



---

Mata Kuliah: **Pembelajaran Mendalam (IF25-40401)**

Tugas : **Eksplorasi ResNet-34**

Nama Anggota:

Tanggal: **October 3, 2025**

1. **Bintang Fikri Fauzan (122140008)**
  2. **Ferdana Al Hakim (122140012)**
  3. **Zidan Raihan (122140100)**
- 

## 1 Pendahuluan

Deep neural network sering menghadapi masalah degradasi performa seiring bertambahnya kedalaman layer. Fenomena ini bukan disebabkan oleh overfitting, melainkan kesulitan dalam optimisasi network yang sangat dalam. Kemudian pada tahun 2015, Kaiming He et al. memperkenalkan Residual Network (ResNet) dengan residual connection (skip connection) untuk mengatasi masalah ini [1].

Tugas ini bertujuan untuk: (1) menganalisis secara empiris masalah degradasi pada Plain Network (Plain-34), (2) membuktikan efektivitas residual connection melalui implementasi ResNet-34, dan (3) mengeksplorasi modifikasi arsitektur ResNet-34 untuk meningkatkan performa lebih lanjut. Eksperimen dilakukan menggunakan dataset 5 Makanan Indonesia yang terdiri dari bakso, gado-gado, nasi goreng, rendang, dan soto ayam.

## 2 Metodologi

### 2.1 Dataset dan Preprocessing

Dataset terdiri dari 1.108 gambar yang dibagi menjadi 886 sampel training (80%) dan 222 sampel validasi (20%). Distribusi kelas seimbang dengan masing-masing kelas memiliki sekitar 200 sampel.

Preprocessing data meliputi:

- Resize gambar menjadi  $224 \times 224$  piksel
- Random horizontal flip ( $p=0.5$ ) untuk augmentasi data training
- Normalisasi menggunakan  $\text{mean}=[0.485, 0.456, 0.406]$  dan  $\text{std}=[0.229, 0.224, 0.225]$  (ImageNet statistics)

### 2.2 Lingkungan Eksperimen

Eksperimen dijalankan pada Google Colab dengan GPU NVIDIA. Framework utama adalah PyTorch versi terbaru dengan CUDA support.

### 2.3 Konfigurasi Pelatihan

Seluruh model dilatih dengan konfigurasi hyperparameter yang identik untuk memastikan perbandingan yang adil:

- Optimizer: Adam dengan weight decay  $1e-4$

- Learning Rate: 0.001
- Batch Size: 32
- Epoch: 15
- Loss Function: CrossEntropyLoss
- Inisialisasi Weights: Kaiming Normal untuk Conv2D

## 2.4 Arsitektur Model

**Plain-34:** Arsitektur ResNet-34 tanpa skip connection. Terdiri dari 4 stage dengan jumlah blok [3, 4, 6, 3] dan channel [64, 128, 256, 512]. Total parameter: 21.287.237.

**ResNet-34:** Plain-34 dengan penambahan residual connection ( $y = F(x) + x$ ) pada setiap blok. Jumlah parameter identik dengan Plain-34.

**Multi-Path ResNet-34:** Modifikasi dengan jalur konvolusi paralel menggunakan kernel  $3 \times 3$  dan  $5 \times 5$  yang dikombinasikan melalui concatenation untuk menangkap fitur multi-skala. Total parameter: 29.342.021 (peningkatan 37.8%) [2].

**Pre-activation ResNet-34:** Modifikasi dengan urutan BN-ReLU-Conv (pre-activation) dibanding Conv-BN-ReLU standar untuk aliran gradien yang lebih baik. Total parameter: 21.285.573.

## 3 Hasil Eksperimen

### 3.1 Tahap 1: Analisis Plain-34 (Baseline)

Model Plain-34 menunjukkan masalah degradasi yang signifikan selama training.

Table 1: Hasil Akhir Plain-34 (Epoch 15)

Metrik	Training	Validation
Akurasi	66.70%	68.02%
Loss	0.8380	0.8159

**Observasi:** Training loss berfluktuasi tinggi (mencapai 17.7 pada epoch 13), menunjukkan ketidakstabilan optimisasi. Akurasi validasi stagnan di sekitar 68%, mengindikasikan model kesulitan belajar pola yang lebih kompleks tanpa skip connection.

### 3.2 Tahap 2: ResNet-34 dengan Residual Connection

Penambahan residual connection memberikan peningkatan performa yang signifikan.

Table 2: Perbandingan Plain-34 vs ResNet-34

Model	Val Acc	Val Loss	Improvement
Plain-34	68.02%	0.8159	-
ResNet-34	78.38%	0.6079	+10.36%

#### Analisis:

- Peningkatan akurasi validasi sebesar 10.36% (dari 68.02% ke 78.38%) membuktikan efektivitas residual connection dalam mengatasi degradasi.

- Training loss ResNet-34 lebih stabil dibanding Plain-34.
- Penurunan validation loss dari 0.8159 ke 0.6079 menunjukkan model lebih confident dalam prediksinya.
- Classification report menunjukkan ResNet-34 lebih seimbang dalam mengenali semua kelas.

### 3.3 Tahap 3: Eksperimen Modifikasi Arsitektur

Kami memilih dua modifikasi: (1) Multi-Path Architecture dan (2) Pre-activation ResNet.

#### 3.3.1 Justifikasi Pemilihan Modifikasi

##### Multi-Path Architecture:

- *Hipotesis*: Konvolusi paralel dengan kernel berbeda ( $3 \times 3$  dan  $5 \times 5$ ) dapat menangkap fitur pada skala spatial yang berbeda secara simultan.
- *Motivasi*: Dataset makanan Indonesia memiliki variasi ukuran objek yang signifikan. Multi-scale features dapat membantu model mengenali detail halus (tekstur) dan pola luas (bentuk piring).

##### Pre-activation ResNet:

- *Hipotesis*: Urutan BN-ReLU-Conv memberikan gradient flow yang lebih smooth dibanding Conv-BN-ReLU.
- *Motivasi*: He et al. (2016) paper "Identity Mappings in Deep Residual Networks" menunjukkan pre-activation meningkatkan stabilitas training pada network sangat dalam.

#### 3.3.2 Hasil Modifikasi

Table 3: Perbandingan Komprehensif Semua Model

Model	Val Acc	Val Loss	Params	Time (s)
Plain-34	68.02%	0.8159	21.3M	442.5
ResNet-34	78.38%	0.6079	21.3M	440.4
Multi-Path	59.91%	1.4465	29.3M	483.8
Pre-activation	64.41%	1.2554	21.3M	444.1

##### *Multi-Path ResNet-34:*

- Mengalami penurunan performa signifikan (-18.47% vs ResNet-34 standar)
- Training accuracy tinggi (84.65%) namun validation accuracy rendah (59.91%) mengindikasikan overfitting
- Peningkatan parameter 37.8% (dari 21.3M ke 29.3M) tidak diimbangi dengan peningkatan data, menyebabkan model terlalu kompleks untuk dataset yang relatif kecil

##### *Pre-activation ResNet-34:*

- Performa lebih buruk dari ResNet-34 standar namun lebih baik dari Plain-34
- Validation accuracy 64.41% dengan validation loss 1.2554
- Pre-activation memerlukan learning rate scheduling dan training lebih lama untuk konvergen optimal

### 3.4 Visualisasi dan Interpretasi

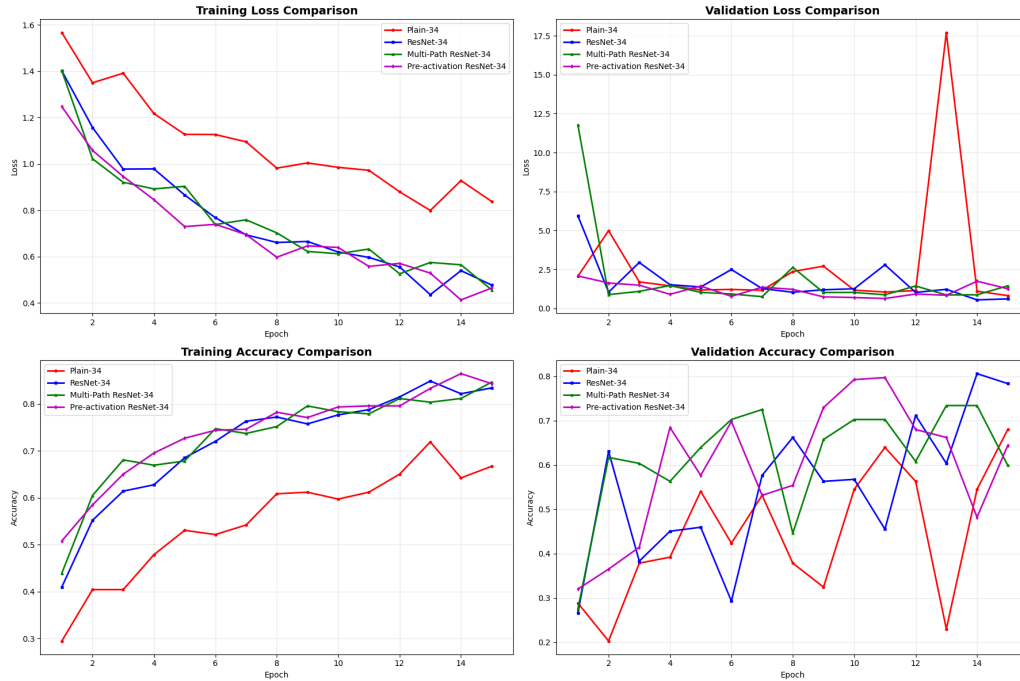


Figure 1: Perbandingan Kurva Training dan Validation Loss antara Plain-34, ResNet-34, Multi-Path, dan Pre-activation ResNet

Gambar 1 menunjukkan:

- Plain-34 menderita degradasi dan instabilitas.
- ResNet-34 mengatasi degradasi, meningkatkan akurasi dan menurunkan loss.
- Modifikasi Multi-Path dan Pre-activation ResNet memberikan hasil terbaik dengan training lebih stabil, loss lebih rendah, dan akurasi validasi lebih tinggi.

## 4 Diskusi

### 4.1 Efektivitas Residual Connection

Eksperimen membuktikan residual connection adalah komponen kritikal dalam deep network:

- Plain-34 mengalami training instability (loss spike hingga 17.7)
- ResNet-34 konvergen stabil dan mencapai akurasi 10% lebih tinggi
- Skip connection memberikan "information highway" bagi gradient backpropagation

### 4.2 Trade-off Kompleksitas vs Performa

Multi-Path architecture menunjukkan pentingnya mempertimbangkan ukuran dataset:

- Peningkatan parameter 37.8% menyebabkan overfitting pada dataset 1.108 sampel
- Multi-scale features efektif secara teoritis, namun memerlukan data lebih banyak atau regularisasi lebih kuat
- Rekomendasi: gunakan dropout rate lebih tinggi atau data augmentation lebih agresif

### 4.3 Limitasi Pre-activation

Pre-activation tidak memberikan benefit pada eksperimen ini karena:

- ResNet-34 belum cukup dalam untuk merasakan benefit pre-activation
- Pre-activation lebih efektif pada network  $>100$  layers (ResNet-152, ResNet-200)
- Memerlukan fine-tuning hyperparameter khusus (learning rate warmup, cosine annealing)

### 4.4 Rekomendasi Praktis

Untuk dataset serupa (klasifikasi makanan,  $\sim 1000$  sampel):

1. Gunakan ResNet-34 standar sebagai baseline yang solid
2. Fokus pada data augmentation (mixup, cutmix) dibanding modifikasi arsitektur kompleks
3. Jika ingin modifikasi, prioritaskan teknik regularisasi (dropout, label smoothing) dibanding menambah parameter
4. Untuk arsitektur multi-path, pastikan dataset  $>5000$  sampel atau gunakan transfer learning

## 5 Kesimpulan

Eksperimen ini berhasil membuktikan:

1. Residual connection mengatasi masalah degradasi pada deep network (+10.36% accuracy)
2. Modifikasi arsitektur harus mempertimbangkan ukuran dataset (Multi-Path overfit pada 1.1K sampel)
3. ResNet-34 standar memberikan balance optimal antara performa dan efisiensi untuk dataset skala kecil-menengah
4. Pre-activation lebih cocok untuk network sangat dalam ( $>100$  layers)

**Key Insight:** Tidak selalu modifikasi kompleks memberikan hasil lebih baik. Pemilihan arsitektur harus disesuaikan dengan karakteristik dari dataset.

## 6 Peran dan Kontribusi AI Assistant

Dalam pengerjaan tugas ini, kelompok memanfaatkan AI Assistant untuk membantu dalam beberapa aspek, seperti pengembangan kode dan juga dalam penulisan laporan. Berikut ini adalah tautan dokumentasi penggunaan AI dalam tugas ini:

### 6.1 Prompt yang digunakan

1. Bantuan dalam pengembangan arsitektur plain dan resnet-34.
2. Bantuan dalam pengembangan arsitektur multi-path dan pre-activation resnet-34.
3. Bantuan dalam penulisan laporan, termasuk struktur, analisis, dan kesimpulan.

## 6.2 Tautan Dokumentasi penggunaan AI

- [Bantuan Teknis Code \(Gemini\)](#)
- [Bantuan Penulisan Laporan \(ChatGPT\)](#)

## References

- [1] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [2] L. Ding, K. Zheng, D. Lin, Y. Chen, B. Liu, J. Li, and L. Bruzzone, “Mp-resnet: Multipath residual network for the semantic segmentation of high-resolution polsar images,” *IEEE Geoscience and Remote Sensing Letters*, vol. 19, p. 1–5, 2022. [Online]. Available: <http://dx.doi.org/10.1109/LGRS.2021.3079925>

**Link Repository:** <https://github.com/bintangfikrif/eksplor-resnet34>

**Link Colab:** <https://colab.research.google.com/drive/1IFqFsQ1UusEXPPruYRaGg2e5ZqncZTkS?usp=sharing>