

# **Laporan Analisis Klasifikasi Gambar Anjing & Kucing**

**Iqbal Maulana  
Teknik Informatika  
A1 | 442023611094**

## Deskripsi Proyek

Proyek ini bertujuan untuk mengembangkan dan melatih model kecerdasan buatan yang mampu mengklasifikasikan gambar digital menjadi dua kategori utama: kucing atau anjing. Kami memanfaatkan teknik deep learning dan konsep transfer learning untuk mencapai akurasi klasifikasi yang tinggi, bahkan dengan dataset yang mungkin terbatas

## Dataset

Dataset yang digunakan dalam proyek ini adalah kumpulan gambar digital yang terbagi menjadi dua kategori utama: kucing dan anjing. Struktur dataset dirancang untuk memfasilitasi pelatihan dan pengujian model klasifikasi, dengan gambar-gambar yang sudah dikelompokkan ke dalam direktori berdasarkan label kelasnya

### Struktur Direktori Dataset

```
cat-and-dog/
├── test_set/
│   └── test_set/
│       ├── cats/
│       │   └── (file gambar kucing, contoh: cat.4029.jpg)
│       └── dogs/
│           └── (file gambar anjing)
└── training_set/
    └── training_set/
        ├── cats/
        │   └── (file gambar kucing)
        └── dogs/
            └── (file gambar anjing)
```

### Pra-pemrosesan Data

Untuk mempersiapkan gambar-gambar ini agar dapat diproses oleh model deep learning, serangkaian transformasi diterapkan pada setiap gambar:

1. **Pengubahan Ukuran (Resize):** Setiap gambar diubah ukurannya menjadi dimensi **224x224 piksel**. Ukuran ini merupakan standar input yang diharapkan oleh sebagian besar model pre-trained seperti ResNet-18, memastikan konsistensi dimensi input ke jaringan saraf.
2. **Konversi ke Tensor (ToTensor):** Gambar-gambar dikonversi dari format PIL Image menjadi **tensor PyTorch**. Selama proses ini, nilai piksel yang awalnya dalam rentang  $[0, 255]$  dinormalisasi secara otomatis ke rentang  $[0.0, 1.0]$ .

3. **Normalisasi (Normalize):** Setelah konversi ke tensor, nilai-nilai piksel dinormalisasi lebih lanjut menggunakan nilai rata-rata (mean) dan standar deviasi (std) yang spesifik. Nilai-nilai ini (mean=[0.485,0.456,0.406] dan std=[0.229,0.224,0.225]) adalah nilai standar yang digunakan pada dataset ImageNet. Normalisasi ini penting karena model ResNet-18 dilatih dengan data yang dinormalisasi dengan cara yang sama, sehingga membantu model untuk memproses input baru secara efektif dan mempercepat konvergensi pelatihan.

## Arsitektur Model: ResNet-18 dengan Transfer Learning

Model yang digunakan dalam proyek ini adalah **ResNet-18**, sebuah arsitektur Jaringan Saraf Tiruan Konvolusional (Convolutional Neural Network/CNN) yang sangat populer dan efektif untuk tugas-tugas visi komputer, terutama klasifikasi gambar

### ResNet (Residual Network)

ResNet diperkenalkan untuk mengatasi masalah penurunan akurasi pada jaringan saraf yang sangat dalam, yang seringkali terjadi akibat masalah *vanishing/exploding gradients*. Konsep kunci dalam ResNet adalah koneksi residual (residual connections) atau skip connections.

### Implementasi dengan Transfer Learning

Dalam proyek ini, ResNet-18 tidak dilatih dari awal. Sebaliknya, kami memanfaatkan **transfer learning**:

1. **Model Awal yang Sudah Pintar (Pre-trained Model):** Kami mengambil model ResNet-18 yang sudah "pintar" karena sebelumnya sudah dilatih dengan sangat baik menggunakan dataset besar bernama **ImageNet**. Dataset ini berisi jutaan gambar dari berbagai kategori. Jadi, ResNet-18 ini sudah belajar mengenali berbagai fitur umum dalam gambar (seperti bentuk, warna, dan tekstur).
2. **Penyesuaian untuk Kucing & Anjing:** Model ResNet-18 asli dirancang untuk mengenali 1000 jenis objek. Karena kita hanya perlu membedakan antara **kucing dan anjing** (2 kategori), kami mengganti bagian terakhir model yang bertanggung jawab untuk klasifikasi. Bagian ini kami ganti dengan lapisan baru yang hanya menghasilkan **dua output** (satu untuk 'kucing' dan satu untuk 'anjing').

## Hasil Training

Proses pelatihan menunjukkan penurunan loss secara signifikan, menandakan bahwa model berhasil belajar.

Epoch 1, Loss: 0.2717131607870231  
Epoch 2, Loss: 0.16561905516984457  
Epoch 3, Loss: 0.16259640218604526  
Epoch 4, Loss: 0.11908817291817443  
Epoch 5, Loss: 0.11877209531141449

Visualisasi hasil:



## Evaluasi Model

model perlu diuji pada seluruh dataset pengujian yang terpisah untuk menghitung metrik

## Kesimpulan

Proyek ini berhasil membuat model yang bisa membedakan kucing dan anjing dari gambar. Dengan menggunakan model ResNet-18 dan teknik "belajar dari yang sudah pintar" (transfer learning), model menunjukkan hasil pelatihan yang baik karena nilai *loss* (kesalahan) terus menurun. Artinya, model berhasil belajar dengan baik dan kini bisa mengklasifikasikan gambar kucing atau anjing dengan tepat.