



Nama: Zefanya Danovanta Tarigan (122140101), Abraham Ganda Napitu (122140095), Ithya Razky Hidayat (122140167)

Tugas: Eksperimen Arsitektur ResNet-34

Mata Kuliah: Pembelajaran Mendalam (IF25-40401)

Tanggal: 03 Oktober 2025

## 1 Pendahuluan

Residual Network (ResNet) adalah untuk mengatasi permasalahan degradasi pada deep neural networks dengan menggunakan skip connection sederhana berbentuk persamaan  $y=F(x)+x$ . Studi ini secara sistematis mengevaluasi ResNet-34 melalui tiga tahap eksperimen, yaitu menggunakan Plain-34 sebagai baseline untuk membuktikan adanya degradasi, menerapkan ResNet-34 untuk menunjukkan efektivitas residual learning dan Dataset yang digunakan terdiri dari 5 kelas makanan Indonesia dengan pembagian 80:20 (train:val), berjumlah 887 citra untuk pelatihan dan 221 citra untuk validasi.

## 2 Metodologi

### 2.1 Setup Parameter

Untuk memastikan proses pelatihan berjalan konsisten dan dapat direproduksi, penelitian ini menetapkan sejumlah hyperparameter utama yang digunakan selama training. Hyperparameter tersebut meliputi laju pembelajaran, jumlah batch, jumlah epoch, optimizer, serta fungsi loss yang digunakan. Rincian lengkap setup hyperparameter dapat dilihat pada Tabel 1.

Tabel 1: Setup Hyperparameter

Parameter	Value
Learning Rate	$1 \times 10^{-3}$
Batch Size	32
Epochs	10
Optimizer	AdamW
Weight Decay	$1 \times 10^{-4}$
Criterion	CrossEntropyLoss

Sebelum data dimasukkan ke dalam model, dilakukan tahap preprocessing agar citra memiliki ukuran yang seragam, distandarisasi, serta mendapat augmentasi dasar untuk meningkatkan kemampuan model. Proses preprocessing yang diterapkan pada dataset ini dirangkum dalam Tabel 2.

Tabel 2: Preprocessing Data

Parameter	Value
Image Size	$224 \times 224$
RandomHorizontalFlip	$p = 0.5$
Normalization Mean	[0.485, 0.456, 0.406]
Normalization Std	[0.229, 0.224, 0.225]
Random Seed	42

## 2.2 Setup Arsitektur Model

Tabel 3: Tabel Arsitektur yang Digunakan

Model	Deskripsi
Plain-34	Model CNN 34 layer tanpa residual connection, digunakan sebagai baseline untuk melihat masalah degradasi pada jaringan dalam.
ResNet-34	Model ResNet dengan 34 layer dan residual connection (skip connection) untuk mengatasi vanishing gradient dan degradasi performa.
SE-ResNet-34	Varian ResNet-34 yang ditambahkan modul Squeeze-and-Excitation (SE) untuk melakukan rekalisasi channel, meningkatkan fokus pada fitur penting.
Mish-ResNet-34	Varian ResNet-34 yang mengganti fungsi aktivasi ReLU dengan Mish untuk menghasilkan distribusi gradien lebih halus dan stabil.

## 3 Hasil & Analisis

### 3.1 Masalah Degradasi

Degradasi merupakan permasalahan ketika jaringan dengan lapisan yang lebih dalam tetapi menghasilkan performa yang lebih buruk dibandingkan jaringan yang lebih dangkal. Masalah ini berbeda dengan overfitting (di mana model hanya menghafal data latih), karena pada degradasi, model mengalami kesulitan untuk belajar sejak awal. Hal ini terlihat dari rendahnya akurasi pada data training, bukan hanya pada data validasi.

### 3.2 Bukti Degradasi

Tabel 4: Perbandingan Plain-34 dan ResNet-34 sebagai bukti degradasi

Model	Final Train Acc	Final Val Acc	Best Val Acc (Epoch)	Catatan
Plain-34	0.6027	0.5811	0.6036 (epoch 9)	Akurasi train dan validasi rendah, tidak meningkat signifikan meskipun lebih dalam. Hal ini menunjukkan terjadinya degradasi.
ResNet-34	0.8442	0.7883	0.7928 (epoch 9)	Dengan adanya residual connection, model lebih mudah dioptimasi dan mampu mencapai akurasi lebih tinggi sehingga menghindari degradasi performa.

**Analisis Degradasi :** Berdasarkan hasil eksperimen, Plain-34 menunjukkan performa yang jauh lebih rendah dibanding ResNet-34 meskipun keduanya memiliki kedalaman (34 lapis) yang sama. Perbandingan kuantitatif:

- Final Train Accuracy: Plain-34 = 0.6027, ResNet-34 = 0.8442 (selisih =  $0.2415 \approx 24.15\%$ ).
- Final Val Accuracy: Plain-34 = 0.5811, ResNet-34 = 0.7883 (selisih =  $0.2072 \approx 20.72\%$ ).
- Best Val Accuracy: Plain-34 = 0.6036 (epoch 9), ResNet-34 = 0.7928 (epoch 9) (selisih =  $0.1892 \approx 18.92\%$ ).

Perbedaan besar ini merupakan bukti empiris kuat bahwa menambah kedalaman tanpa mekanisme khusus (seperti residual connection) dapat menurunkan kemampuan optimisasi

#### Penyebab Degradasi :

- Tidak ada residual connection : informasi sinyal dan gradien harus melewati banyak layer tanpa shortcut, membuat gradien cepat hilang.
- Fungsi aktivasi ReLU standar : meski relatif stabil, tetap berpotensi menyebabkan banyak neuron mati (dead ReLU) sehingga memperparah kehilangan informasi.

- Optimisasi lambat : learning rate dan optimizer sama dengan ResNet, tapi arsitektur Plain tidak bisa memanfaatkannya secara efektif.

### 3.3 Residual Connection Mengatasi Degradasi

Residual connection berfungsi dengan menyalurkan informasi asli dari input secara langsung ke layer yang lebih dalam melalui skip connection. Dengan cara ini, informasi penting tetap terjaga dan aliran gradien tetap stabil saat proses backpropagation, sehingga jaringan dengan kedalaman lebih besar dapat dilatih tanpa menghadapi masalah degradasi kinerja.

Tabel 5: Perbandingan Plain-34 dan ResNet-34

Model	Final Val Acc	Best Val Acc (Epoch)
Plain-34	0.5811	0.6036 (epoch 9)
ResNet-34	0.7883	0.7928 (epoch 9)

**Performa Plain-34 VS ResNet-34.** Terdapat peningkatan  $\approx 20\%$  pada akurasi validasi ketika residual connection digunakan. Hal ini menunjukkan bahwa penambahan kedalaman dengan skip connection tidak menyebabkan degradasi, melainkan memperbaiki optimisasi dan generalisasi.

**residual connection efektif mengatasi degradasi karena:**

1. **Memperbaiki aliran gradien.** Jalur identitas memungkinkan gradien mengalir langsung ke lapisan awal, mengurangi risiko *vanishing gradient*.
2. **Fungsi identitas sebagai baseline.** Jika transformasi non-linear sulit dipelajari, blok residual tetap dapat merepresentasikan fungsi identitas. Ini menjamin performa model dalam tidak lebih buruk daripada model dangkal.
3. **Optimisasi lebih stabil.** Residual connection membuat lanskap loss lebih "halus" dan mempermudah pencarian parameter optimal.
4. **Kemampuan representasi lebih kuat.** Dengan menggabungkan informasi identitas dan hasil transformasi non-linear, jaringan dapat belajar fitur yang lebih kaya dan mendalam.

Residual connection terbukti secara mampu mengatasi masalah degradasi yang terjadi pada Plain-34. Dengan adanya jalur identitas, model dalam seperti ResNet-34 dapat dilatih dengan stabil, menghasilkan akurasi training dan validasi yang jauh lebih tinggi dibanding arsitektur tanpa residual. Hal ini menegaskan pentingnya residual connection dalam pengembangan arsitektur deep learning modern.

### 3.4 Analisis Arsitektur

#### 3.4.1 ResNet-34 + Residual Connection (RC)

Residual Connection menambahkan shortcut identity yang memungkinkan gradien mengalir tanpa hambatan meski kedalaman jaringan bertambah, sehingga mengatasi masalah degradasi pada Plain-34.

Tabel 6: Plain-34 vs ResNet-34

Model	Final Val Acc	Best Val Acc
Plain-34	0.5811	0.6036
ResNet-34	0.7883	0.7928

**Analisis:** Residual connection efektif mengatasi degradasi. Untuk jaringan dalam seperti ResNet-34, shortcut membantu mempertahankan stabilitas training.

### 3.4.2 ResNet-34 + Mish Activation

Mish bersifat *smooth*, sehingga aliran gradien lebih halus, mengurangi risiko *dead neuron problem* yang kadang muncul pada ReLU.

Tabel 7: ReLU vs Mish pada ResNet-34

Epoch	ReLU Train Acc	Mish Train Acc	ReLU Val Acc	Mish Val Acc
1	32.24%	34.18%	29.41%	31.55%
5	81.06%	82.77%	66.97%	68.44%
10	91.66%	92.85%	80.09%	81.72%
15	97.07%	97.68%	74.66%	79.81%

**Hasil:** Mish menunjukkan tren validasi yang lebih stabil dan sedikit lebih tinggi dibandingkan ReLU pada beberapa titik epoch.

**Analisis:** Mish berpotensi meningkatkan performa karena sifat smooth dan gradien yang lebih stabil. Pada ResNet-34, kombinasi residual connection + Mish dapat menjadi arsitektur lebih kuat daripada standar ResNet dengan ReLU.

### 3.5 Analisis Detail Eksperimen Arsitektur

**1. Residual Connection sebagai Solusi Fundamental:** Pada Plain-34, model mengalami degradasi performa dengan akurasi validasi hanya 60.36% dan training 60.27%. Ketika residual connection ditambahkan pada ResNet-34, performa meningkat signifikan dengan akurasi validasi mencapai 79.28% dan training 84.42%. Hal ini membuktikan bahwa skip connection bukan sekadar trik tambahan, melainkan solusi mendasar untuk mengatasi masalah degradasi pada jaringan yang lebih dalam.

**2. SE-Block paling unggul:** Model SE-ResNet-34 mencatat akurasi validasi tertinggi sebesar 81.98% pada epoch 10. Mekanisme *channel attention* membuat model mampu menyorot fitur penting sekaligus mereduksi fitur yang tidak relevan, sehingga lebih tahan terhadap noise dan overfitting.

**3. Mish meningkatkan stabilitas training:** Varian Mish-ResNet-34 menunjukkan performa validasi yang stabil, dengan best validation accuracy sebesar 77.93%. Mish membantu menghasilkan distribusi gradien lebih halus dibandingkan ReLU, meski pada eksperimen ini peningkatannya tidak melampaui ResNet standar. **4. Pre-activation belum digunakan:** Pada eksperimen ini hanya diuji ResNet standar, Mish-ResNet, dan SE-ResNet. Namun, dari literatur terdahulu, pre-activation biasanya lebih terasa manfaatnya pada arsitektur yang jauh lebih dalam (misalnya ResNet-110), sehingga pada ResNet-34 kontribusinya terbatas.

Tabel 8: Summary Semua Model (Best Performance)

Model	Best Epoch	Train Acc	Val Acc	Gain vs Plain
Plain-34	9	60.27%	60.36%	+0.09%
ResNet-34	9	84.09%	79.28%	+18.9%
Mish-ResNet-34	9	81.94%	77.93%	+17.6%
SE-ResNet-34	10	89.05%	81.98%	+21.6%

Residual connection terbukti menjadi kunci untuk mengatasi degradasi pada Plain-34. SE-Block menghasilkan akurasi validasi tertinggi karena mekanisme channel attention, sementara Mish memberi distribusi gradien lebih halus yang membuat training lebih stabil. Pada dataset terbatas, SE-ResNet-34 adalah varian terbaik dengan peningkatan signifikan terhadap Plain-34 maupun ResNet standar.

## 4 Lampiran

- M. D. A. Pranatha, M. A. Maricar, and G. H. Setiawan, "Implementasi Arsitektural ResNet-34 dalam Klasifikasi Gambar Penyakit pada Daun Kentang," *Jurnal Teknologi dan Sistem Informasi Bisnis*, vol. 6, no. 3, pp. 575–580, Jul. 2024.

- I. Santoso, A. M. Manurung, and E. R. Subhiyakto, “Comparison of ResNet-50, EfficientNet-B1, and VGG-16 Algorithms for Cataract Eye Image Classification,” *Journal of Applied Informatics and Computing*, vol. 9, no. 2, pp. 284–294, Apr. 2025.
- S. Lasniari, Jasril, S. Sanjaya, F. Yanto, and M. Affandes, “Pengaruh Hyperparameter CNN Arsitektur ResNet-50 pada Klasifikasi Citra Daging Sapi dan Daging Babi,” *Jurnal Nasional Komputasi dan Teknologi Informasi*, vol. 5, no. 3, pp. 474–482, Jun. 2022.
- D. A. Agustina, “Klasifikasi Citra Jenis Kulit Wajah dengan CNN ResNet-50,” *Jurnal Riset Sistem Informasi (JISSI)*, vol. 1, no. 3, pp. 01–07, Jul. 2024.
- Github Repository : [https://github.com/danovantaa/RestNet\\_Explore](https://github.com/danovantaa/RestNet_Explore)

**Catatan:** Plot gambar tersedia dalam folder di setiap modelnya

## 5 Peran dan Kontribusi AI Assistant

### 5.1 Prompt dan Masalah yang Diajukan

Pada proyek ini, AI Assistant digunakan untuk mendukung proses implementasi arsitektur ResNet-34 dan variasinya. GitHub Copilot dimanfaatkan untuk membantu penulisan kode, khususnya dalam membangun *training loop*, memodifikasi fungsi aktivasi dari ReLU menjadi Mish, serta integrasi *Squeeze-and-Excitation (SE) Block* ke dalam arsitektur ResNet-34.

Selain itu, Claude AI digunakan untuk menjelaskan konsep teoretis, seperti perbedaan antara ResNet standar dan ResNet dengan pre-activation, serta analisis mengenai masalah degradasi pada jaringan yang lebih dalam. AI juga dimanfaatkan dalam penyusunan deskripsi metodologi dan hasil eksperimen, sehingga mempercepat proses dokumentasi laporan.