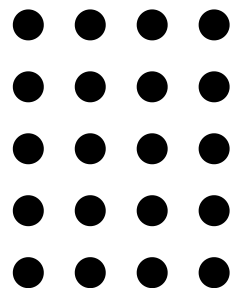


## IMPLEMENTASI JARINGAN SARAF TIRUAN



**Berliani Utami**

Npm : 2208107010082

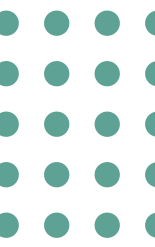
**TUGAS 2 - KECERDASAN  
ARTIFISIAL**

# Jenis Kasus

## Jenis Kasus : Klasifikasi Gambar

Model ini dirancang untuk mengklasifikasikan gambar binatang menjadi dua kelas yaitu, Kucing dan Anjing. Dimana model dilatih untuk mengidentifikasi apakah sebuah gambar termasuk dalam kategori Kucing (Cat) atau Anjing (Dog). Model ini menggunakan jaringan saraf tiruan jenis Convolutional Neural Network (CNN).

# Dataset yang Digunakan



## Nama Dataset

### 01. Kaggle Cats & Dogs Dataset

### 02. Deskripsi Dataset

Dataset ini merupakan kumpulan gambar binatang kucing dan anjing yang digunakan untuk tugas klasifikasi. Dataset ini awalnya dibuat untuk melatih mesin guna mendeteksi kucing dan anjing dalam sistem CAPTCHA. Dataset ini juga digunakan untuk meningkatkan keamanan dalam melindungi layanan web dari serangan brute-force dan mendeteksi spam berbasis gambar.

### 03. Jumlah Dataset

**Total Gambar :** 23.262 gambar (11.709 gambar kucing dan 11.553 gambar anjing).

**Sumber Dataset:** <https://datasets.activeloop.ai/docs/ml/datasets/kaggle-cats-dogs-dataset/#kaggle-cats-dogs-dataset>

# Jumlah Fitur

Setiap gambar diproses format 150 X 150 piksel dan memiliki channel warna (RGB).

- Jumlah Fitur per Gambar :  $150 \times 150 \times 30 = 67.500$  fitur

Ini adalah jumlah nilai piksel dalam sebuah gambar yang digunakan sebagai input untuk model.

# Jumlah Label

**Dataset memiliki dua label, yaitu :**



**0 : Kucing**

**1 : Anjing**

Karena ini adalah kasus klasifikasi biner, output model adalah probabilitas untuk masing - masing kelas, di mana nilai probabilitas mendekati 0 berarti kucing dan mendekati 1 berarti anjing.

# Jenis Jaringan Saraf Tiruan yang Digunakan



Model yang digunakan adalah Convolutional Neural Network (CNN).

CNN digunakan karena kemampuannya dalam memproses data berbentuk gambar dengan sangat baik melalui ekstraksi fitur spasial.

## Jenis Optimasi

### Adam Optimizer

Dipilih karena stabilitasnya dalam berbagai situasi, kemampuan menyesuaikan learning rate, dan efisiensi pada pelatihan model gambar.

## Jenis Fungsi Aktivasi yang Digunakan

### ReLu (Rectified Linear Unit)

Digunakan pada semua lapisan konvolusi dan hidden layer karena membantu mengatasi masalah vanishing gradient dan meningkatkan efisiensi pelatihan.

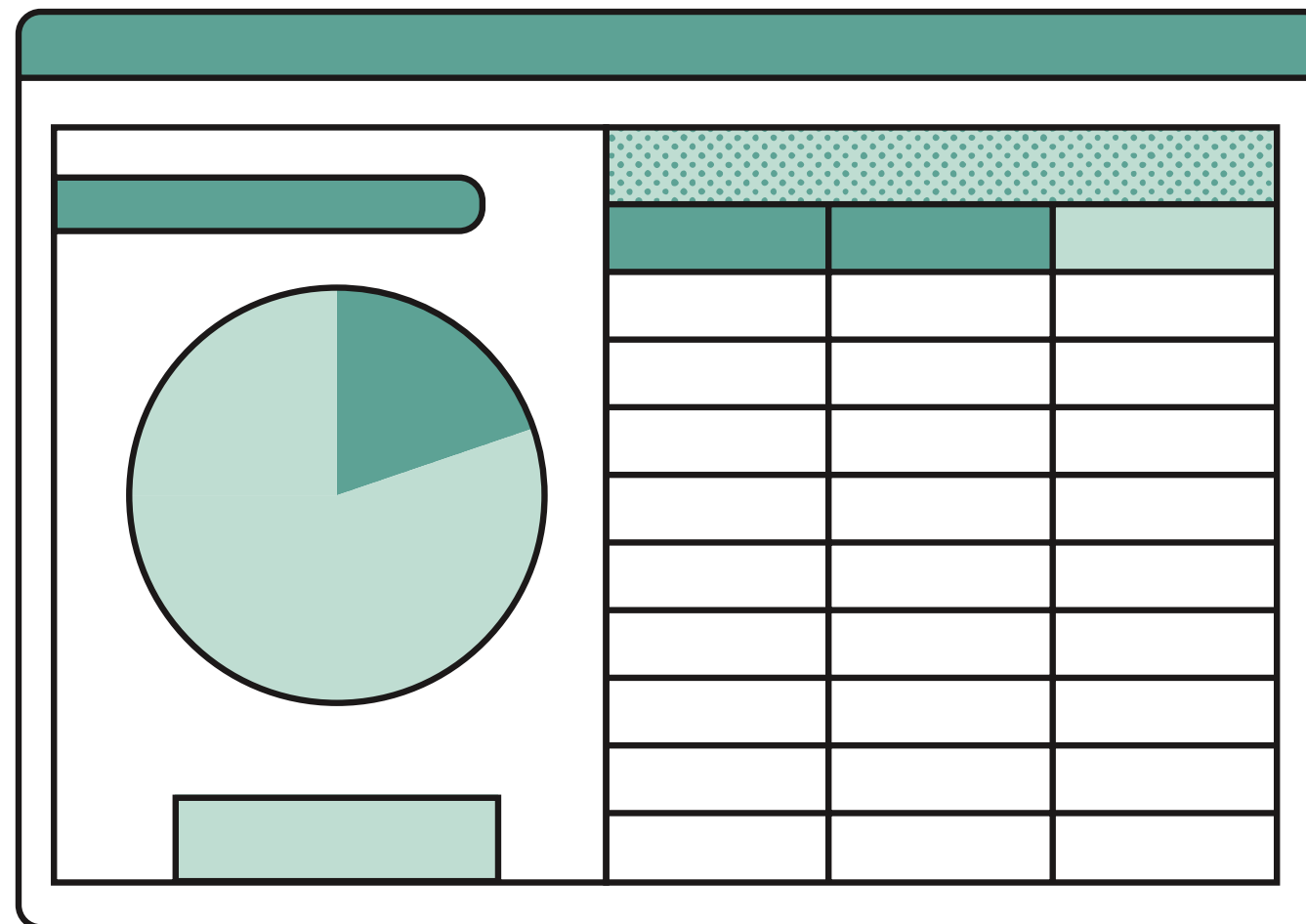
### Sigmoid

Digunakan pada output layer karena masalah ini adalah klasifikasi biner. Sigmoid mengeluarkan nilai antara 0 dan 1.

## Jumlah Hidden Layer

Model memiliki 3 hidden layer berupa lapisan konvolusi yang diikuti oleh pooling layer untuk ekstraksi fitur.

# Jumlah Total Hidden Node per Layer



- **Hidden Layer 1:** 32 filter (Conv2D)
- **Hidden Layer 2:** 64 filter (Conv2D)
- **Hidden Layer 3:** 128 filter (Conv2D)



# Jumlah Total Bobot (Weight)

**Jumlah bobot dihitung berdasarkan formula:**

Jumlah Bobot = (ukuran kernel  $\times$  input channel  $\times$  output channel) + bias

**Detail Bobot per Lapisan:**

**Hidden Layer 1:**

- Kernel:  $3 \times 3$ , Input Channel: 3, Output Channel: 32

Total Bobot =  $(3 \times 3 \times 3 \times 32) + 32 = 896$

**Hidden Layer 2:**

- Kernel:  $3 \times 3$ , Input Channel: 32, Output Channel: 64
- Total Bobot =  $(3 \times 3 \times 32 \times 64) + 64 = 18.496$

### Hidden Layer 3:

- Kernel:  $3 \times 3$ , Input Channel: 64, Output Channel: 128
- Total Bobot =  $(3 \times 3 \times 64 \times 128) + 128 = 73.856$

### Fully Connected Layer:

- Input Node: 18.432 (hasil dari layer sebelumnya), Output Node: 128
- Total Bobot =  $(18.432 \times 128) + 128 = 2.359.808$

### Fully Connected Layer:

Input Node: 128 Output Node: 1

Total Bobot =  $(128 \times 1) + 1 = 129$

### Jumlah Total Bobot (Weight):

$896 + 18.496 + 73.856 + 2.359.808 + 129 = 2.453.185$



Terima Kasih