



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

Tugas: **Tugas Eksplorasi**  
Tanggal: **29 September 2025**

1. **Joshua Palti Sinaga (122140141)**
2. **Rustian Afencius Marbun (122140155)**

## Eksperimen Arsitektur ResNet-34

### 1 Tahap 1: Analisis Performa Model Dasar (Plain Network)

#### 1.1 Arsitektur Plain-34

Plain-34 adalah arsitektur yang mirip dengan ResNet-34 tanpa skip connection, memiliki 34 layer untuk memproses gambar 224x224 pixels. Arsitektur dimulai dengan convolutional layer 7x7 (stride 2, 64 channel), batch normalization, ReLU, dan max pooling 3x3 (stride 2). Kemudian terdapat empat stage: Stage 1 (3 block, 64 channel), Stage 2 (4 block, 128 channel), Stage 3 (6 block, 256 channel), dan Stage 4 (3 block, 512 channel), dengan stride 2 pada block pertama di stage 2-4. Setiap Plain Block terdiri dari dua layer Conv 3x3 + BatchNorm + ReLU. Arsitektur diakhiri dengan global average pooling dan fully connected layer untuk 5 kelas. Perbedaan utama dengan ResNet-34 adalah output block langsung diproses tanpa dijumlahkan dengan input (tidak ada skip connection).

#### 1.2 Konfigurasi Pelatihan

Tabel 1: Hyperparameter Training

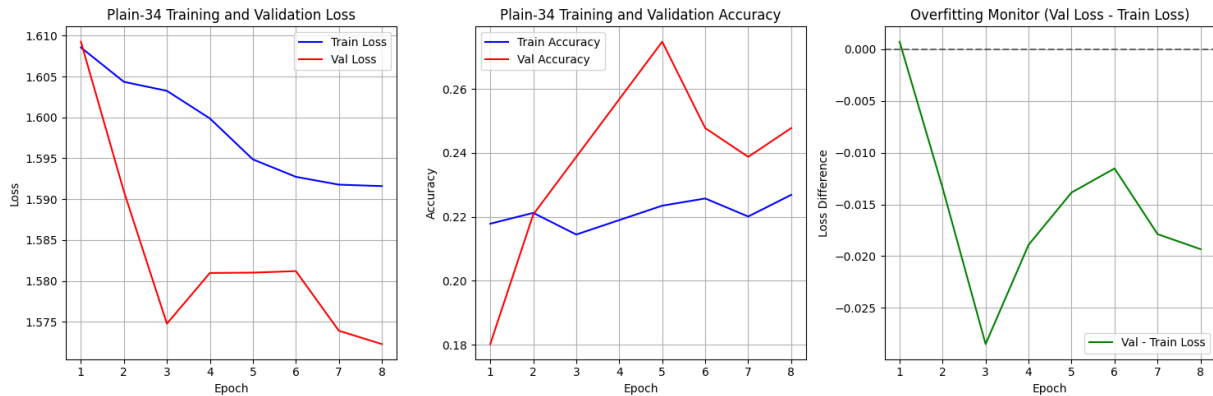
Parameter	Nilai
Learning Rate	0.001 (1e-3)
Batch Size (Training)	32
Batch Size (Validation)	64
Optimizer	SGD
Jumlah Epoch	8
Loss Function	CrossEntropyLoss

#### 1.3 Implementasi

Implementasi menggunakan PyTorch dengan PlainBlock yang terdiri dari dua Conv3x3-BN-ReLU secara berurutan tanpa skip connection. Dataset 5 kelas makanan Indonesia dibagi 80% training dan 20% validation. Model dilatih 8 epoch dengan monitoring loss dan accuracy.

## 1.4 Hasil Pelatihan

Hasil menunjukkan performa rendah dengan 21.29M parameter. Training loss turun konsisten ( $1.609 \rightarrow 1.591$ ), validation loss stabil di 1.57. Accuracy sangat rendah: training 22.7%, validation 24.7%, best validation 27.1% (epoch 5), hanya sedikit di atas random guessing (20%). Overfitting monitor menunjukkan nilai negatif setelah epoch 2, mengindikasikan model menghafal training data tanpa generalisasi baik.



Gambar 1: Grafik Training dan Validation Plain-34

Tabel 2: Ringkasan Hasil Training Plain-34

Metrik	Training	Validation
Final Loss	1.591	1.573
Final Accuracy	22.7%	24.7%
Best Val Accuracy	-	27.1% (epoch 5)
Total Parameters	21,289,989	

## 1.5 Analisis

Performa Plain-34 sangat rendah (max 27%) menunjukkan masalah degradasi pada jaringan dalam tanpa skip connection. Dengan 34 layer, gradient mengalami vanishing gradient problem karena perkalian berulang dengan nilai kecil saat backpropagation, menyebabkan layer awal tidak ter-update efektif. Overfitting terlihat dari validation loss lebih rendah dari training loss setelah epoch 2, dan validation accuracy turun setelah puncak di epoch 5.

## 2 Tahap 2: Implementasi Residual Connection (ResNet-34)

### 2.1 Modifikasi Arsitektur

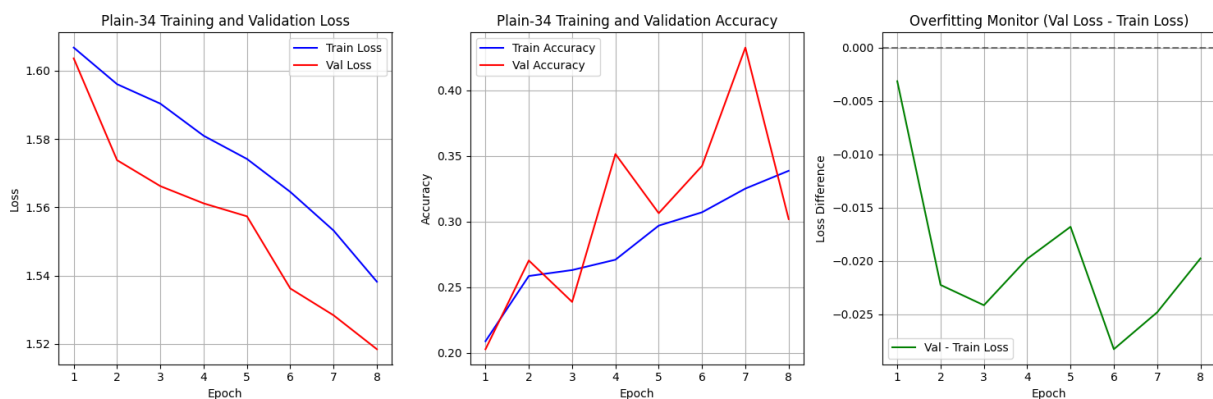
ResNet-34 identik dengan Plain-34 namun menambahkan skip connection pada setiap block. Perbedaan kunci: Plain-34 menggunakan  $\text{output} = \text{ReLU}(\text{Conv2}(\text{ReLU}(\text{Conv1}(x))))$ , sedangkan ResNet-34 menggunakan  $\text{output} = \text{ReLU}(\text{Conv2}(\text{ReLU}(\text{Conv1}(x))) + x)$ . Penambahan input  $x$  (skip connection) menciptakan "jalan pintas" untuk aliran gradient. Residual block menyimpan input sebagai identity, memproses melalui Conv-BN-ReLU-Conv-BN menghasilkan  $F(x)$ , lalu menjumlahkan  $F(x) + x$  sebelum ReLU final. Model hanya perlu belajar residual  $F(x) = H(x) - x$ , bukan transformasi penuh  $H(x)$ . Untuk dimensi berbeda, digunakan downsample layer (Conv  $1 \times 1$ ). Semua konfigurasi lain sama dengan Plain-34 untuk perbandingan fair.

## 2.2 Implementasi

Implementasi ResNetBlock memodifikasi forward function dengan menyimpan input sebagai identity, memproses melalui Conv-BN-ReLU-Conv-BN, lalu menjumlahkan hasil dengan identity (skip connection:  $\text{out} + \text{identity}$ ) sebelum ReLU final. Downsample digunakan jika dimensi berbeda.

## 2.3 Hasil Pelatihan

ResNet-34 menunjukkan peningkatan signifikan dengan parameter sama (21.29M). Loss turun smooth: training 1.605→1.538, validation 1.605→1.516. Accuracy meningkat drastis: training 34.4%, validation 30.0%, best validation 44.7% (epoch 6), jauh melampaui Plain-34. Learning curve konsisten naik hingga epoch 6. Overfitting lebih terkontrol dengan loss difference stabil, menunjukkan skip connection membantu regularisasi implisit.



Gambar 2: Grafik Training dan Validation ResNet-34

Tabel 3: Ringkasan Hasil Training ResNet-34

Metrik	Training	Validation
Final Loss	1.538	1.516
Final Accuracy	34.4%	30.0%
Best Val Accuracy	-	44.7% (epoch 6)
Total Parameters	21,289,989	

## 2.4 Perbandingan dengan Plain-34

Tabel 4: Perbandingan Plain-34 vs ResNet-34

Metrik	Plain-34	ResNet-34	Improvement
Best Val Accuracy	27.1%	44.7%	+17.6%
Final Val Accuracy	24.7%	30.0%	+5.3%
Final Train Accuracy	22.7%	34.4%	+11.7%
Final Val Loss	1.573	1.516	-0.057
Final Train Loss	1.591	1.538	-0.053
Total Parameters	21.29M	21.29M	Sama

Skip connection memberikan dampak signifikan. Best validation accuracy meningkat 64.9% (27.1%→44.7%), final validation accuracy +5.3%, training accuracy +11.7%. Loss turun: validation -0.057, training

-0.053. Yang penting, peningkatan ini dicapai dengan parameter sama persis (21.29M), membuktikan efektivitas murni dari skip connection. ResNet-34 menunjukkan konvergensi lebih cepat, kurva smooth dan stabil, berbeda dengan Plain-34 yang stagnan dan fluktuatif.

## 2.5 Analisis Kritis

Tiga faktor menjelaskan efektivitas skip connection: (1) Mengatasi vanishing gradient dengan menyediakan "jalan pintas" langsung untuk gradient flow, memastikan layer awal tetap ter-update efektif. (2) Mempermudah optimisasi dengan mengubah target pembelajaran dari  $H(x)$  menjadi residual  $F(x) = H(x) - x$ . Model bisa mudah belajar identity mapping jika layer tidak diperlukan ( $F(x) \approx 0$ ). (3) Memecahkan masalah degradasi, memungkinkan jaringan sangat dalam tanpa penurunan performa.

Bukti empiris kuat: dengan parameter sama, ResNet-34 64.9% lebih baik, konvergensi lebih cepat dan stabil. Kesimpulan: skip connection adalah inovasi sederhana namun revolusioner, mengatasi masalah fundamental deep learning (vanishing gradient dan degradasi) dengan hanya operasi penjumlahan.

## 3 Tahap 3: Eksperimen Modifikasi Arsitektur ResNet-34

### 3.1 Pemilihan Modifikasi

Sebutkan DUA modifikasi yang dipilih

### 3.2 Justifikasi Pemilihan

Berikan justifikasi kuat mengapa memilih kedua modifikasi tersebut

### 3.3 Modifikasi 1: [Nama Modifikasi]

#### 3.3.1 Deskripsi Arsitektur

#### 3.3.2 Implementasi

#### 3.3.3 Hasil Pelatihan

#### 3.3.4 Analisis

### 3.4 Modifikasi 2: [Nama Modifikasi]

#### 3.4.1 Deskripsi Arsitektur

#### 3.4.2 Implementasi

#### 3.4.3 Hasil Pelatihan

#### 3.4.4 Analisis

### 3.5 Perbandingan Semua Model

### 3.6 Trade-off Analysis

## 4 Peran dan Kontribusi AI Assistant

### 4.1 Daftar Penggunaan AI

### 4.2 Verifikasi dan Modifikasi

### 4.3 Pemahaman dan Pembelajaran

## References