimation* (Adam) adalah fungsi optimasi berbasis *stochastic gradient descent* (SGD) yang bertujuan memperbarui parameter agar dapat meminimumkan *loss function* dengan menghitung *adaptive learning rate* pada gradien sebelumnya

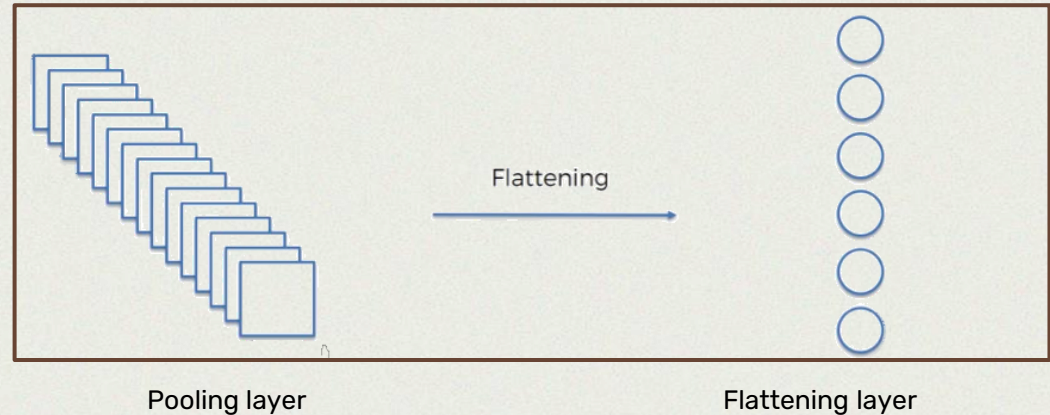
$$m_t = \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$$

$$v_t = \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$$

dengan  $\beta_1 = 0,9$  dan  $\beta_2 = 0,999$ ,  $g_t$ : gradien *loss function* pada iterasi ke- $t$ .

# *Flattening Layer*

*Output dari pooling layer harus diubah menjadi vektor berukuran satu dimensi dengan menggunakan flattening layer agar bisa diproses oleh fully connected layer (Zhang et al., 2021).*



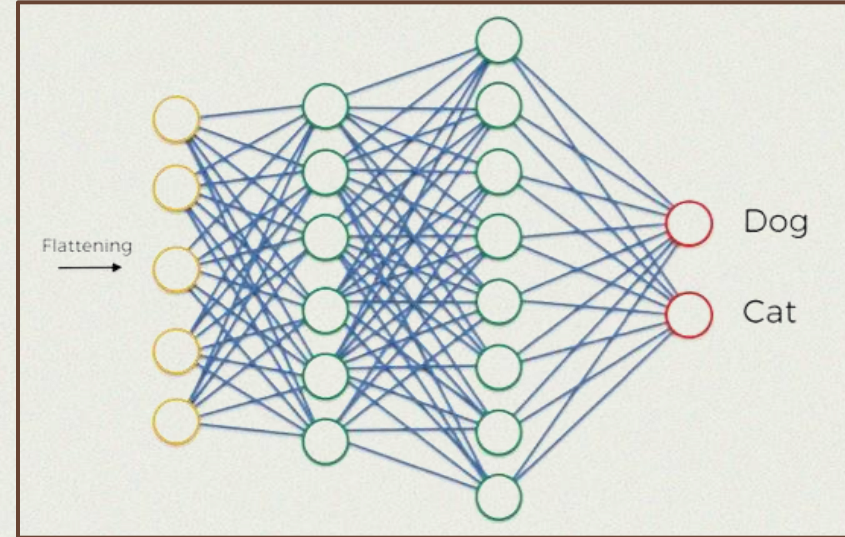
Sumber: Gomaa et al., 2023



# Fully-connected Layer

Lapisan di mana setiap neuron pada suatu *layer* terhubung ke semua neuron dari *layer* sebelumnya (Goodfellow, 2016). Didapatkan probabilitas yang merepresentasikan hasil klasifikasi sebagai *output* akhir model.

$$z_i = \sum_{j=1}^n w_{ij}x_j + b_i$$



Sumber: superdatascience.com

# Evaluasi Kinerja Model

Aktual	Prediksi	
	Normal	AMD
Normal	TN	FP
AMD	FN	TP

- $Akurasi = \frac{TN+TP}{TN+FP+TP+FN}$
- $Sensitivity = \frac{TP}{TP+FN}$
- $Specifity = \frac{TN}{TN+FP}$

dengan:

- TN (*True Negative*): citra normal yang diklasifikasi benar sebagai citra normal
- FP (*False Positive*): citra normal yang diklasifikasi salah sebagai citra AMD
- FN (*False Negative*): citra AMD yang diklasifikasi salah sebagai citra normal
- TP (*True Positive*): citra AMD yang diklasifikasi benar sebagai citra AMD



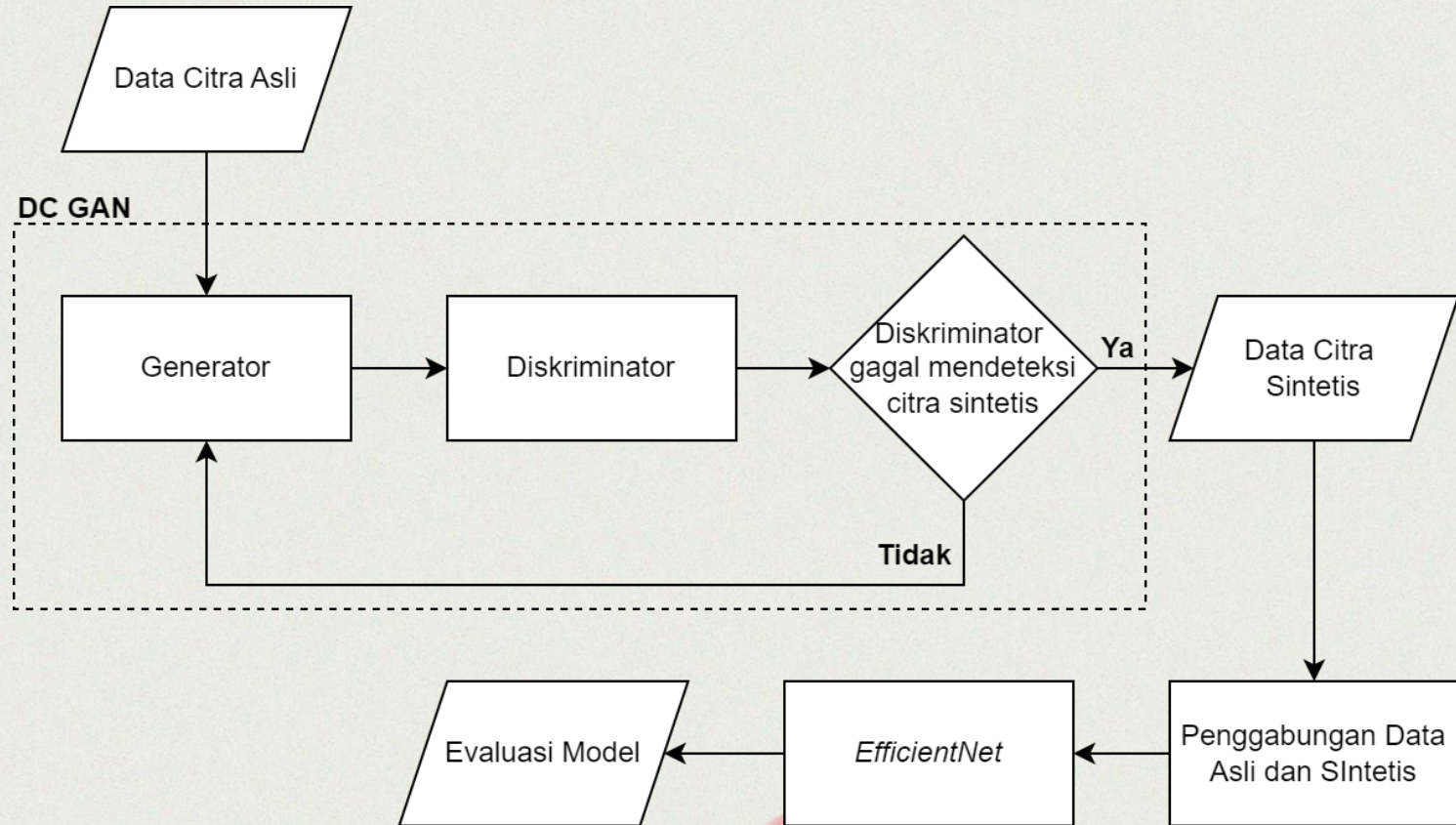
# 03

## Penerapan GAN pada EfficientNet



Deep Convolutional GAN  
EfficientNet  
Integrasi DCGAN pada EfficientNet

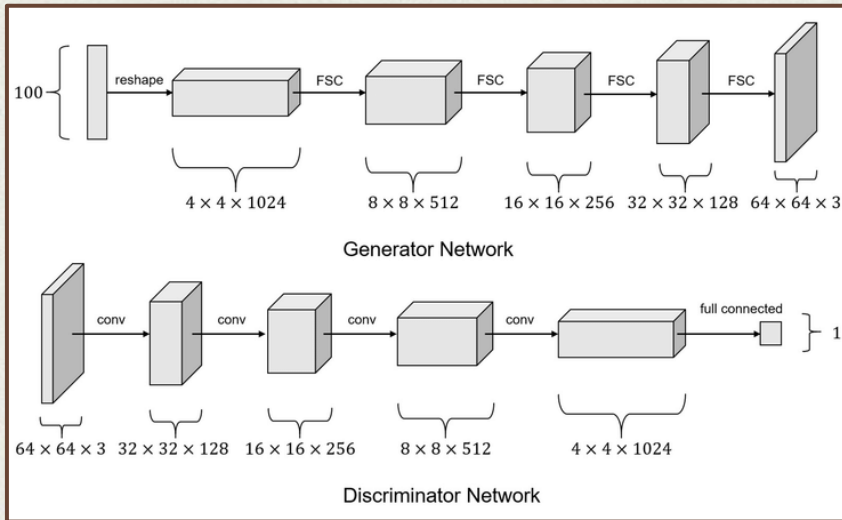
# Integrasi DCGAN pada *EfficientNet*





# 3.1 Deep Convolutional GAN

DCGAN merupakan model GAN yang menggunakan *convolutional layer* pada arsitekturnya (Radford, Metz, & Chintala, 2014)



## Modifikasi pada DCGAN

1. Pada generator, *pooling layer* diganti dengan *fractional-strided convolution*. Sedangkan, pada diskriminator diganti dengan *strided convolution*.
2. Menggunakan *Batch Normalization* pada generator dan diskriminator.
3. Menghilangkan *fully connected layer* pada arsitektur yang lebih dalam.
4. Menggunakan fungsi ReLU pada generator, kecuali pada output layer menggunakan fungsi tanh.
5. Menggunakan fungsi LeakyReLU pada diskriminator

# DCGAN (Generator)

Misalkan terdapat vektor *noise*  $Z$  berdimensi 9. Pada *layer* pertama generator, terdapat filter  $W_1$  dan bobot bias  $b_1$

$$Z = \begin{bmatrix} 0,5 \\ -0,3 \\ 0,2 \\ -1,1 \\ 0,7 \\ -0,6 \\ 0,9 \\ -0,8 \\ 0,4 \end{bmatrix}, \quad W_1 = \begin{bmatrix} 0,1 & 0,2 & 0,3 & 0,4 & 0,1 & 0,2 & 0,3 & 0,4 & 0,1 \\ 0,2 & 0,3 & 0,4 & 0,1 & 0,2 & 0,3 & 0,4 & 0,1 & 0,2 \\ 0,3 & 0,4 & 0,1 & 0,2 & 0,3 & 0,4 & 0,1 & 0,2 & 0,3 \\ 0,4 & 0,1 & 0,2 & 0,3 & 0,4 & 0,1 & 0,2 & 0,3 & 0,4 \end{bmatrix}, \quad b_1 = \begin{bmatrix} 0,1 \\ 0,2 \\ 0,3 \\ 0,4 \end{bmatrix}$$

*Layer* pertama dihitung dengan fungsi aktivasi ReLU :  $h_1 = \text{ReLU}(W_1 \cdot Z + b_1)$

$$h_1 = \text{ReLU} \left( \begin{bmatrix} 0,1 & 0,2 & 0,3 & 0,4 & 0,1 & 0,2 & 0,3 & 0,4 & 0,1 \\ 0,2 & 0,3 & 0,4 & 0,1 & 0,2 & 0,3 & 0,4 & 0,1 & 0,2 \\ 0,3 & 0,4 & 0,1 & 0,2 & 0,3 & 0,4 & 0,1 & 0,2 & 0,3 \\ 0,4 & 0,1 & 0,2 & 0,3 & 0,4 & 0,1 & 0,2 & 0,3 & 0,4 \end{bmatrix} \cdot \begin{bmatrix} 0,5 \\ -0,3 \\ 0,2 \\ -1,1 \\ 0,7 \\ -0,6 \\ 0,9 \\ -0,8 \\ 0,4 \end{bmatrix} + \begin{bmatrix} 0,1 \\ 0,2 \\ 0,3 \\ 0,4 \end{bmatrix} \right) = \text{ReLU} \left( \begin{bmatrix} -0,35 \\ 0,50 \\ 0,15 \\ 0,60 \end{bmatrix} \right) = \begin{bmatrix} 0 \\ 0,50 \\ 0,15 \\ 0,60 \end{bmatrix}$$



# DCGAN (Generator)

Pada *layer* kedua generator, terdapat filter  $W_2$  dan bobot bias  $b_2$

$$W_2 = \begin{bmatrix} 0,1 & 0,2 & 0,3 & 0,4 \\ 0,5 & 0,6 & 0,7 & 0,8 \\ 0,9 & 0,1 & 0,2 & 0,3 \\ 0,4 & 0,5 & 0,6 & 0,7 \\ 0,8 & 0,9 & 0,1 & 0,2 \\ 0,3 & 0,4 & 0,5 & 0,6 \\ 0,7 & 0,8 & 0,9 & 0,1 \\ 0,2 & 0,3 & 0,4 & 0,5 \\ 0,6 & 0,7 & 0,8 & 0,9 \end{bmatrix}, b_2 = \begin{bmatrix} 0,1 \\ 0,2 \\ 0,3 \\ 0,4 \\ 0,5 \\ 0,6 \\ 0,7 \\ 0,8 \\ 0,9 \end{bmatrix}$$

*Layer* kedua dihitung dengan fungsi aktivasi tanh:  $G(Z) = \tanh(W_2 \cdot h_1 + b_2)$

$$G(Z) = \tanh \left( \begin{bmatrix} 0,1 & 0,2 & 0,3 & 0,4 \\ 0,5 & 0,6 & 0,7 & 0,8 \\ 0,9 & 0,1 & 0,2 & 0,3 \\ 0,4 & 0,5 & 0,6 & 0,7 \\ 0,8 & 0,9 & 0,1 & 0,2 \\ 0,3 & 0,4 & 0,5 & 0,6 \\ 0,7 & 0,8 & 0,9 & 0,1 \\ 0,2 & 0,3 & 0,4 & 0,5 \\ 0,6 & 0,7 & 0,8 & 0,9 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0,50 \\ 0,15 \\ 0,60 \end{bmatrix} + \begin{bmatrix} 0,1 \\ 0,2 \\ 0,3 \\ 0,4 \\ 0,5 \\ 0,6 \\ 0,7 \\ 0,8 \\ 0,9 \end{bmatrix} \right) = \tanh \left( \begin{bmatrix} 0,49 \\ 1,09 \\ 0,56 \\ 1,16 \\ 1,09 \\ 1,24 \\ 1,30 \\ 1,31 \\ 1,91 \end{bmatrix} \right) = \begin{bmatrix} 0,45 \\ 0,80 \\ 0,51 \\ 0,82 \\ 0,80 \\ 0,84 \\ 0,86 \\ 0,86 \\ 0,96 \end{bmatrix}$$

# DCGAN (Diskriminator)

Dari generator, didapatkan *ouput*  $G(Z)$  sebagai *input*  $x$ . Pada *layer* pertama diskriminator terdapat filter  $W_3$  dan bobot bias  $b_3$

$$W_3 = \begin{bmatrix} 0,1 & 0,3 & 0,2 & 0,3 & 0,1 & 0,3 & 0,2 & 0,3 & 0,1 \\ 0,2 & 0,1 & 0,3 & 0,1 & 0,2 & 0,1 & 0,3 & 0,1 & 0,2 \\ 0,3 & 0,2 & 0,1 & 0,2 & 0,3 & 0,2 & 0,1 & 0,2 & 0,3 \end{bmatrix}, b_3 = \begin{bmatrix} 0,1 \\ 0,2 \\ 0,3 \end{bmatrix}$$

*Layer* pertama dihitung dengan fungsi aktivasi LeakyReLU :  $h_3 = \text{LeakyReLU}(W_3 \cdot x + b_3)$

$$h_3 = \text{LeakyReLU} \left( \begin{bmatrix} 0,1 & 0,3 & 0,2 & 0,3 & 0,1 & 0,3 & 0,2 & 0,3 & 0,1 \\ 0,2 & 0,1 & 0,3 & 0,1 & 0,2 & 0,1 & 0,3 & 0,1 & 0,2 \\ 0,3 & 0,2 & 0,1 & 0,2 & 0,3 & 0,2 & 0,1 & 0,2 & 0,3 \end{bmatrix} \cdot \begin{bmatrix} 0,45 \\ 0,80 \\ 0,51 \\ 0,82 \\ 0,80 \\ 0,84 \\ 0,86 \\ 0,86 \\ 0,96 \end{bmatrix} + \begin{bmatrix} 0,1 \\ 0,2 \\ 0,3 \end{bmatrix} \right) = \begin{bmatrix} 1,59 \\ 1,38 \\ 1,76 \end{bmatrix}$$



# DCGAN (Diskriminator)

Pada *layer* kedua , anggap terdapat filter  $W_4$  dan bobot bias  $b_4$  sebagai berikut.

$$W_4 = \begin{bmatrix} 0,1 \\ 0,2 \\ 0,3 \end{bmatrix}, b_4 = [0,1]$$

Hitung *layer* kedua diskriminator dengan menggunakan fungsi sigmoid.

$$D(G(\mathbf{Z})) = \sigma(W_4 \cdot h_3 + b_4)$$

$$D(G(\mathbf{Z})) = \sigma \left( \begin{bmatrix} 0,1 & 0,2 & 0,3 \end{bmatrix} \cdot \begin{bmatrix} 1,59 \\ 1,38 \\ 1,76 \end{bmatrix} + [0,1] \right) = \sigma([0,96] + [0,1]) = \sigma([1,06])$$

$$D(G(\mathbf{Z})) \approx 0,7432$$

# DCGAN (Output)

Dari generator didapat  $G(\mathbf{Z})$  yang merepresentasikan citra sintetis

$$x = G(\mathbf{Z}) = \begin{bmatrix} 0,45 \\ 0,80 \\ 0,51 \\ 0,82 \\ 0,80 \\ 0,84 \\ 0,86 \\ 0,86 \\ 0,96 \end{bmatrix} \xrightarrow{\text{Transformasi linear}} x' = \begin{bmatrix} 185 \\ 229 \\ 192 \\ 232 \\ 229 \\ 235 \\ 237 \\ 238 \\ 250 \end{bmatrix}$$

$G(\mathbf{Z})$  memiliki range  $[-1,1]$  sesuai dengan range fungsi  $\tanh$  . Sedangkan, *input* model *EfficientNet* berupa matriks yang merepresentasikan piksel citra dengan range  $[0,255]$  untuk citra *grayscale*. Sehingga, perlu dilakukan transformasi linear.

$$x' = \frac{x + 1}{2} \times 255$$



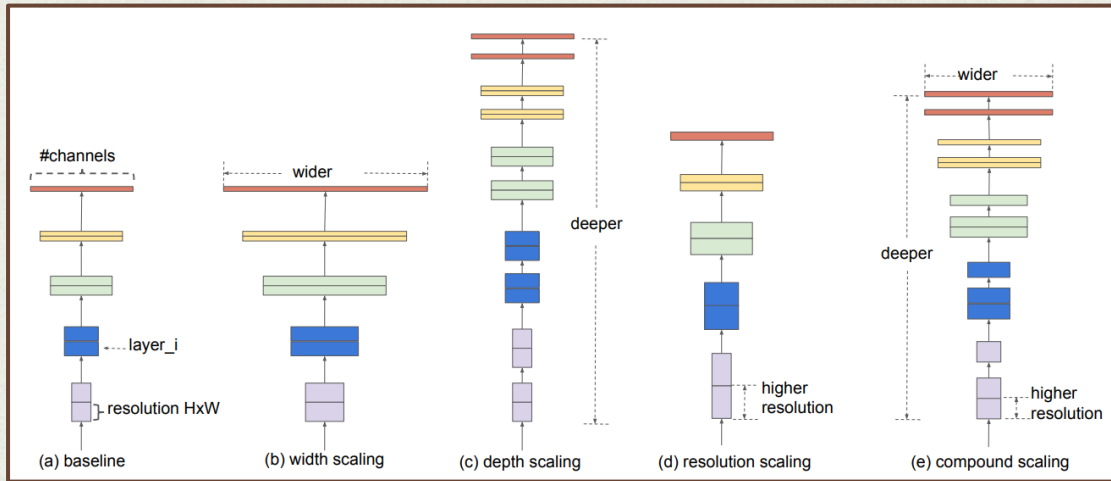
# EfficientNet

- EfficientNet menyelesaikan permasalahan peningkatan skala model dengan tetap meningkatkan akurasi pada model yang dibuat.
- EfficientNet memiliki tiga mekanisme utama, yaitu **compound scaling**, **mobile inverted residual block** (MBConv), dan **squeeze-and-excitation** (SE) block.

EfficientNet	Accuracy	Sensitivity	Specifity	Running time (s)
B0	0,8197	0,7333	0,9032	105,88
B1	0,8197	0,8000	0,8387	163,64
B2	0,8033	0,7667	0,8377	199,18
B3	0,8525	0,8333	0,8709	348,64
B4	0,7541	0,7000	0,8065	726,42
B5	0,8525	0,8000	0,90323	1548,02
B6	0,8197	0,7333	0,90323	2781,44
B7	0,7868	0,8000	0,77419	5022,15

# Compound Scaling

*Compound scaling* meningkatkan seluruh dimensi yaitu kedalaman, lebar, dan resolusi secara seragam dengan koefisien *scaling* tetap.



kedalaman:  $d = \alpha^\phi$

luas:  $w = \beta^\phi$

resolusi:  $r = \gamma^\phi$

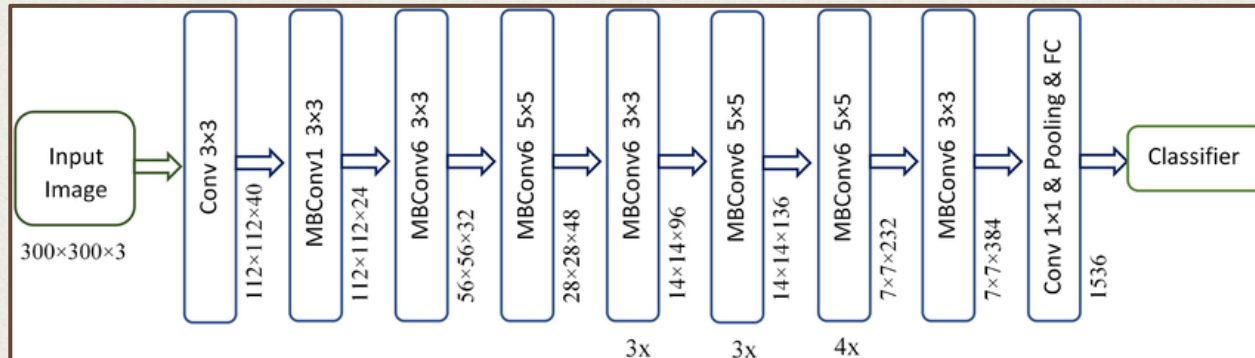
d.k.  $\alpha * \beta^2 * \gamma^2 \approx 2$

$\alpha \geq 1, \beta \geq 1, \gamma \geq 1$

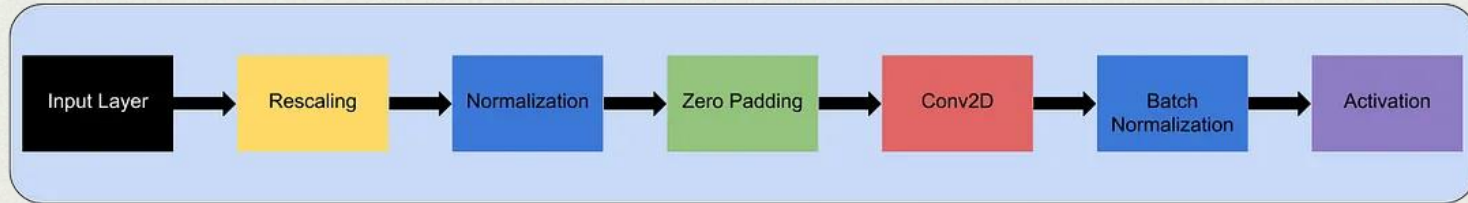


# MBConv

- MBConv atau *Mobile Inverted Bottleneck Convolution* merupakan blok konvolusi yang menjadi dasar pembentukan model EfficientNet.
- Prinsip utama MBConv adalah menggabungkan kemampuan untuk menangkap fitur citra melalui beberapa arsitektur.
- Terdapat 2 jenis MBConv yang digunakan pada arsitektur EfficientNet, yaitu MBConv 1 dan MBConv6. Angka 1 dan 6 merujuk pada ukuran faktor ekspansi.



# Modul pada MBConv



Stem



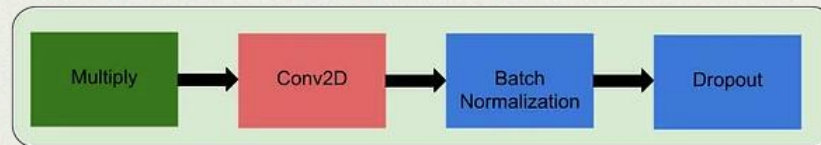
Module 1



Module 4



Module 2



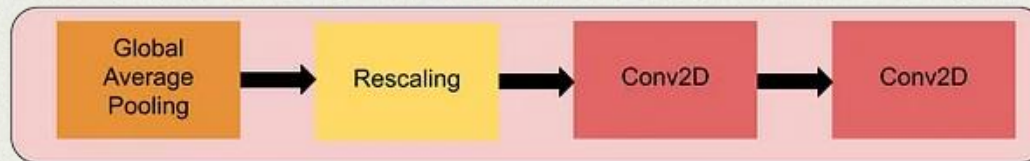
Module 5



# Block SE

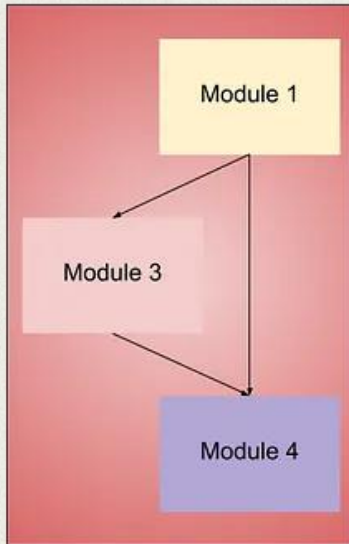
*Squeeze-and-Excitation* (SE) block adalah *attention mechanism* yang bertujuan untuk meningkatkan representasi fitur penting dengan memperkuat *channel* fitur yang relevan dan mengurangi pengaruh dari *channel* yang kurang relevan. SE Block terdiri dari dua tahap utama, yaitu *Squeeze* dan *Excitation*.

- **Squeeze** mengoversi informasi dari setiap fitur menggunakan GAP, sehingga model belajar memahami fitur berdasarkan informasi global dari keseluruhan gambar.
- **Excitation** menggunakan fungsi Sigmoid untuk menghasilkan *attention weight* yang menunjukkan seberapa penting fitur dalam data. Skala ini yang akan menentukan apakah fitur akan ditekankan atau dikurangi pengaruhnya.

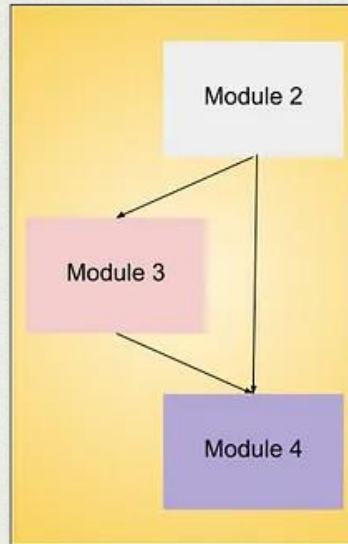


Modul 3: Block SE (towardsdatascience.com, dengan modifikasi)

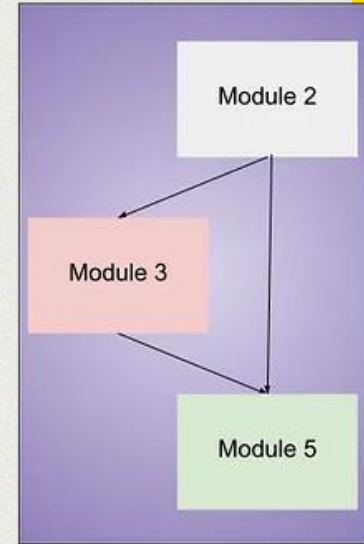
# Subblock



Sub-block 1



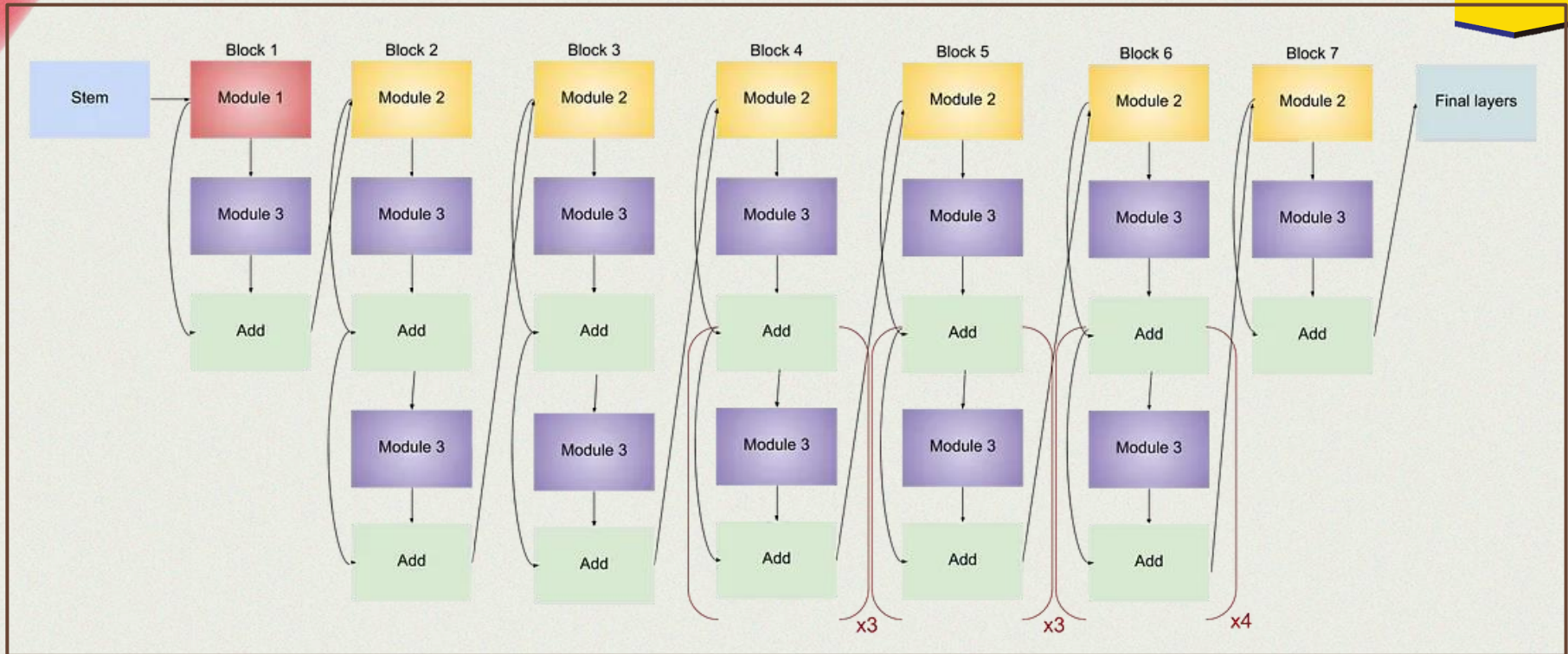
Sub-block 2



Sub-block 3



# EfficientNet (Arsitektur)



Sumber : [towardsdatascience.com](https://towardsdatascience.com/efficientnet-architecture), dengan modifikasi

# 04

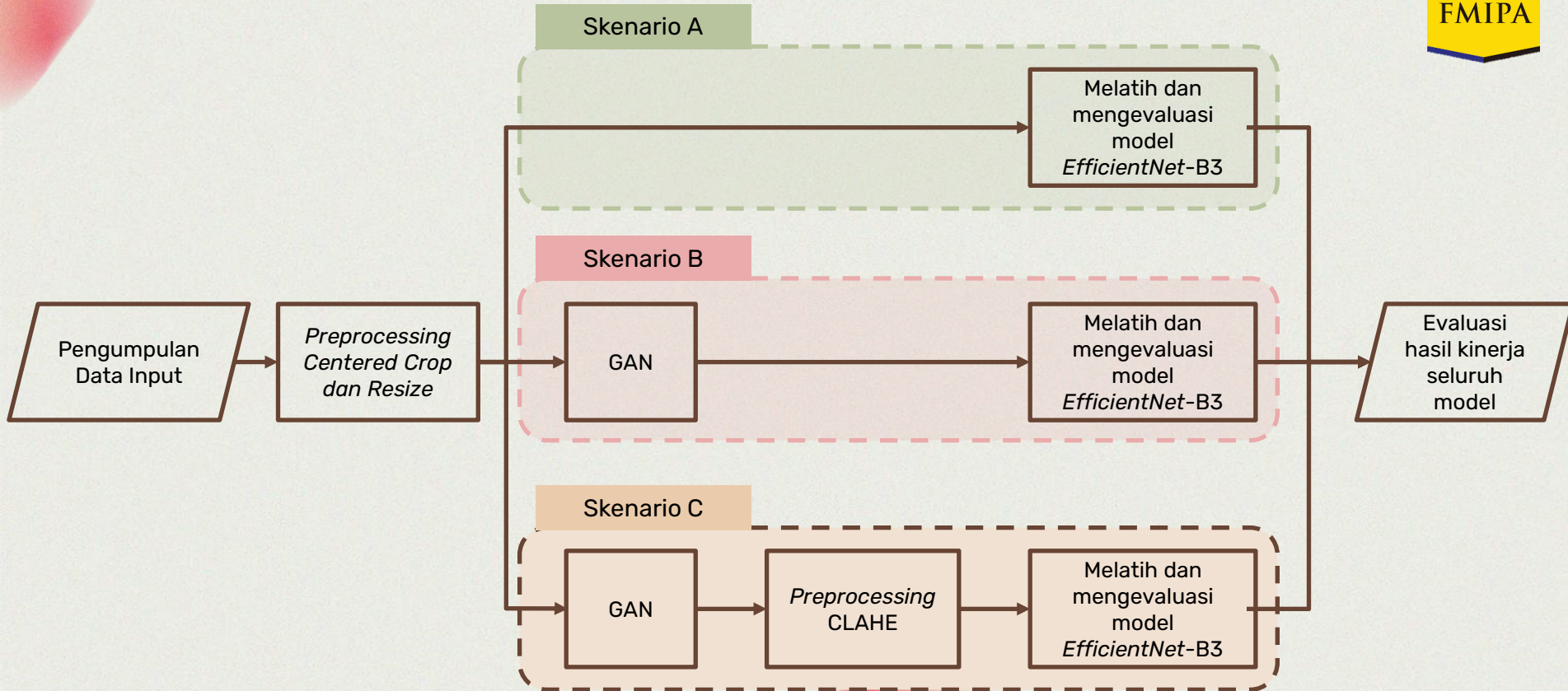
## Analisis dan Simulasi Hasil



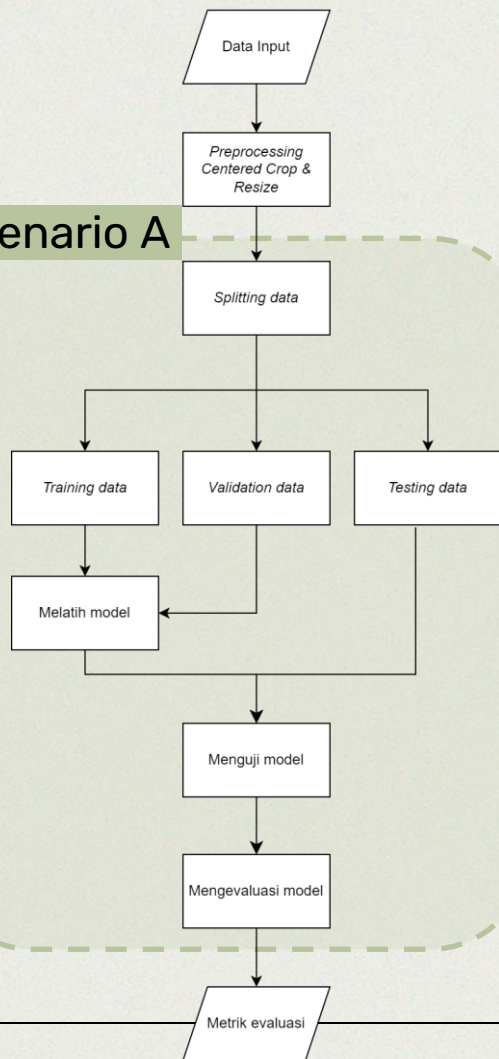
Alur Penelitian  
Pengumpulan Data  
Augmentasi Data  
Preprocessing Data  
Analisis Hasil Simulasi



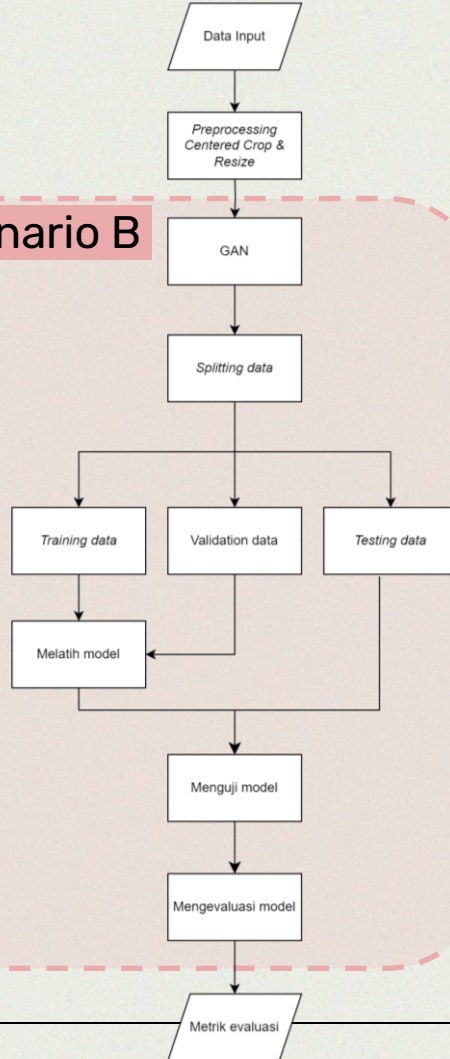
# Alur Penelitian



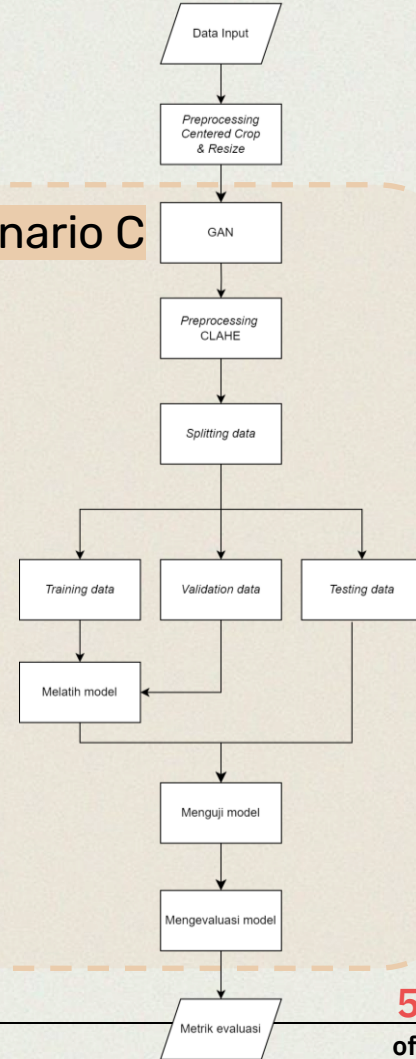
## Skenario A



## Skenario B

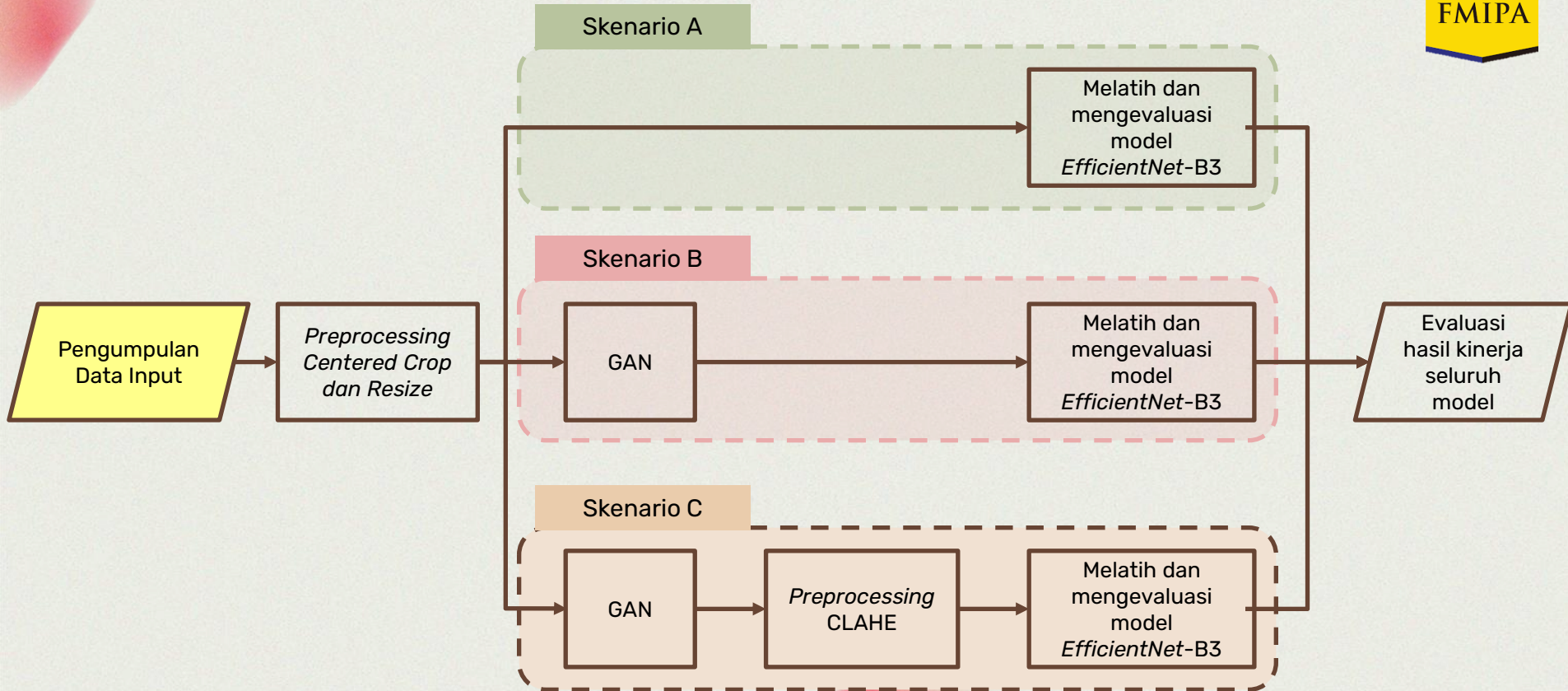


## Skenario C





# Alur Penelitian



# Pengumpulan Data

<b>Dataset</b>	<b>Kategori</b>		<b>Sumber</b>
	<b>Normal</b>	<b>AMD</b>	
<i>Joint Shantou International Eye Centre (JSIEC)</i>	38	74	<i>(Cen et al., 2021)</i>
<i>Structured Analysis of the Retina (STARE)</i>	-	44	<i>(Hoover et al., 2003)</i>
<i>Retinal Fundus Multidisease Image Dataset 2.0 (RFMID 2.0)</i>	-	5	<i>(Panchal et al., 2023)</i>
<i>Ocular Disease Recognition (ODIR)</i>	286	158	<i>(“ODIR Grand Challenge”, 2019)</i>
<b>Total</b>	<b>324</b>	<b>281</b>	



## JSIEC



Normal



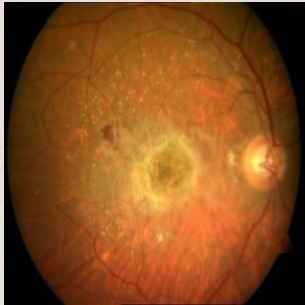
AMD

## STARE



AMD

## RFMID 2.0

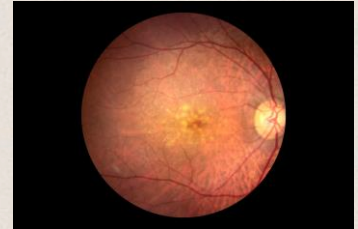


AMD

## ODIR

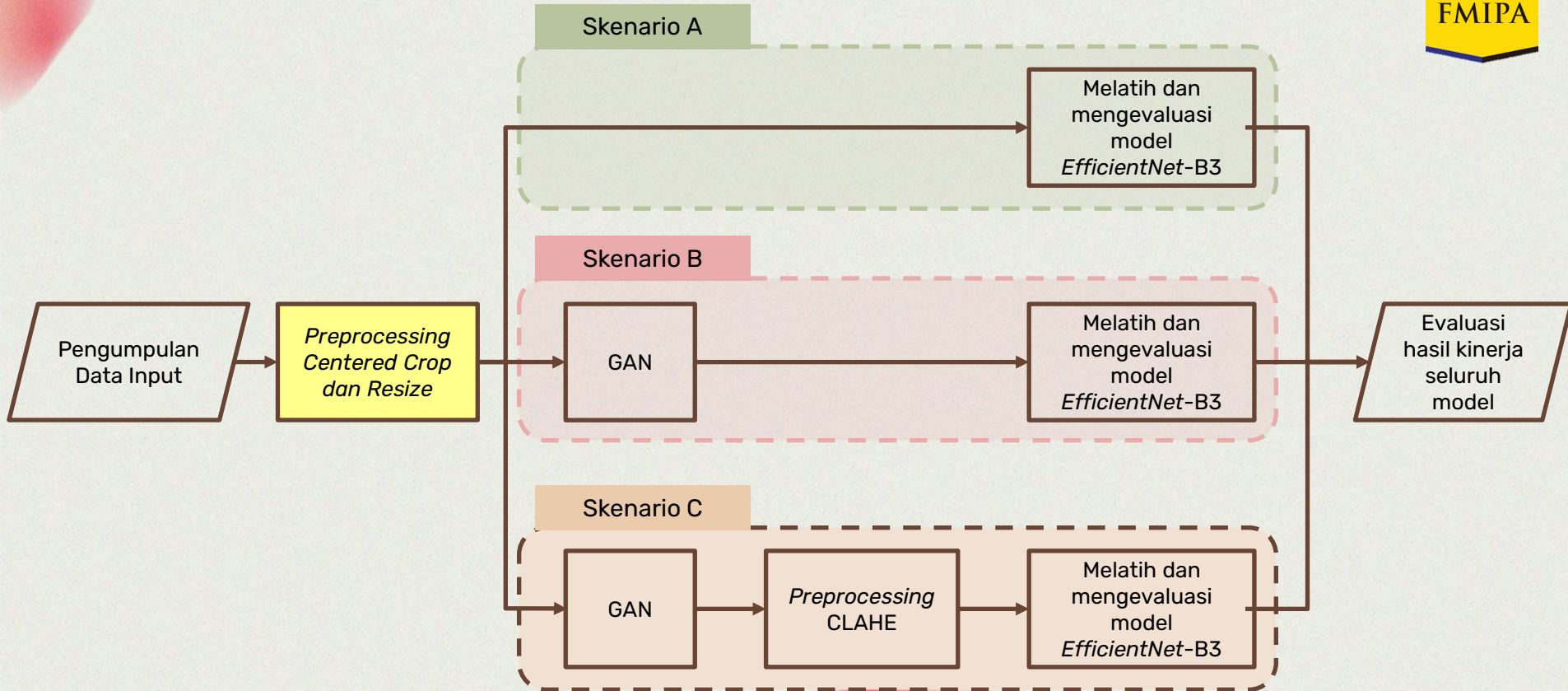


Normal



AMD

# Alur Penelitian





# *Preprocessing: Centered Crop dan Resize*

Normal



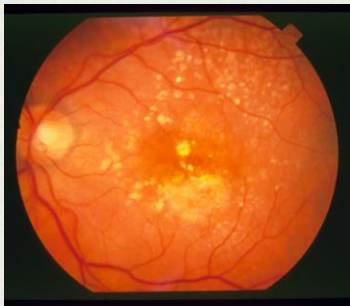
*Centered Crop*



*Resize*



AMD



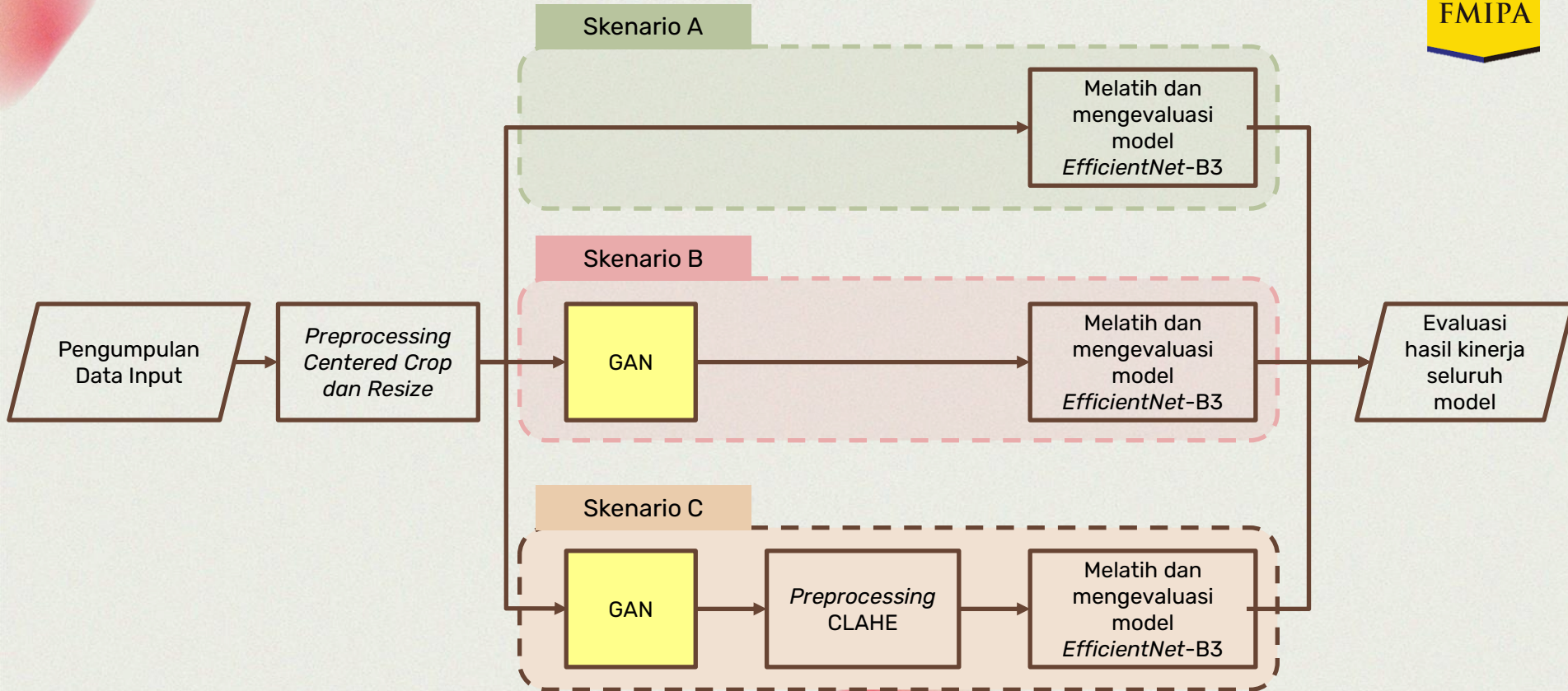
*Centered Crop*



*Resize*



# Alur Penelitian





# Augmentasi Data dengan DCGAN

Simulasi menggunakan fungsi optimasi Adam dengan :

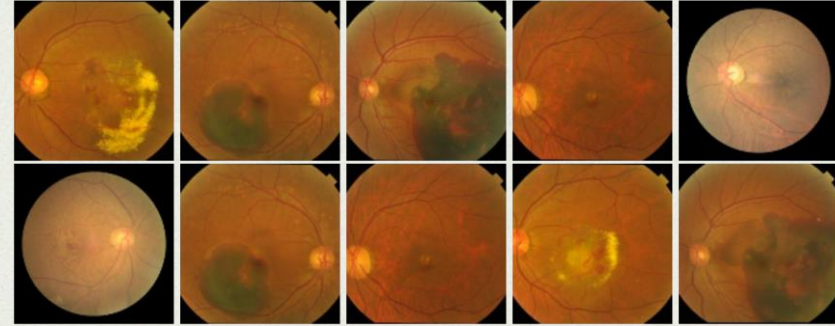
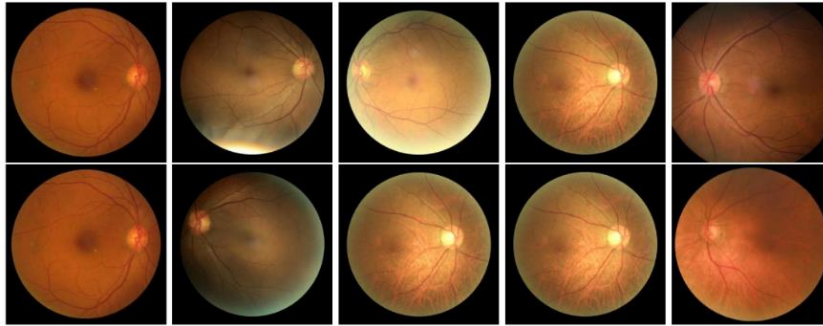
- *learning rate* = 0,0002
- *steps per epoch* = 2200
- *epoch* = 10
- *noise dimension* = 100
- *batch* = 4

Model DCGAN yang telah dibentuk dilatih dengan *dataset* Normal dan AMD secara terpisah. Data sintesis yang dihasilkan oleh generator DCGAN sudah disesuaikan dengan input model *EfficientNet-B3* yaitu berukuran 296×296 piksel. Dari proses training dihasilkan masing-masing 100 data sintesis. Kemudian, data sintesis dipilih secara manual untuk mendapatkan citra fundus yang realistis. Evaluasi kinerja model dilakukan secara visual dengan memilah data sintesis yang dihasilkan

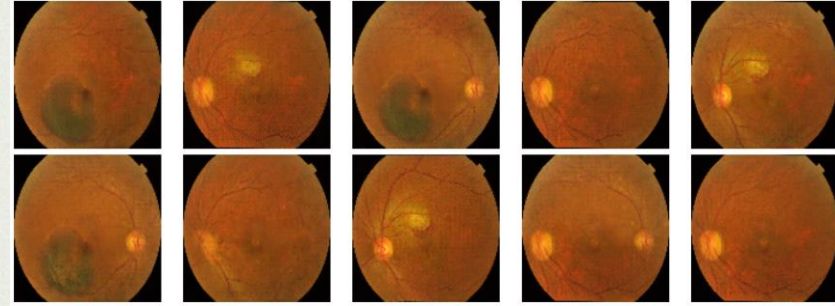
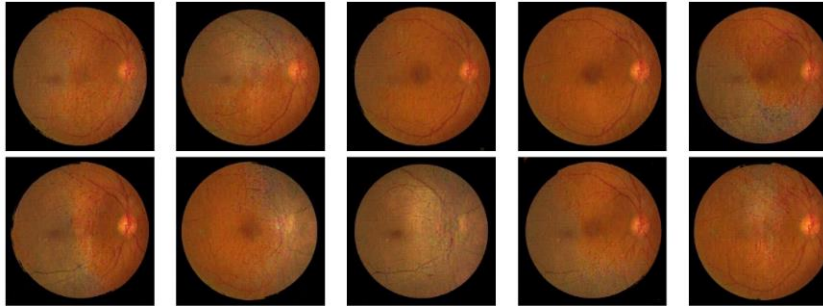
Normal

AMD

Sample  
Asli



Sample  
Sintetis





# Augmentasi Data dengan DCGAN

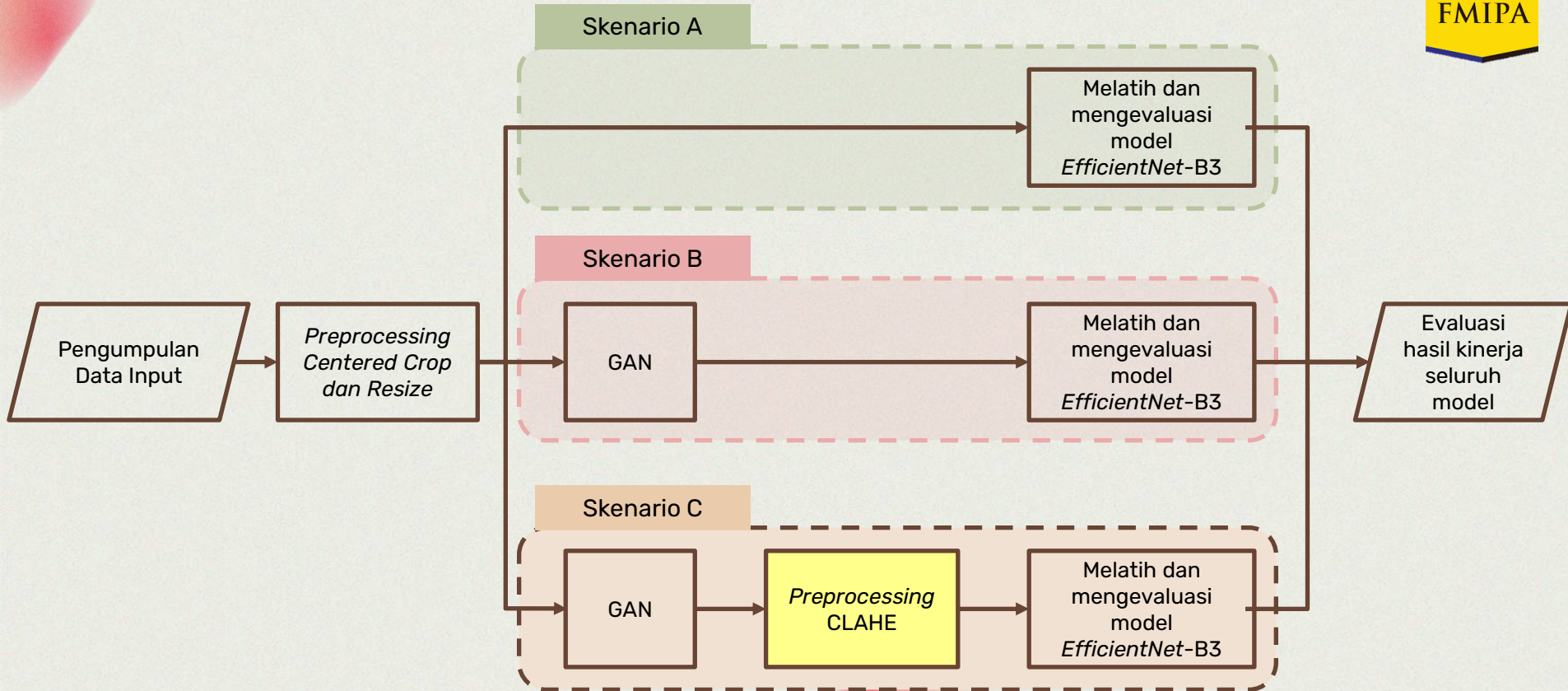
Pembagian *dataset* sebelum augmentasi dengan DCGAN:

Dataset	Training		Validation		Testing	
	Normal	AMD	Normal	AMD	Normal	AMD
Dataset 1 (80:10:10)	264	220	30	30	30	31
Dataset 2 (70:15:15)	234	189	44	47	46	45

Pembagian *dataset* setelah augmentasi dengan DCGAN:

Dataset	Training		Validation		Testing	
	Normal	AMD	Normal	AMD	Normal	AMD
Dataset 1 (80:10:10)	280	288	35	36	40	31
Dataset 2 (70:15:15)	254	242	47	60	54	53

# Alur Penelitian





# *Preprocessing: CLAHE*

CLAHE bertujuan untuk menyamakan kontras sehingga fitur pada citra dapat terlihat lebih jelas. CLAHE diterapkan dengan *clip limit* 2,0 dan *stride*  $8 \times 8$ .

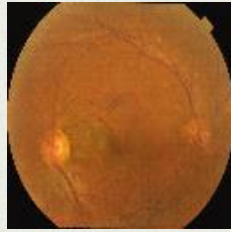
Normal (Asli)



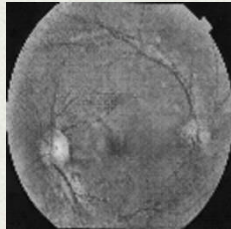
CLAHE



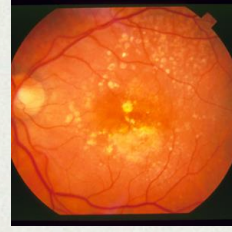
Normal (Sintetis)



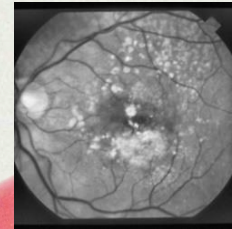
CLAHE



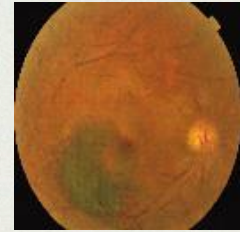
AMD (Asli)



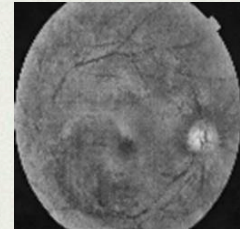
CLAHE



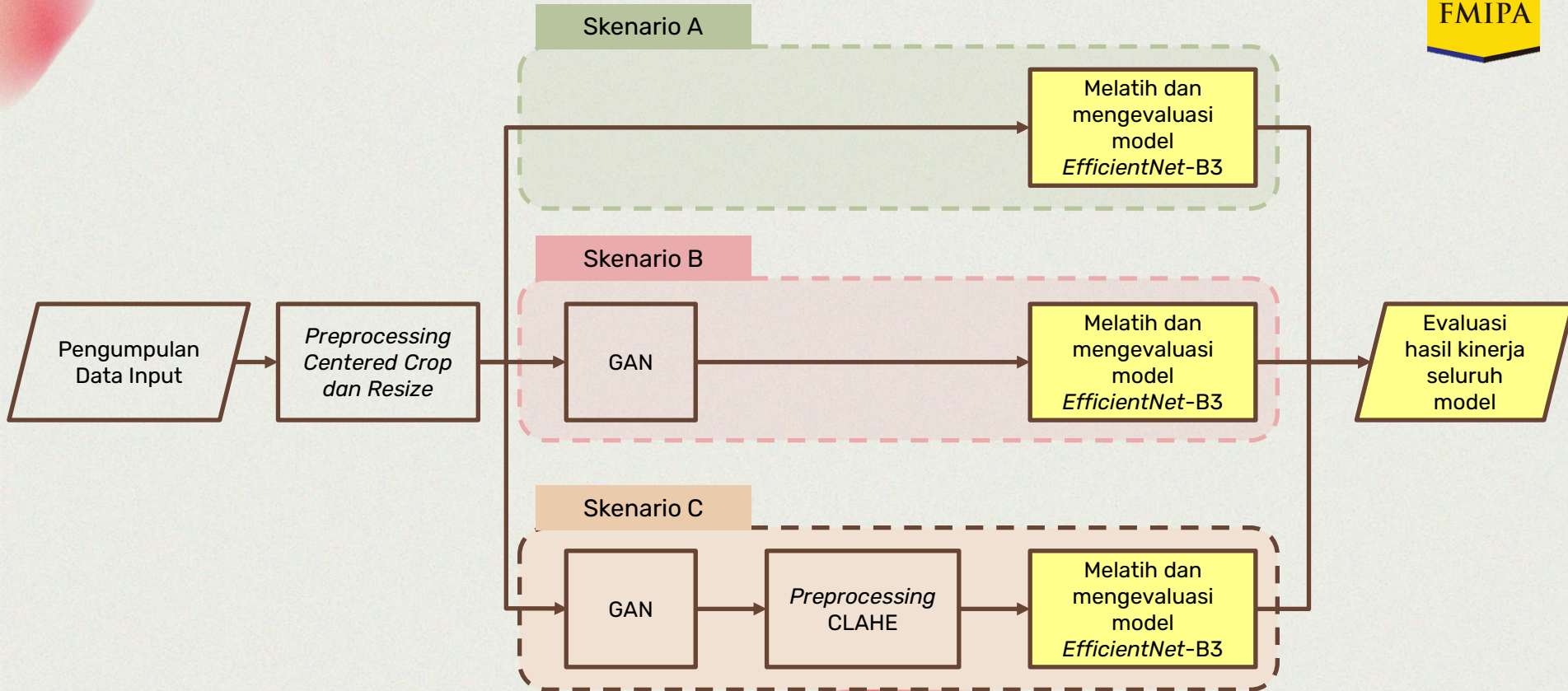
AMD (Sintetis)



CLAHE



# Alur Penelitian



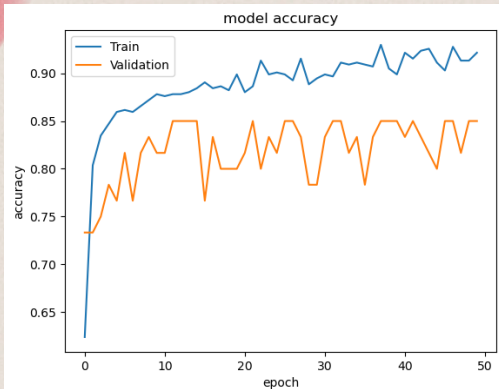


# Analisis Hasil Simulasi

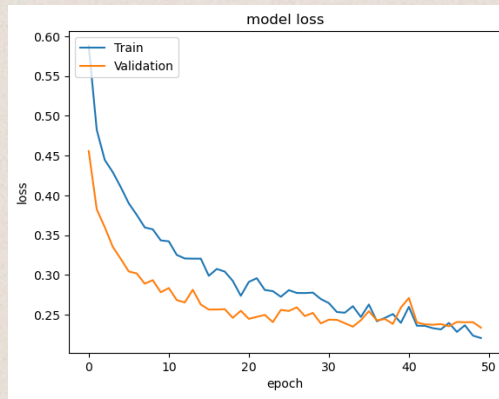
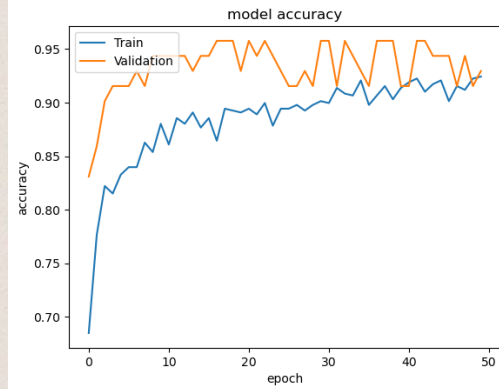
Model dilatih dengan arsitektur *EfficientNet*-B3 fungsi optimasi Adam dengan *epoch* 50 dan *learning rate* 0,001. Simulasi model diterapkan pada 3 skenario dan 2 dataset dengan 5 kali uji coba pada masing-masing kasus.

Dataset	Skenario	Accuracy	Loss	Sensitivity	Specificity
Dataset 1	Skenario A	0,8852	0,3285	0,8194	0,9533
	Skenario B	0,8986	0,3190	0,9161	0,8850
	Skenario C	0,9127	0,3119	0,8968	0,9250
Dataset 2	Skenario A	0,8901	0,3160	0,8267	0,9522
	Skenario B	0,8710	0,3066	0,7962	0,9444
	Skenario C	0,8897	0,2713	0,8868	0,8926

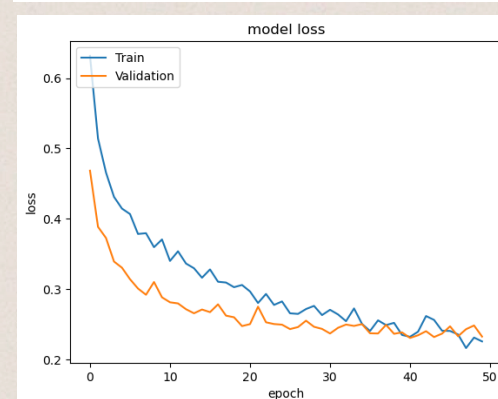
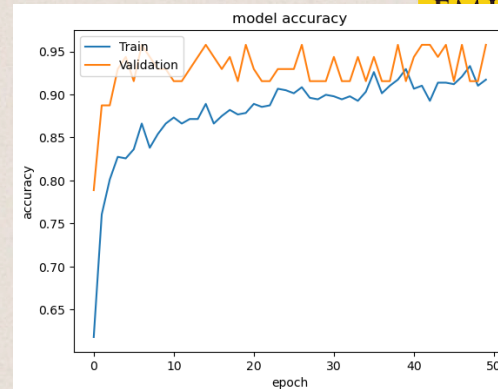
# Analisis Hasil Simulasi



Tanpa GAN



Dengan GAN



Dengan GAN & CLAHE





# 05

## Penutup

---

Kesimpulan  
Saran

# Kesimpulan

1

Model GAN mampu menghasilkan citra sintesis yang realistis dan memiliki karakteristik menyerupai citra asli. Namun, perlu dilakukan pemilihan data secara manual pada data sintesis untuk memisahkan data sintesis yang kurang realistis.

2

Model *EfficientNet-B3* dipilih setelah melakukan uji coba pada setiap variasi *EfficientNet*. Model *EfficientNet-B3* dengan rasio pembagian data 80:10:10 memberikan hasil yang lebih konsisten pada ketiga skenario dalam melakukan klasifikasi. Akurasi tertinggi mencapai 0,9296 pada skenario C. *Sensitivity* tertinggi mencapai 0,9355 pada skenario B dan C. *Specifity* tertinggi mencapai 1,000 pada skenario A.



# Kesimpulan

3

Penggunaan DCGAN sebagai metode augmentasi data membuat waktu *training EfficientNet-B3* menjadi lebih lama. Kemudian, penggunaan *preprocessing* CLAHE pada data yang sudah melalui augmentasi GAN memberikan pengaruh yang cukup signifikan dalam meningkatkan akurasi dan *sensitivity* pada rasio pembagian data 80:10:10. Nilai rata-rata akurasi mencapai 0,9127 dan nilai rata-rata *sensitivity* mencapai 0,8968. Jadi, Berdasarkan nilai rata-rata akurasi, *sensitivity*, dan *specifity*, skenario C pada rasio pembagian data 80:10:10 merupakan pendekatan yang paling optimal dalam mendeteksi AMD.

# Saran

1

Menggunakan *dataset* dengan jumlah yang lebih banyak sehingga model DCGAN dapat belajar dengan lebih efektif dan menghasilkan citra sintesis yang lebih realistis dan beragam

2

Menggunakan arsitektur *EfficientNet* lain yang lebih kompleks disesuaikan dengan *dataset* yang digunakan.

3

Menerapkan metode *preprocessing* lain sebagai upaya meningkatkan kualitas citra dan menekankan fitur tertentu yang ada pada citra

4

Menggunakan dan membandingkan parameter lain (*epoch*, *batch size*, *learning rate*) untuk mencari kombinasi parameter yang lebih efektif dalam meningkatkan performa model DCGAN dan *EfficientNet*.



# DAFTAR PUSTAKA

- Ahlawat, P., Borgman, J., Eden, S., Huels, S., landiorio, J., Kumar, A., & Zakahi, P. (2023). A New Architecture to Manage Data Costs and Complexity. Boston Consulting Group (BCG), 1-12
- Anam, D. A., Novamizanti, L., & Rizal, S. (2021). Classification of retinal pathology via OCT images using convolutional neural network. In 2021 International Conference on Computer System, Information Technology, and Electrical Engineering (COSITE) (pp. 12-17). IEEE.
- Baskar, A., Rajappa, M., Vasudevan, S. K., & Murugesh, T. S. (2023). Digital Image Processing. CRC Press.
- Cantor L.B., Rapuano C.J., Cioffi G.A. (2015). American Academy of Ophthalmology. Basic and Clinical Science Course (BCSC): Section 12: Retina and Vitreous.
- Cen, Ling-Ping. (2021). "Automatic detection of 39 fundus diseases and conditions in retinal photographs using deep neural network." Nature communications: 4828.
- Chollet, F. (2021). Deep learning with Python. Simon and Schuster.
- Deng, Y., Qiao, L., Du, M., Qu, C., Wan, L., Li, J., & Huang, L. (2022). Age-related macular degeneration: Epidemiology, genetics, pathophysiology, diagnosis, and targeted therapy. Genes & diseases, 9(1), 62-79.
- Flaxel, C. J., Adelman, R. A., Bailey, S. T., Fawzi, A., Lim, J. I., Vemulakonda, G. A., & Ying, G. S. (2020). Age-related macular degeneration preferred practice pattern®. Ophthalmology, 127(1), P1-P65.
- Ghayoumi, M. (2021). Deep Learning in Practice. Chapman and Hall/CRC.
- Gheorghe, A., Mahdi, L., & Musat, O. (2015). AGE-RELATED MACULAR DEGENERATION. Romanian journal of ophthalmology, 59(2), 74-77.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., & Bengio, Y. (2014). Generative adversarial nets. Advances in neural information processing systems, 27.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep learning. MIT press.
- Gonzalez, R. C., & Woods, R. E., (2018). Digital image processing (4th ed). Pearson Education Limited.



# DAFTAR PUSTAKA

- Hu, J., Shen, L., & Sun, G. (2018). Squeeze-and-excitation network. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7132-7141).
- Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In International conference on machine learning (pp. 448-456).
- Isola, P., Zhu, J. Y., Zhou, T., & Efros, A. A. (2017). Image-to-image translation with conditional adversarial network. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 1125-1134).
- Jauro, F., Chiroma, H., Gital, A. Y., Almutairi, M., Shafi'i, M. A., & Abawajy, J. H. (2020). Deep learning architectures in emerging cloud computing architectures: Recent development, challenges and next research trend. Applied Soft Computing, 96, 106582.
- Kadhim, Y. A., Khan, M. U., & Mishra, A. (2022). Deep Learning-Based Computer-Aided Diagnosis (CAD): Applications for Medical Image Datasets. Sensors (Basel, Switzerland), 22(22), 8999.
- Khan, A., Sohail, A., Zahoor, U., & Qureshi, A. S. (2020). A survey of the recent architectures of deep convolutional neural network. Artificial intelligence review, 53, 5455-5516.
- Kingma, D. P., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.
- Le, N. T., Le Truong, T., Pongsachareonnont, P. F., Suwajanakorn, D., Mavichak, A., Itthipanichpong, R., & Benjapolakul, W. (2023). Deep Learning Approach for Age-related Macular Degeneration Detection Using Retinal Images: Efficacy Evaluation of Different Deep Learning Models. Egyptian Informatics Journal, 24(4), 100402.
- Lin, M., Chen, Q., & Yan, S. (2013). Network in network. arXiv preprint arXiv:1312.4400.
- Maswood, M. M. S., Hussain, T., Khan, M. B., Islam, M. T., & Alharbi, A. G. (2020, November). Cnn based detection of the severity of diabetic retinopathy from the fundus photography using efficientnet-b5. In 2020 11th IEEE Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON) (pp. 0147-0150). IEEE.



# DAFTAR PUSTAKA

- Meruva, S., Tulasi, V., Vinnakota, N., & V, Bhavana. (2022). Risk Level Prediction of Diabetic Retinopathy based on Retinal Images using Deep Learning Algorithm. *Procedia Computer Science*. 215. 722-730. 10.1016/j.procs.2022.12.074.
- Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.
- Murphy, K. P. (2012). *Machine learning: a probabilistic perspective*. MIT press.
- Musa, P., Al Rafi, F., & Lamsani, M. (2018). A Review: Contrast-Limited Adaptive Histogram Equalization (CLAHE) methods to help the application of face recognition. In *2018 third international conference on informatics and computing (ICIC)* (pp. 1-6). IEEE.
- Pan, Z., Yu, W., Yi, X., Khan, A., Yuan, F., & Zheng, Y. (2019). Recent progress on generative adversarial network (GANs): A survey. *IEEE access*, 7, 36322-36333.
- Paradisa, R. H., Sarwinda, D., Bustamam, A., & Argyadiva, T. (2020, November). Classification of diabetic retinopathy through deep feature extraction and classic machine learning approach. In *2020 3rd International Conference on Information and Communications Technology (ICOIACT)* (pp. 377-381). IEEE.
- Prawira, R., Bustamam, A., & Anki, P. (2021, December). Multi label classification of retinal disease on fundus images using AlexNet and VGG16 architectures. In *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)* (pp. 464-468). IEEE.
- Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial network. *arXiv preprint arXiv:1511.06434*.
- Razi, F. A., Bustamam, A., & Latifah, A. L. (2023). Development of efficient brain tumor classification on MRI image results using EfficientNet. In *2023 International Seminar on Intelligent Technology and Its Applications (ISITIA)* (pp. 575-580). IEEE.
- Ruas, Terry & Grosky, William. (2020). *Software Engineering: A Practitioner's Approach 9 th Edition*
- Smaida, M., Yaroshchak, S., & El Barg, Y. (2021). DCGAN for Enhancing Eye Diseases Classification. In *CMIS* (pp. 22-33)..



# DAFTAR PUSTAKA

- Soleimanipour, A., Azadbakht, M., & Rezaei Asl, A. (2022). Cultivar identification of pistachio nuts in bulk mode through EfficientNet deep learning model. *Journal of Food Measurement and Characterization*, 16(4), 2545-2555.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1), 1929-1958.
- Tan, M., & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural network. In *International conference on machine learning* (pp. 6105-6114). PMLR.
- Veras, R., Medeiros, F., Silva, R., & Ushizima, D. (2013). Assessing the accuracy of macula detection methods in retinal images. In *2013 18th International Conference on Digital Signal Processing (DSP)* (pp. 1-6). IEEE.
- Virkus, S., & Garoufallou, E. (2019). Data science from a library and information science perspective. *Data Technologies and Applications*, 53(4), 422-441.
- Vyawahare, H., & Shinde, P. (2022). Age-related macular degeneration: Epidemiology, pathophysiology, diagnosis, and treatment. *Cureus*, 14(9).
- Wong, W. L., Su, X., Li, X., Cheung, C. M. G., Klein, R., Cheng, C. Y., & Wong, T. Y. (2014). Global prevalence of age-related macular degeneration and disease burden projection for 2020 and 2040: a systematic review and meta-analysis. *The Lancet Global Health*, 2(2), e106-e116.
- Xia, D. H., Song, S., Tao, L., Qin, Z., Wu, Z., Gao, Z., & Luo, J. L. (2020). material degradation assessed by digital image processing: Fundamentals, progresses, and challenges. *Journal of Materials Science & Technology*, 53, 146-162.
- Xu, Z., Liu, X., & Chen, X. (2009). Fog removal from video sequences using contrast limited adaptive histogram equalization. In *2009 International Conference on Computational Intelligence and Software Engineering* (pp. 1-4). IEEE.
- Zhang, Y. D., Satapathy, S. C., Liu, S., & Li, G. R. (2021). A five-layer deep convolutional neural network with stochastic pooling for chest CT-based COVID-19 diagnosis. *Machine vision and applications*, 32, 1-13.
- Zhu, S., Lu, B., Wang, C., Wu, M., Zheng, B., Jiang, Q., & Yang, W. (2022). Screening of common retinal diseases using six-category models based on EfficientNet. *Frontiers in medicine*, 9, 808402.



# Terima Kasih

Lab Bioinformatics  
and Advanced  
Computing



FACULTY OF  
MATHEMATICS AND  
NATURAL SCIENCES