

Optimizing ConvNeXt for TinyImageNet Classification using Self-Attention and Aggressive Regularization

PROPOSED IMPROVEMENT FOR CONVNEXT FROM THE PAPER A CONVNET FOR THE 2020S

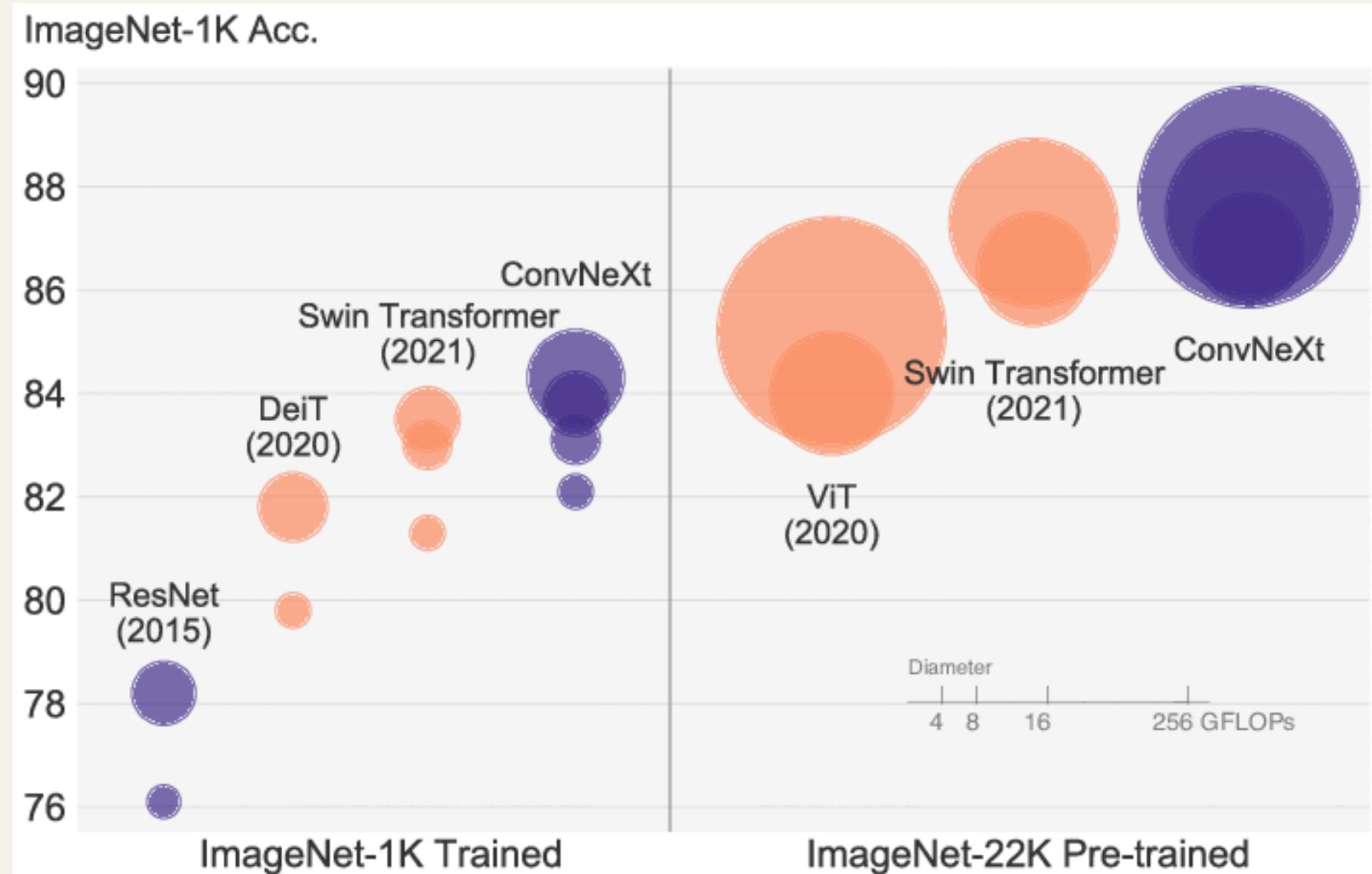
Proposed by

- 1 Nadhif Adita Fernanda
- 2 Bisma Adika Kusuma Putra
- 3 Farhan Vier Syarif Hilmi
- 4 Khairumam Fikri
- 5 Reyno Benedict

Latar Belakang

Convolutional Neural Networks (CNN) telah lama menjadi fondasi utama dalam bidang computer vision karena kemampuannya mengekstraksi fitur hierarkis dari citra secara efektif. CNN banyak digunakan dalam berbagai aplikasi seperti pengenalan objek, sistem pengawasan, dan teknologi smart city.

Namun, seiring meningkatnya kompleksitas tugas visual, keterbatasan arsitektur CNN tradisional mulai terlihat. Munculnya Vision Transformer yang memanfaatkan mekanisme self-attention mendorong pengembangan arsitektur CNN modern, salah satunya adalah ConvNeXt.

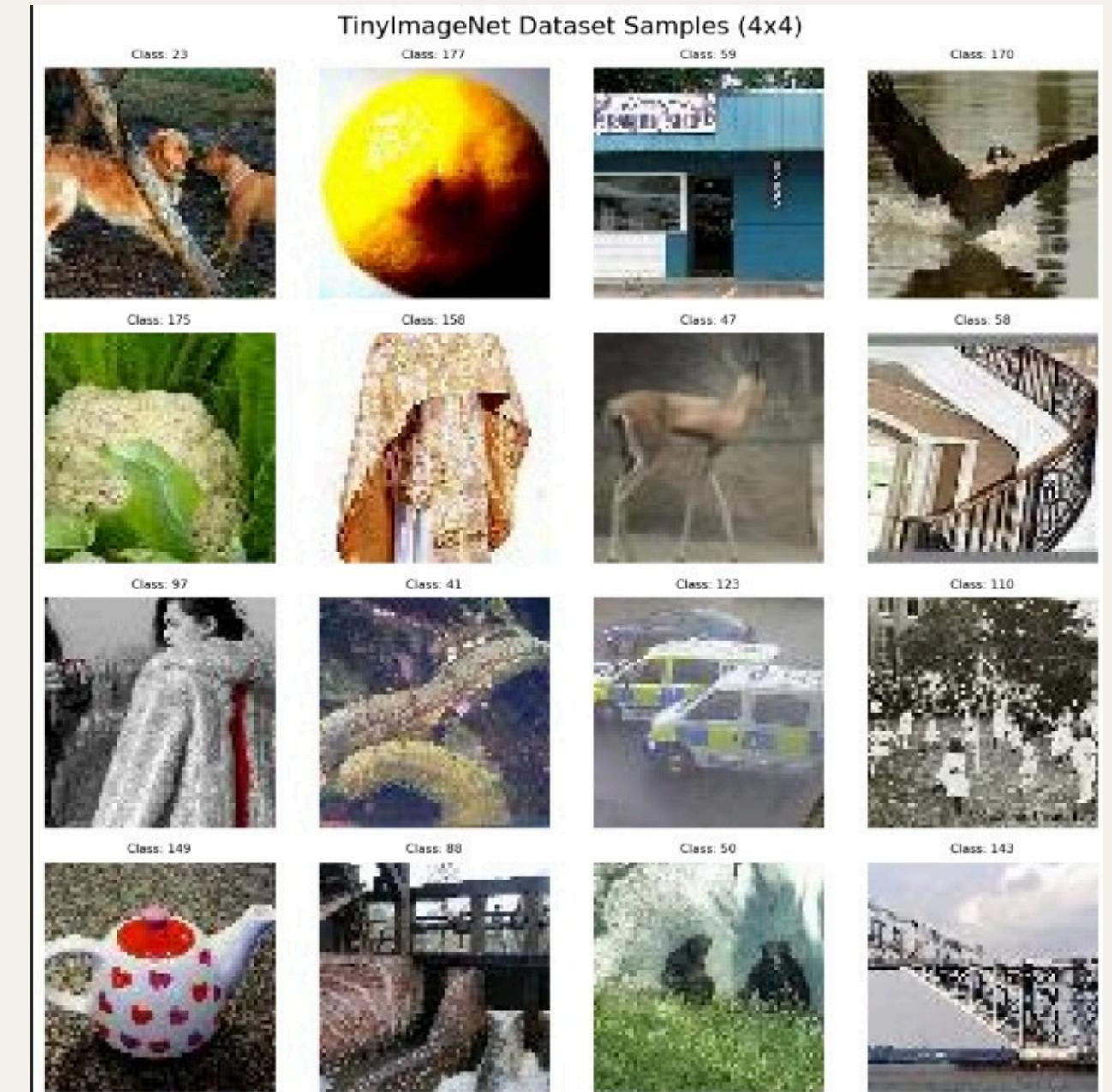


Model ConvNet dapat mencapai tingkat skalabilitas yang sama dengan Transformer visi hierarkis, namun dengan desain yang jauh lebih sederhana.

Masalah Penelitian

Meskipun **ConvNeXt** menunjukkan performa yang sangat baik pada dataset berskala besar seperti ImageNet-1K, penerapannya pada dataset kecil masih menghadapi tantangan serius. Model dengan kapasitas besar cenderung mengalami overfitting ketika jumlah data terbatas.

Tiny-ImageNet, dengan resolusi rendah dan kemiripan visual antar kelas yang tinggi, menjadi studi kasus yang tepat untuk menguji sejauh mana ConvNeXt dapat digeneralisasi pada kondisi data terbatas.

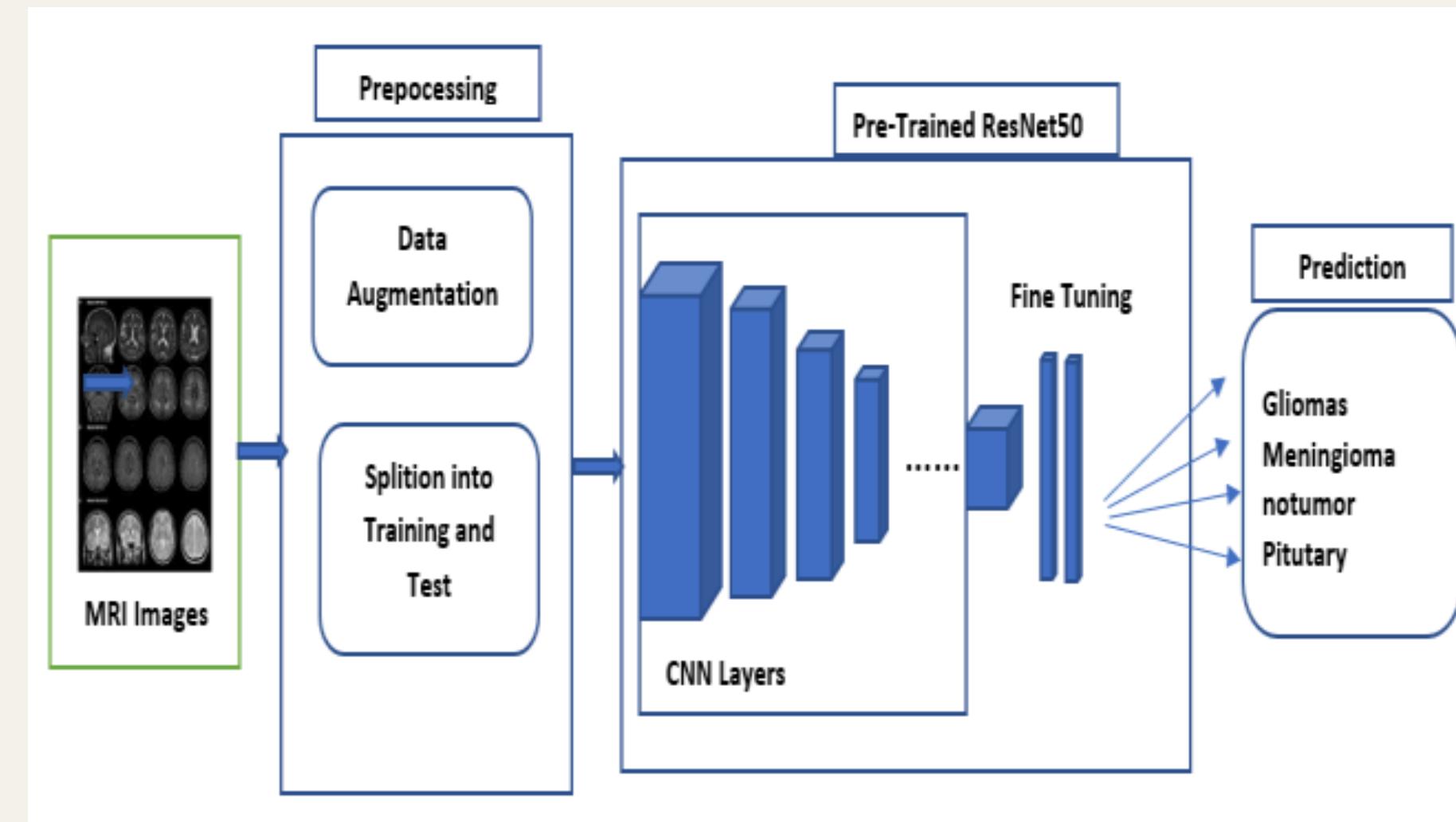


gambar Tiny-ImageNet dengan objek kecil dan background kompleks

Tujuan Penelitian

Penelitian ini bertujuan untuk **menevaluasi performa ConvNeXt-Tiny** pada Tiny-ImageNet dengan dua pendekatan pelatihan, yaitu konfigurasi baseline berbasis transfer learning dan konfigurasi yang ditingkatkan melalui fine-tuning penuh.

Selain itu, penelitian ini juga bertujuan untuk mengkaji efektivitas integrasi self-attention ringan, data augmentation berbasis RandAugment, serta regularisasi agresif dalam meningkatkan generalisasi model.



Contoh Salah satu CNN Fine Tuning Model

Dataset

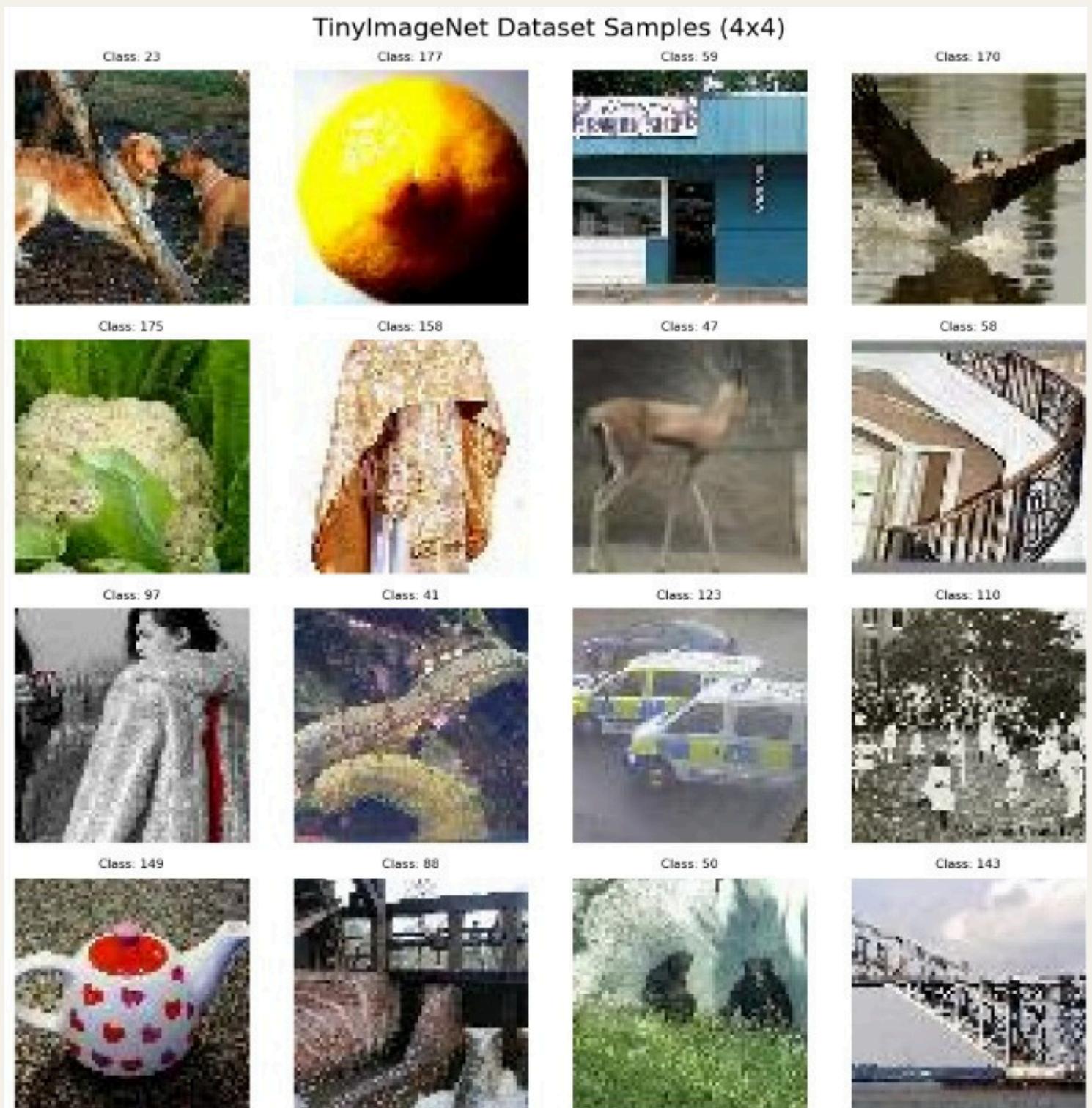
Tiny-ImageNet Dataset

[link dataset](#)

Tiny-ImageNet merupakan subset resmi dari ImageNet yang dirancang sebagai benchmark untuk evaluasi model klasifikasi citra pada skala menengah. Dataset ini mempertahankan keragaman semantik ImageNet dengan ukuran data yang jauh lebih ringan, sehingga cocok untuk eksperimen dan fine-tuning model deep learning.

Deskripsi Dataset

Attribute	Description
Number of classes	200
Training images	100,000 (500 per class)
Validation images	10,000 (50 per class)
Test images	10,000 (50 per class)
Image resolution	64 × 64 pixels
Task type	Multi-class image classification

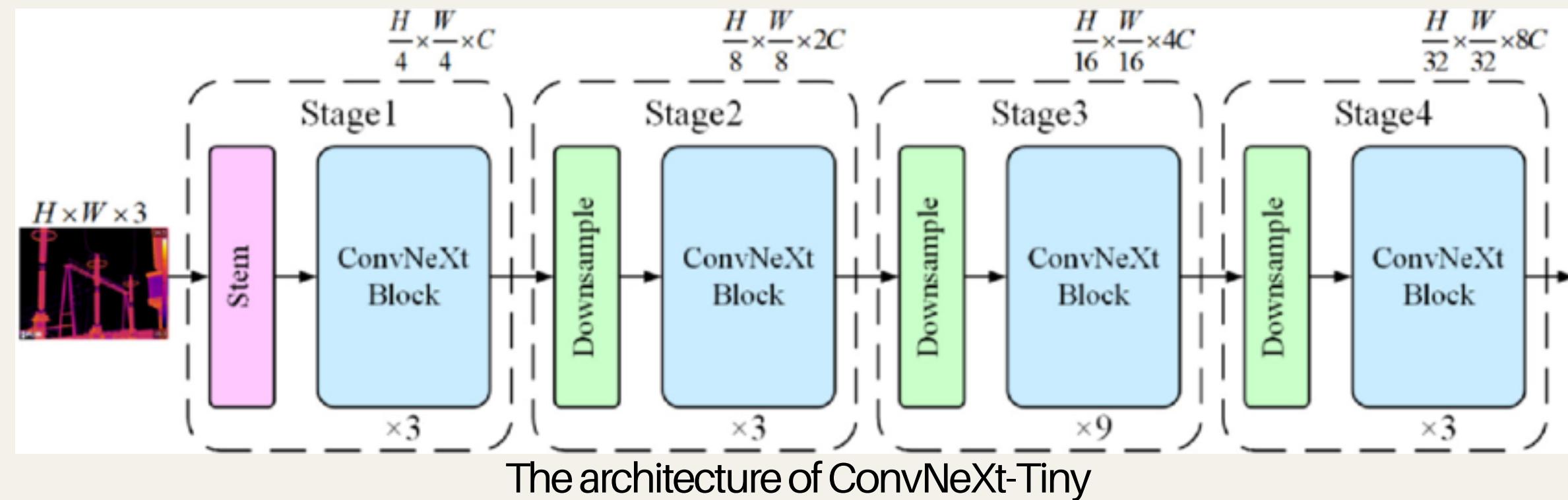


Isi Gambar pada Tiny-ImageNet Dataset

Arsitektur Convnext

ConvNeXt merupakan arsitektur CNN modern yang mengadopsi prinsip desain Vision Transformer tanpa meninggalkan operasi konvolusi. Beberapa pembaruan utama meliputi penggunaan kernel besar, inverted bottleneck, Layer Normalization, serta penyederhanaan struktur stage.

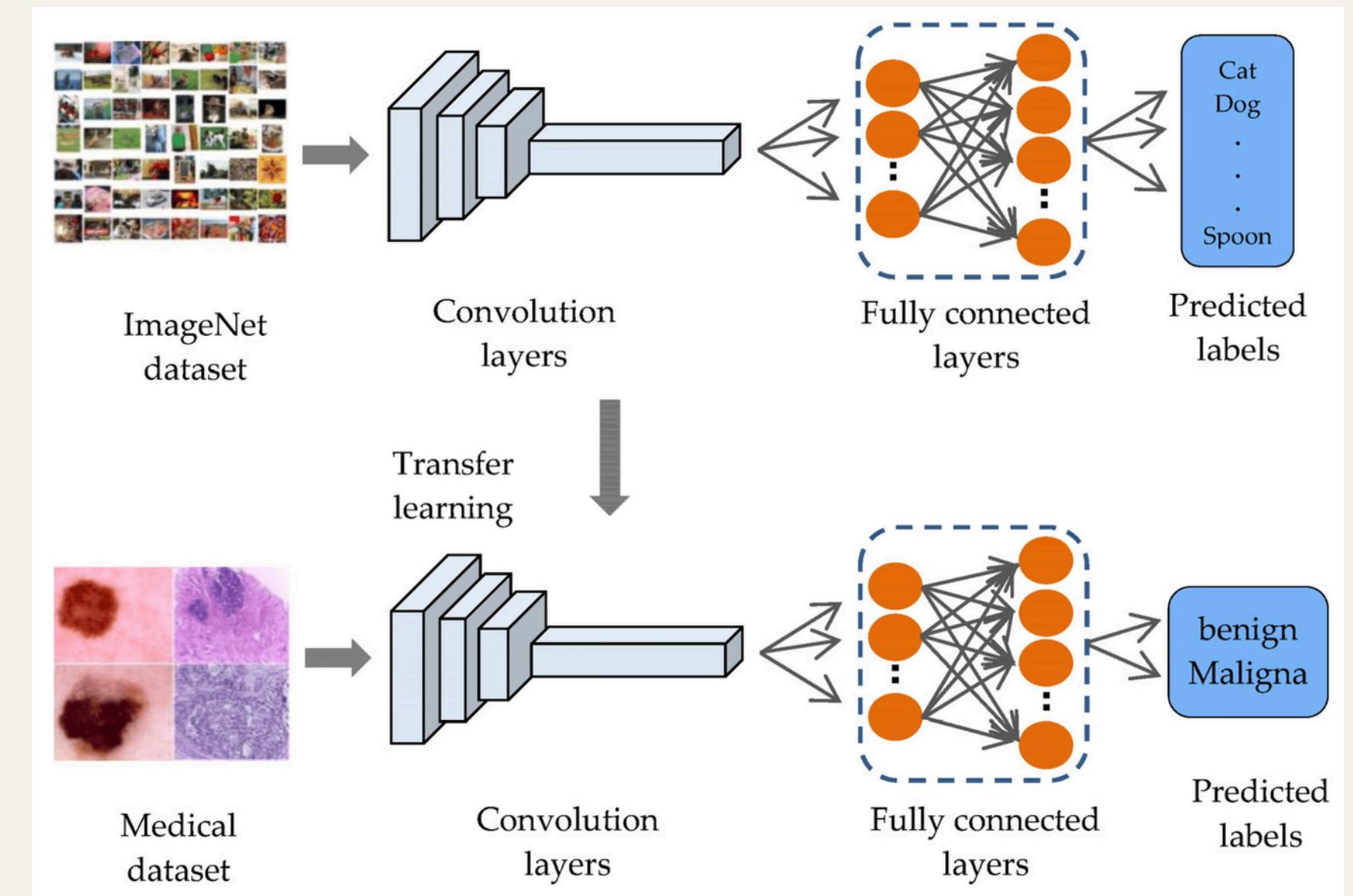
Dengan pendekatan ini, ConvNeXt mampu menutup kesenjangan performa antara CNN dan Transformer, sekaligus mempertahankan efisiensi komputasi dan bias induktif konvolusi.



Transfer Learning

ConvNeXt merupakan arsitektur CNN modern yang mengadopsi prinsip desain Vision Transformer tanpa meninggalkan operasi konvolusi. Beberapa pembaruan utama meliputi penggunaan kernel besar, inverted bottleneck, Layer Normalization, serta penyederhanaan struktur stage.

Dengan pendekatan ini, ConvNeXt mampu menutup kesenjangan performa antara CNN dan Transformer, sekaligus mempertahankan efisiensi komputasi dan bias induktif konvolusi.



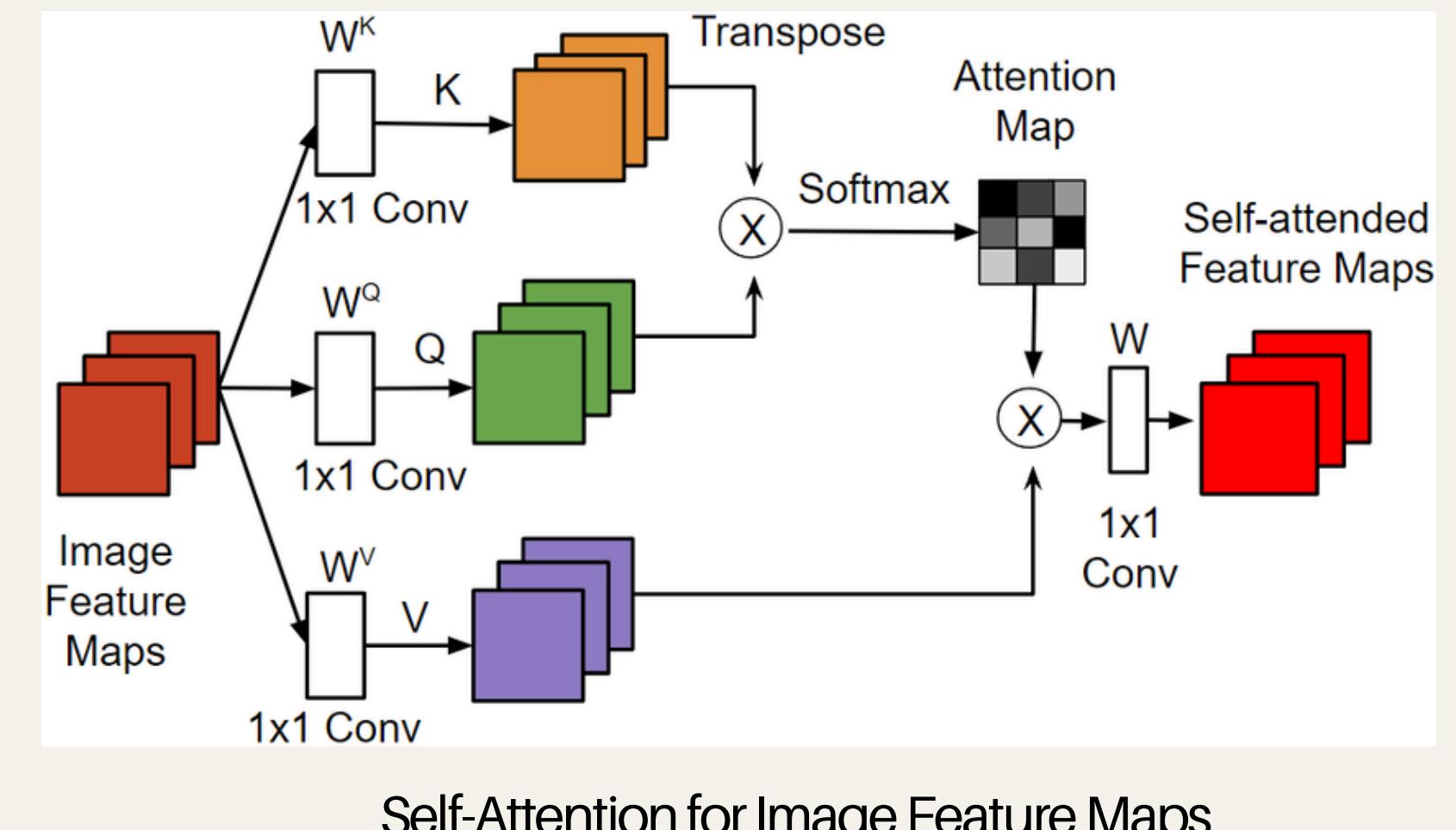
Cara Kerja Transfer Learning

Mekanisme Attention

Mekanisme attention memungkinkan model untuk memfokuskan perhatian pada fitur yang paling relevan.

Dalam CNN, attention dapat membantu menyoroti channel atau area spasial yang penting, sekaligus menekan fitur yang kurang informatif.

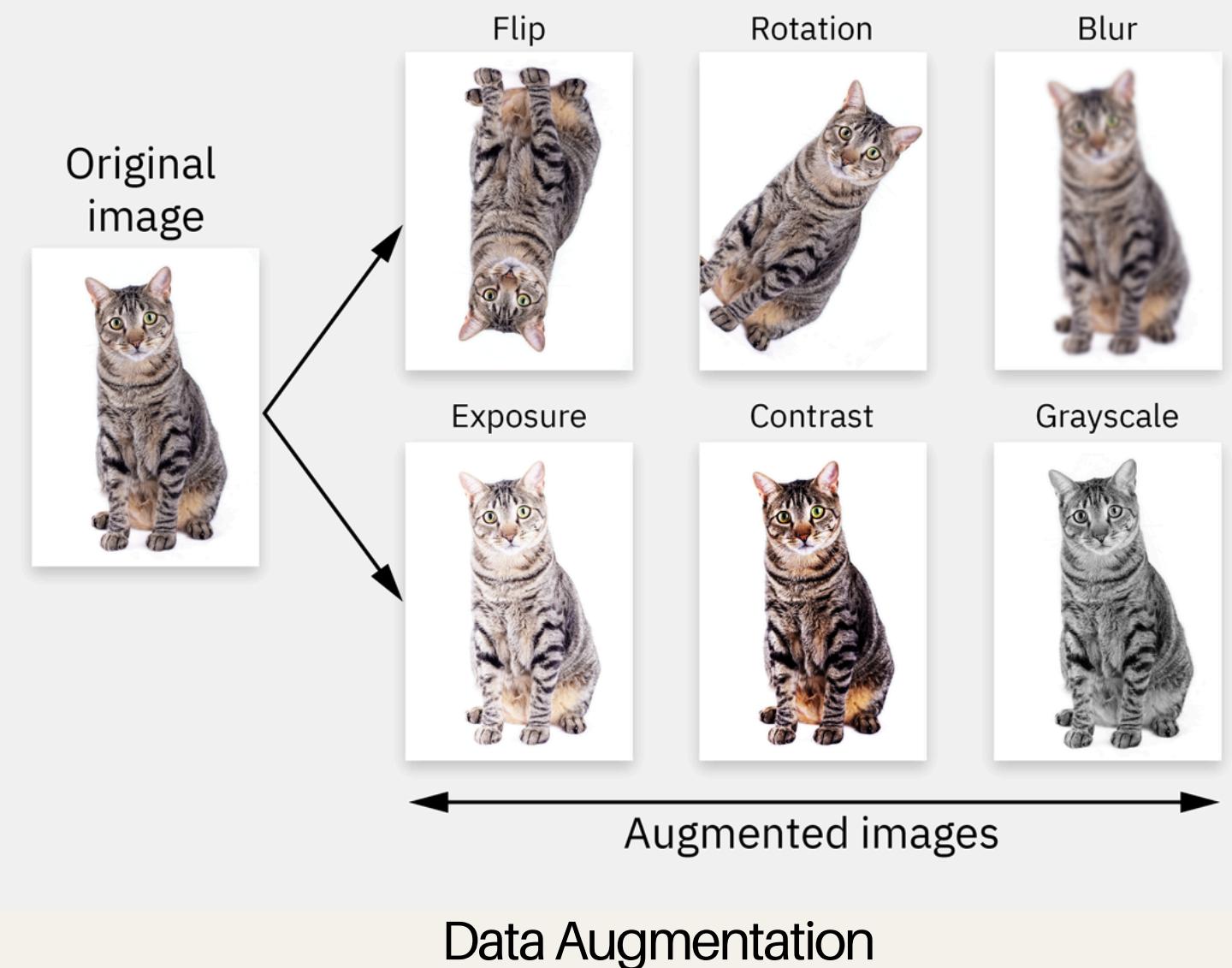
Dengan kemampuan menangkap konteks global dan hubungan jarak jauh, attention melengkapi keterbatasan konvolusi yang bersifat lokal.



Data Augmentation

Data augmentation digunakan untuk meningkatkan keragaman data pelatihan dan mengurangi overfitting. Teknik konvensional seperti flipping dan cropping dikombinasikan dengan RandAugment, yang secara acak menerapkan beberapa transformasi dengan intensitas tertentu.

Pendekatan ini terbukti efektif dalam meningkatkan generalisasi, terutama pada dataset kecil seperti Tiny-ImageNet.



Ide Improvement

Hybrid Self-Attention pada ConvNeXt

Menambahkan modul self-attention ringan pada tahap akhir ConvNeXt untuk memperkaya pemodelan konteks global tanpa mengubah backbone CNN.

Meningkatkan Robustness Model

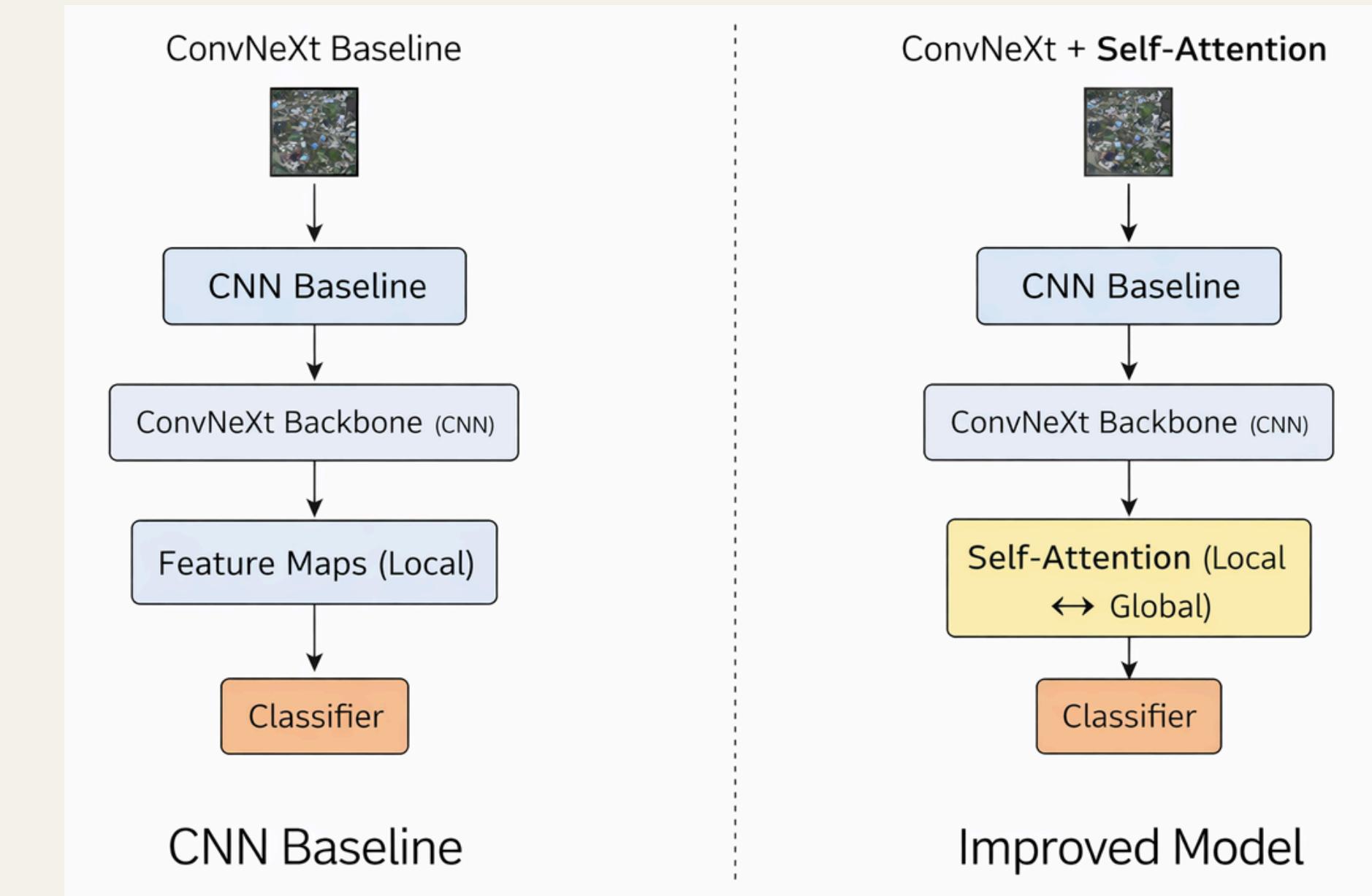
Menggunakan regularisasi seperti data augmentation, label smoothing, dan early stopping untuk mengurangi overfitting

Mengurangi Bias Data

Melakukan subset balancing dan sampling terkontrol pada Tiny-ImageNet untuk mengurangi dominasi kelas tertentu.

Efisiensi Adaptasi Model

Manfaatkan model pretrained ImageNet dengan fine-tuning terbatas sehingga tidak memerlukan pretraining skala besar dari nol.

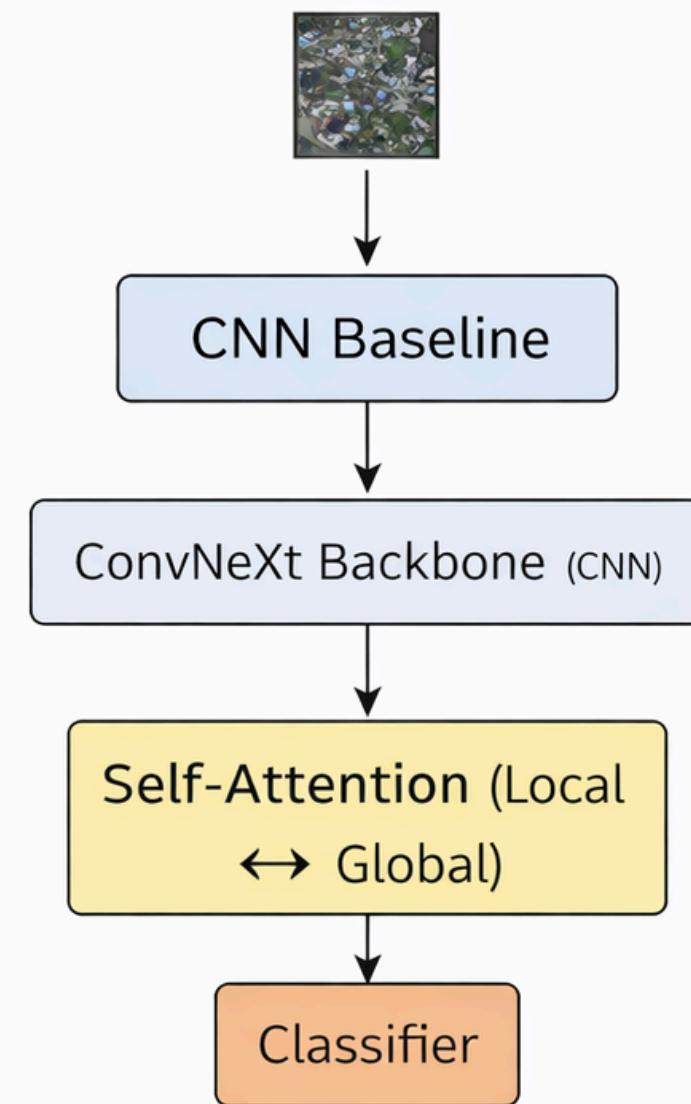


Integrasi self-attention pada tahap akhir ConvNeXt

Arsitektur yang Diusulkan

Model yang diusulkan menggunakan ConvNeXt-Tiny pretrained ImageNet-1K sebagai backbone untuk mengekstraksi fitur visual tingkat tinggi. Feature map diambil dari tahap akhir ConvNeXt, kemudian diproses melalui pooling dan diubah menjadi spatial tokens. Pada tahap ini, modul self-attention ringan ditambahkan untuk memodelkan relasi global antar area gambar. Fitur hasil attention selanjutnya di-aggregate dan diteruskan ke classification head untuk menghasilkan prediksi kelas akhir.

ConvNeXt + **Self-Attention**



Improved Model

Self-attention digunakan untuk memperkaya representasi global-lokal tanpa mengubah struktur utama ConvNeXt.

Strategi Training

1. Strategi baseline menggunakan backbone ConvNeXt-Tiny dengan *freezing backbone* dan hanya melatih classifier.
2. Strategi improved menggunakan fine-tuning penuh dengan Automatic Mixed Precision, RandAugment, dropout, weight decay tinggi, dan early stopping untuk mengontrol overfitting.

Rencana #1

Dataset yang digunakan:

Tiny-ImageNet

Alasan pemilihan dataset:

- Ukuran data lebih realistik untuk dikerjakan dibandingkan menggunakan ImageNet-1K.
- Turunan resmi dari ImageNet sehingga relevan dengan model pretrained.
- Bisa di-resize ke 224×224 agar sesuai pipeline ConvNeXt.
- Cocok untuk tugas klasifikasi multi-kelas (200 kelas).
- Banyak digunakan sebagai small-scale benchmark pengganti ImageNet.

Rencana #2

Model baseline:

ConvNeXt (pretrained ImageNet-1K)

Rencana Modifikasi/Improvement:

Penambahan Modul Cross-Attention

- Diletakkan pada stage akhir ConvNeXt
- Memberikan kemampuan menangkap global relation
- Perubahan arsitektur utama (kontribusi model)

2. Augmentasi Tambahan

- RandAugment / Cutout
- Mengurangi overfitting & meningkatkan generalisasi

3. Debiasing Dataset

- Reweight Sampling
- Menyeimbangkan kelas pada Tiny-ImageNet

Rencana #3

Eksperimen

1. Baseline

- Fine-tuning ConvNeXt-T pretrained pada TinyImageNet
- Freeze backbone + train classifier
- Evaluasi: top-1 accuracy, loss curve, confusion matrix

2. Improvement

- Implementasi cross-attention
- Tambah balancing sampling
- Training ulang & bandingkan dengan baseline

3. Analisis

- Perbandingan hasil sebelum & sesudah improvement pada dataset TinyImageNet

Timeline

Minggu	Rentang Tanggal	Kegiatan
Minggu 1	27 Nov – 3 Des	<ul style="list-style-type: none">- Setup environment & dataset Tiny-ImageNet- Menjalankan baseline ConvNeXt (pretrained)- Fine-tuning baseline- Mengumpulkan hasil awal (loss, akurasi, confusion matrix)- Menyusun dokumentasi baseline
Minggu 2	4 – 10 Des	<ul style="list-style-type: none">- Implementasi modul Cross-Attention- Menambahkan augmentasi (RandAugment/Cutout)- Debiasing dengan reweight sampling- Menjalankan training dengan improvement- Membandingkan hasil dengan baseline
Minggu 3	11 – 17 Des	<ul style="list-style-type: none">- Analisis akurasi & grafik final- Penyusunan paper- Membuat slide presentasi akhir- Revisi dan finalisasi keseluruhan proyek- Presentasi revisi

Eksperimen dan Hasil

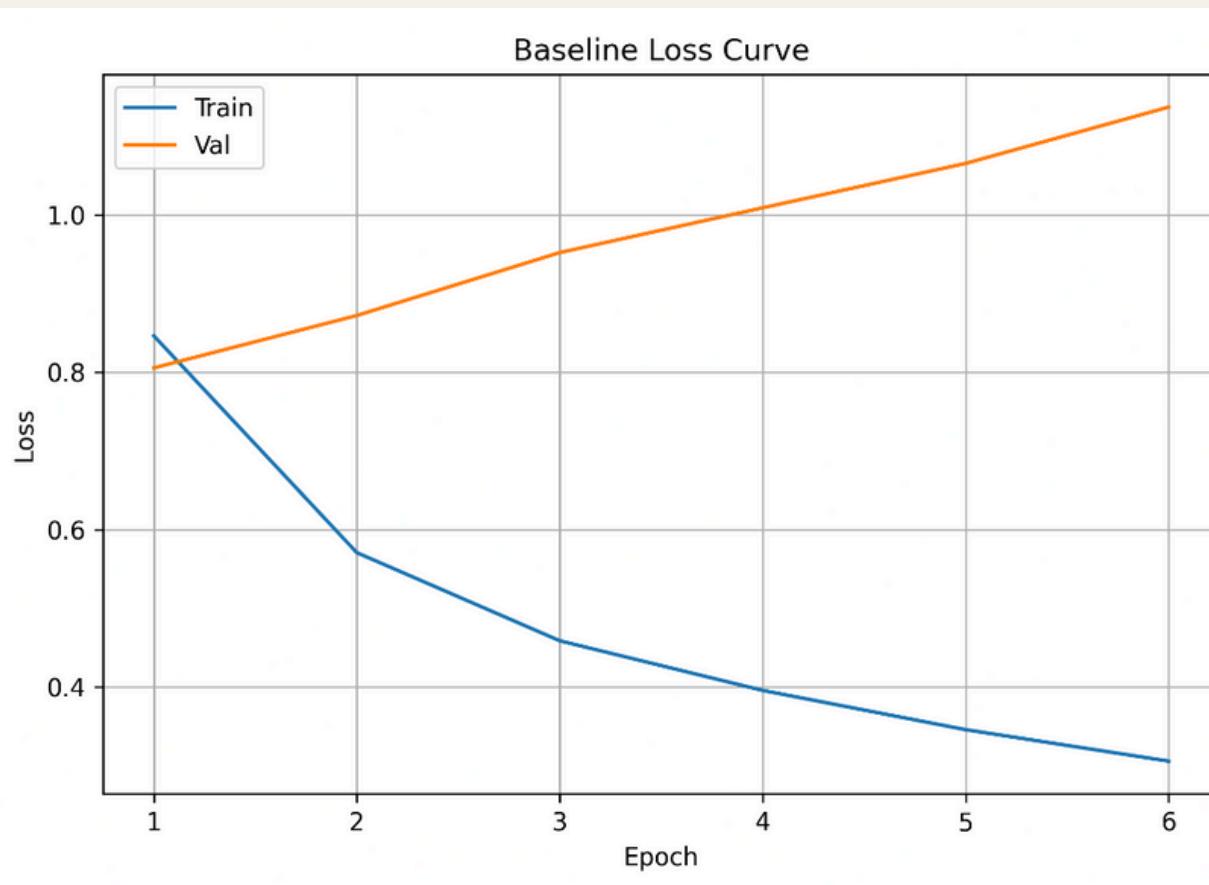
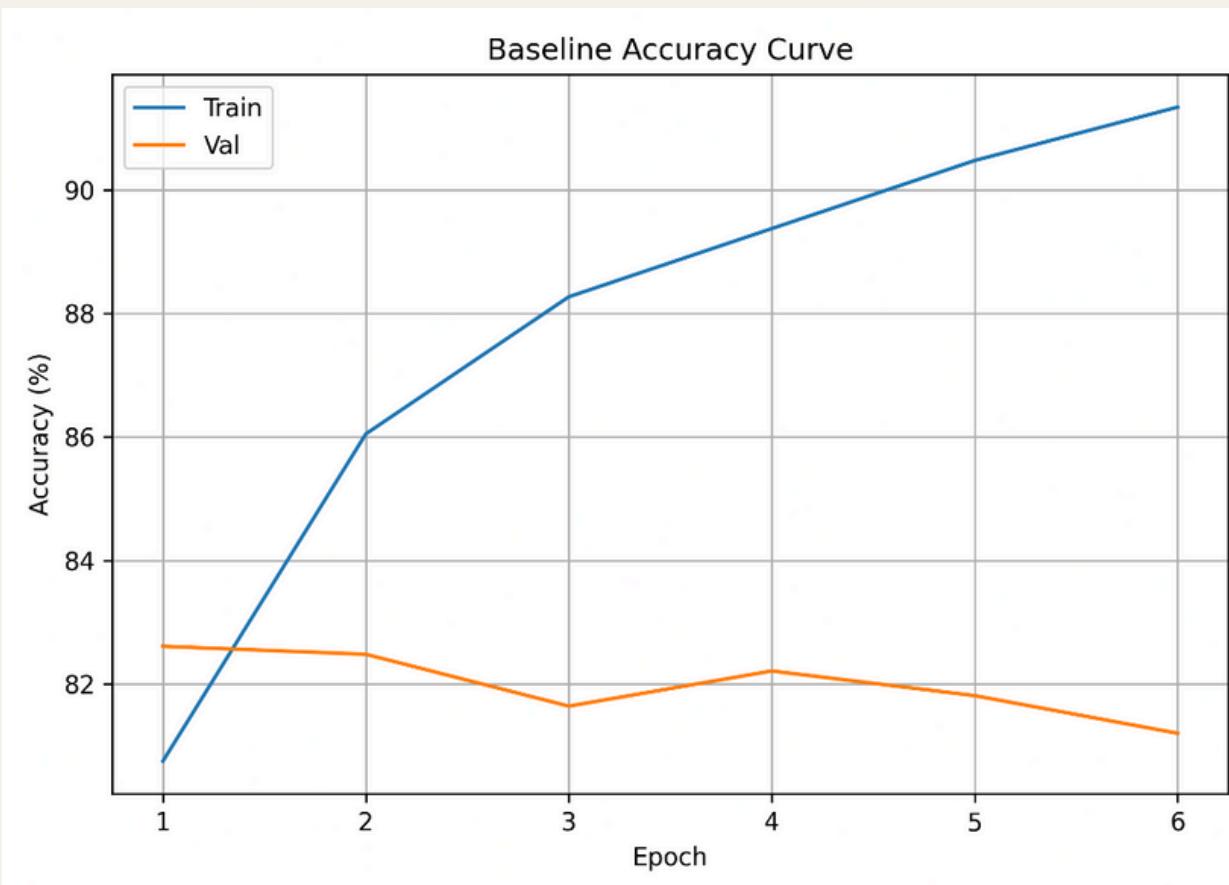
Model Scenario	Improvement	Training Acc	Validation Acc	Best Epoch at
Baseline	<i>no improvement</i>	80.75%	82.61%	1
Improved	Self-Attention, RandAugmentation, Dropout, Weight Decay	73.90%	86.30%	7

Eksperimen dijalankan dengan menggunakan GPU A100 dari Google Colab

Eksperimen dan Hasil (1)

Model Baseline

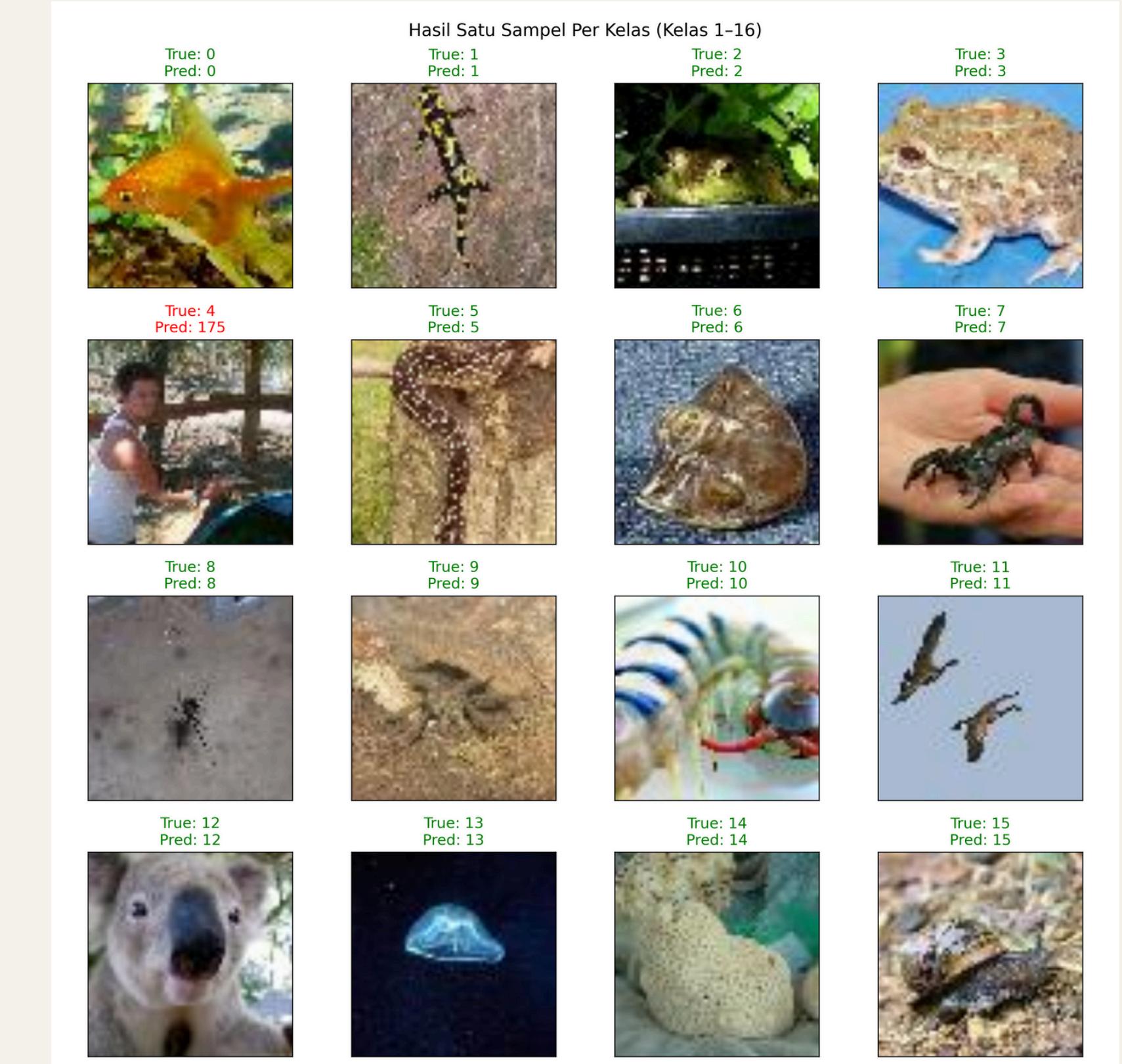
- Model baseline dijalankan dengan *freezing backbone* dan *early stopping*
- Hasil akurasi *training* dan *validation* terbaik di epoch 1
- Epoch 2 hingga 6 memicu early stop karena overfitting meningkat



```
...
Epoch 1: ....  
-> Train Loss: 0.8464 | Val Loss: 0.8057  
-> Train Acc: 80.75% | Val Acc: 82.61%  
-> Epoch Time: 1.99 menit  
  
Epoch 2: ....  
-> Train Loss: 0.5710 | Val Loss: 0.8723  
-> Train Acc: 86.05% | Val Acc: 82.48%  
-> Epoch Time: 1.76 menit  
-> EarlyStopping counter: 1 out of 5  
  
Epoch 3: ....  
-> Train Loss: 0.4590 | Val Loss: 0.9522  
-> Train Acc: 88.27% | Val Acc: 81.64%  
-> Epoch Time: 1.77 menit  
-> EarlyStopping counter: 2 out of 5  
  
Epoch 4: ....  
-> Train Loss: 0.3960 | Val Loss: 1.0091  
-> Train Acc: 89.38% | Val Acc: 82.21%  
-> Epoch Time: 1.78 menit  
-> EarlyStopping counter: 3 out of 5  
  
Epoch 5: ....  
-> Train Loss: 0.3461 | Val Loss: 1.0655  
-> Train Acc: 90.48% | Val Acc: 81.81%  
-> Epoch Time: 1.77 menit  
-> EarlyStopping counter: 4 out of 5  
  
Epoch 6: ....  
-> Train Loss: 0.3062 | Val Loss: 1.1370  
-> Train Acc: 91.34% | Val Acc: 81.20%  
-> Epoch Time: 1.76 menit  
-> EarlyStopping counter: 5 out of 5  
  
Early Stopping dipicu pada Epoch ke-6!  
  
Baseline Training Selesai!  
Total Waktu: 10.84 menit  
<All keys matched successfully>
```

Eksperimen dan Hasil (1)

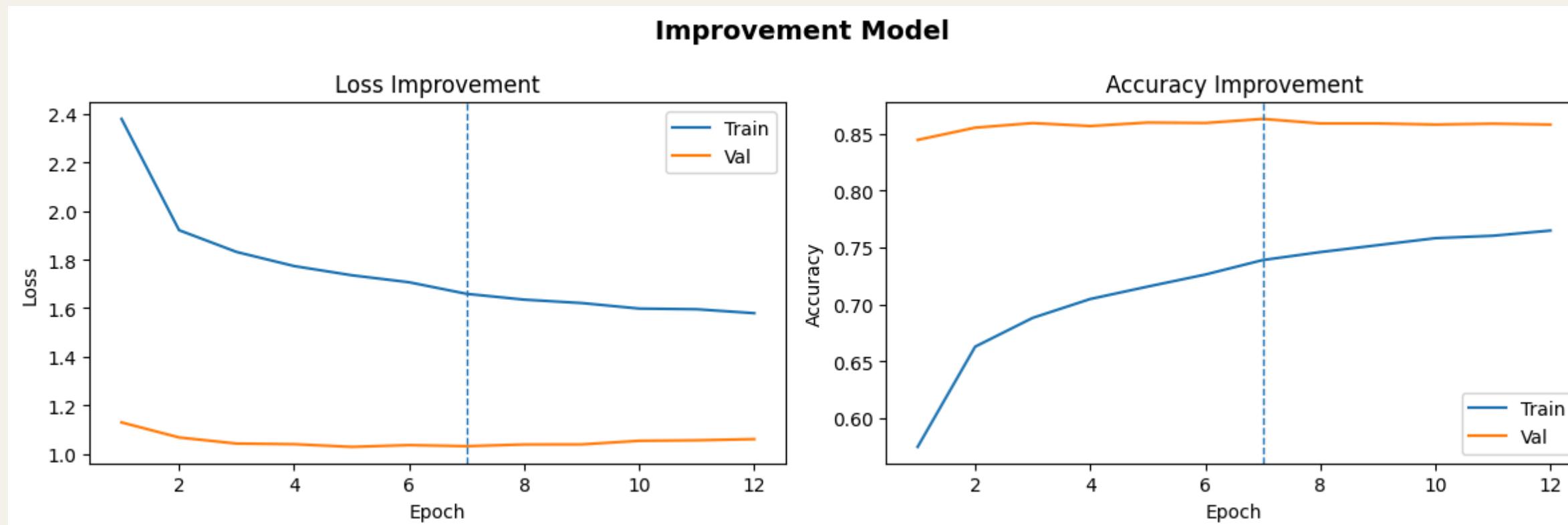
Prediksi Model Baseline pada 16 kelas pertama



Eksperimen dan Hasil (2)

Model Improved

- Model improved dijalankan dengan menambahkan Self-Attention, RandAugmentation, Dropout, dan Weight Decay
- Didapati epoch terbaik adalah epoch 7 dengan waktu *training* selama 52.17 menit dengan rata-rata per epoch adalah 4.35 menit.

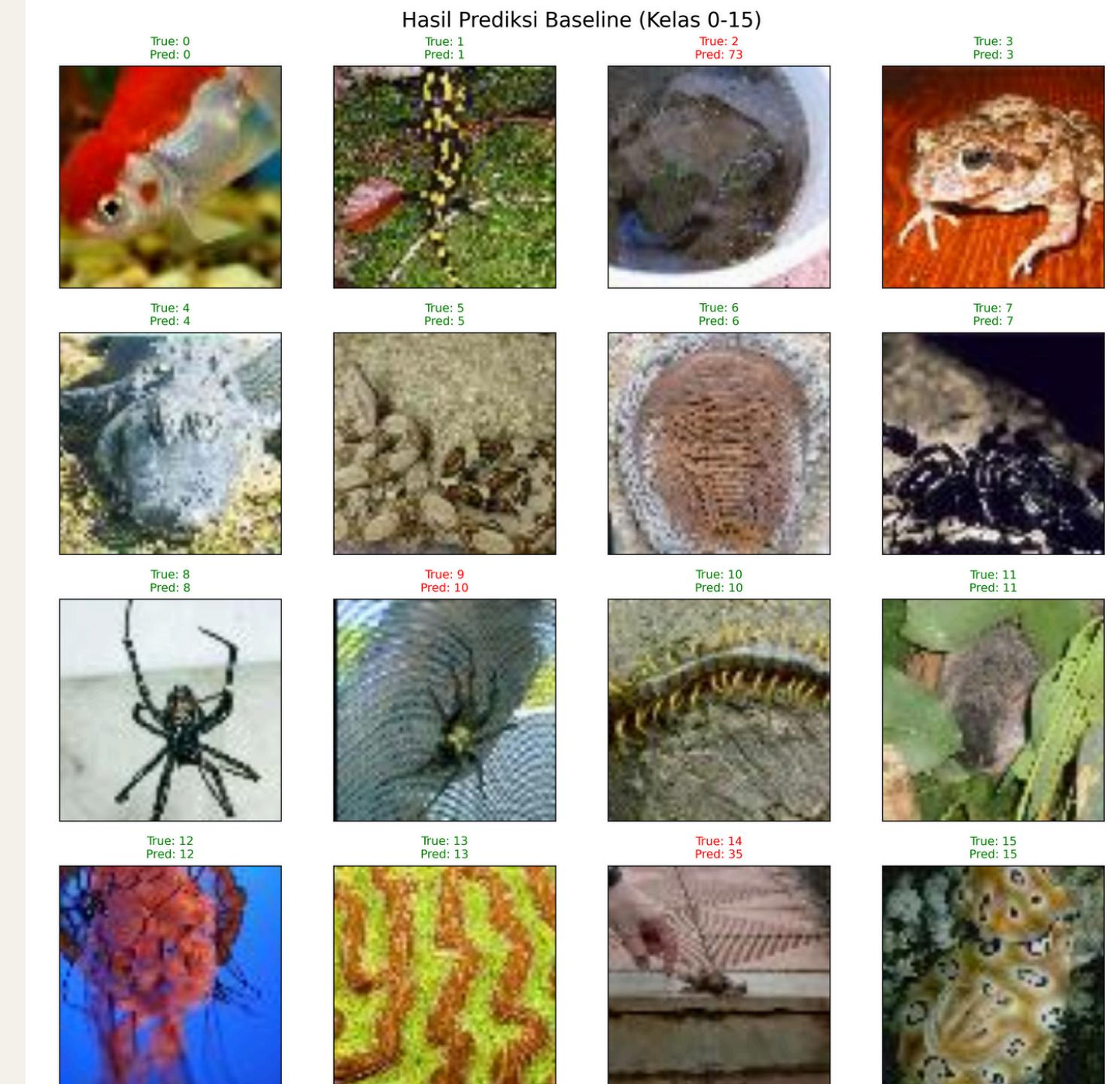


===== TRAINING SUMMARY =====

Best Val Accuracy :	86.30%
Best Epoch :	7
Total Epochs :	12
Total Time :	52.17 minutes
Avg Time / Epoch :	4.35 minutes

Eksperimen dan Hasil (2)

Prediksi Model Improved pada 16 kelas pertama



Analisis Hasil

Model Improved ConvNeXt mencapai akurasi validasi tertinggi sebesar 86,30% pada Epoch 7, meningkat sekitar 4% dibandingkan model baseline. Perbedaan dinamika training yang terlihat pada kurva akurasi dan loss menunjukkan bahwa model improved memiliki kemampuan generalisasi yang lebih baik. Akurasi validasi yang secara konsisten lebih tinggi dibandingkan akurasi training mengindikasikan efektivitas regularisasi agresif yang diterapkan selama proses pelatihan. Penggunaan RandAugment, dropout, dan weight decay tinggi mencegah model menghafal data pelatihan, sehingga mendorong pembelajaran fitur yang lebih robust. Selain itu, integrasi self-attention membantu model menangkap konteks global gambar, yang berdampak positif pada stabilitas pelatihan dan peningkatan performa klasifikasi pada Tiny-ImageNet.

Kesimpulan

Penelitian ini menunjukkan bahwa ConvNeXt-Tiny dapat diadaptasi secara efektif pada dataset kecil seperti Tiny-ImageNet dengan strategi pelatihan yang tepat. Integrasi self-attention ringan dan regularisasi agresif terbukti mampu meningkatkan generalisasi model secara signifikan.

Hasil ini menegaskan bahwa modernisasi CNN tidak hanya bergantung pada arsitektur, tetapi juga pada kombinasi desain pelatihan yang matang.

Model improvement berdampak langsung ke performa, walaupun meningkat tidak signifikan. Namun, sudah cukup untuk mengatasi masalah pada *baseline*, terlebih soal *overfitting*.

Thank you

CONVNEXT