

Ringkasan Buku 1

Nama Buku: Tensorflow in Actions

Team: Wilhelmina Arlene, Luthfiah Maulidya, Fadhilah Dwi Istiani

Chapter 6 — *Teaching Machines to See: Image Classification with CNNs*

Tujuan Utama Chapter 6

Chapter 6 adalah **awal Part 2 (Real-World Deep Networks)**.

Tujuan besarnya adalah:

Menerapkan CNN secara end-to-end untuk masalah nyata: klasifikasi gambar

Di chapter ini, kamu belajar:

- bagaimana **dataset image nyata** diorganisasi
- bagaimana membuat **pipeline data gambar**
- bagaimana membangun **CNN modern (Inception)**
- bagaimana **melatih & mengevaluasi model dengan benar**

Berbeda dari Chapter 4:

- di sini fokus **bukan konsep dasar**
- tapi **best practice industri**

STRUKTUR BESAR CHAPTER 6

Chapter 6 dibagi menjadi **4 tahap besar**:

1. Exploratory Data Analysis (EDA) pada data gambar
2. Data pipeline untuk image
3. Membangun CNN modern (Inception)
4. Training & evaluasi model

6.1 Exploratory Data Analysis (EDA) untuk Image Data **Kenapa EDA Penting untuk Image?**

Banyak orang langsung training CNN → **ini kesalahan besar**.

EDA membantu:

- memahami struktur dataset
- mendeteksi imbalance
- menemukan potensi bug sejak awal

6.1.1 Struktur Folder Dataset

Dataset image biasanya berbentuk:

dataset/

```
|— class_1/
|   |— img1.jpg
|   |— img2.jpg
|— class_2/
|   |— img3.jpg
|   |— img4.jpg
```

➔ **Folder = label**

Ini penting karena:

- Keras bisa membaca label otomatis
- pipeline jadi sederhana & konsisten

6.1.2 Memahami Kelas dalam Dataset

Langkah yang dilakukan:

- menghitung jumlah gambar per kelas
- memvisualisasikan contoh gambar
- memastikan label masuk akal

Masalah umum yang dicari:

- class imbalance
- gambar rusak
- resolusi tidak konsisten

6.1.3 Statistik Sederhana Dataset

Penulis menekankan:

- jangan remehkan statistik sederhana

Contoh:

- distribusi ukuran gambar
- rata-rata intensitas pixel
- jumlah total data

➔ Ini membantu menentukan:

- ukuran input CNN
- augmentasi yang dibutuhkan

6.2 Data Pipeline dengan ImageDataGenerator

Kenapa Pipeline Khusus Image?

Image:

- besar
- tidak muat memori
- butuh preprocessing

Solusi:

- **Keras ImageDataGenerator**

Fungsi ImageDataGenerator

ImageDataGenerator bisa:

- membaca gambar dari disk
- resize otomatis
- normalisasi pixel
- augmentasi data

Contoh augmentasi:

- rotation
- flipping
- zoom
- shift

Filosofi Augmentasi

Augmentasi:

- menambah variasi data
- **bukan menambah informasi**
- membantu generalisasi

➡ Model belajar **konsep**, bukan hafalan gambar.

6.3 Inception Net — CNN State-of-the-Art

Recap Singkat CNN

CNN klasik:

- convolution
- pooling
- dense

Masalah CNN klasik:

- pilih ukuran kernel itu sulit
- satu ukuran tidak cocok untuk semua fitur

Ide Revolusioner Inception

Inception memperkenalkan:

multi-scale feature extraction

Artinya:

- beberapa kernel size **secara paralel**
- lalu digabung

Contoh:

- 1×1 conv
- 3×3 conv
- 5×5 conv
- pooling

6.3.1 Inception v1

Struktur Inception module:

- jalur paralel
- output digabung (concatenate)

Manfaat:

- menangkap fitur kecil & besar
- efisien secara komputasi

1×1 Convolution (Bottleneck Layer)

Fungsi 1×1 conv:

- mengurangi dimensi channel
- menekan jumlah parameter
- mempercepat training

Ini adalah **trik arsitektural penting**.

Arsitektur Besar Model

Model Inception:

- terdiri dari banyak Inception block
- semakin dalam → semakin abstrak fitur

➡ Ini menunjukkan:

CNN modern ≠ sekadar stack conv layer.

6.4 Training & Evaluating the Model

6.4.1 Training Strategy

Langkah training:

1. compile model
2. tentukan:
 - loss
 - optimizer
 - metric
3. training dengan generator

Penulis menekankan:

- training CNN **butuh waktu**
- monitoring sangat penting

6.4.2 Evaluasi Model

Evaluasi dilakukan dengan:

- accuracy
- validation accuracy
- loss curve

Yang diperhatikan:

- gap train vs validation
- tanda overfitting

Insight Penting Evaluasi

Accuracy tinggi \neq model bagus.

Harus dicek:

- generalisasi
- stabilitas training
- performa pada data baru