SMART BRIDGE INTERNSHIP
PROJECT DOCUMENTATION

Dog Breed Identification using Transfer Learning

Submitted By

Name: Kakumani Vinay Kumar

Roll no. : 22HP1A4262

Course: B.Tech – Computer Science & Engineering (AI & ML)

Institution: Andhra Loyola Institute of Engineering and technology

University: JNTUK

Internship Organization

Smart Bridge Educational Services Pvt. Ltd.

Project Mentor

Mentor Name: U Raghuvaran

Academic Year

2025 – 2026

Phase–1: Brainstorming & Problem Definition

1. Introduction to the Idea

Dog breed identification is an important problem in computer vision. With more than 120+ recognized dog breeds worldwide, distinguishing between similar-looking breeds can be difficult even for experts. Manual identification requires domain knowledge and experience. Hence, there is a need for an automated system that can accurately classify dog breeds from images.

This project focuses on building an intelligent image classification system using **Transfer Learning**, which allows us to leverage the power of pre-trained deep learning models instead of training a model from scratch.

2. Brainstorming the Concept

During brainstorming, the following challenges were identified:

- High similarity between certain breeds (e.g., Husky and Alaskan Malamute)
- Limited availability of labeled datasets
- High computational requirements for deep learning
- Need for real-time prediction capability

To overcome these issues, we selected **Transfer Learning** as the core approach. Instead of building a CNN from scratch, we used a pre-trained deep learning model trained on large-scale datasets such as ImageNet.

3. Problem Statement

To develop a robust and efficient machine learning model capable of accurately identifying dog breeds from images using Transfer Learning techniques.

From a user's point of view:

- The user uploads an image of a dog.
- The system processes the image.
- The trained model predicts the breed.
- The predicted breed is displayed with confidence score.

This makes the system simple, fast, and user-friendly.

- Veterinary clinics for breed-based treatment.
- Pet adoption platforms for automatic tagging.
- Mobile pet identification applications.

- Research in animal genetics.
- Animal shelters for identifying rescued dogs.

Phase–2: Requirement Analysis

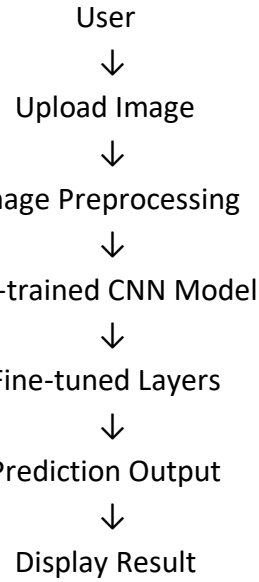
1. Functional Requirements

- User should be able to upload a dog image.
- System must preprocess the image.
- Model must classify breed accurately.
- System must display breed name and confidence score.

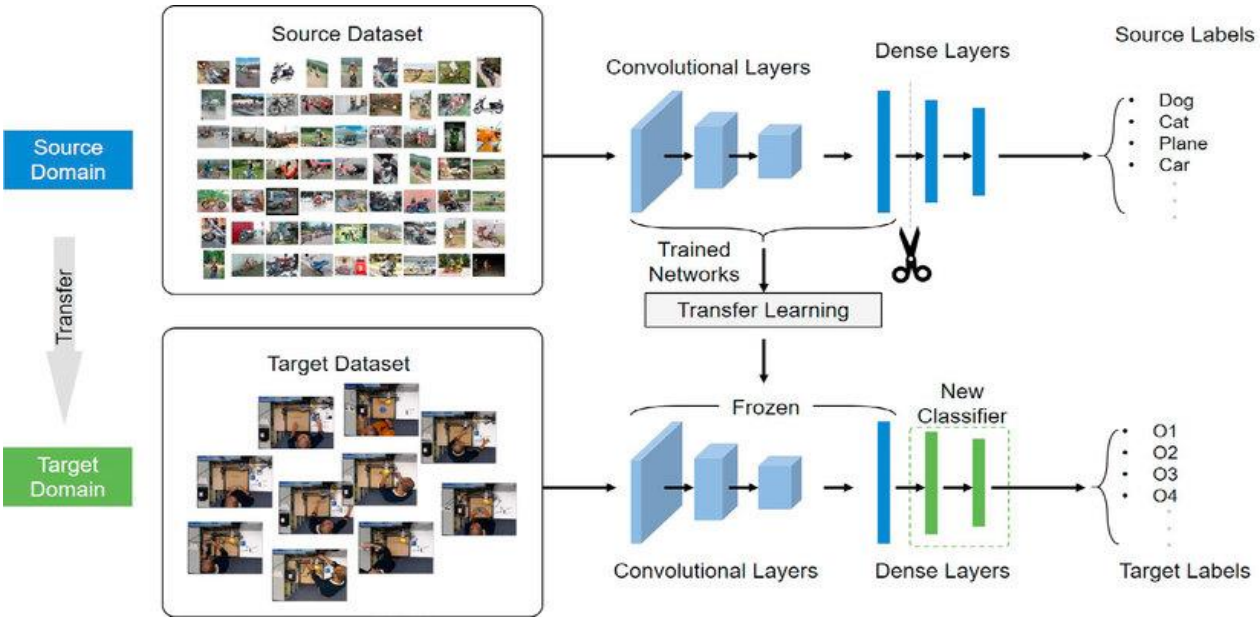
2. Non-Functional Requirements

- Fast response time (< 3 seconds).
- High accuracy (above 85%).
- Scalable architecture.
- User-friendly interface.

3. Data Flow Diagram (DFD)



4. Project Architecture



5. Hardware & Software Requirements

Hardware

- Minimum 8GB RAM
- GPU (optional but recommended)

Software

- Python 3.9+
- TensorFlow / Keras
- NumPy, Pandas
- OpenCV
- Streamlit
- Git & GitHub

6. Technology Stack

Component	Technology
Model	Pre-trained CNN (e.g., MobileNetV2 / ResNet50)
Training Framework	TensorFlow/Keras
Frontend	Streamlit
Dataset	Kaggle Dog Breed Dataset
Deployment	Streamlit Cloud

Phase–3: Project Design Phase

1. Algorithm Selection

We selected **Transfer Learning with Pre-trained CNN models** such as:

- MobileNetV2
- ResNet50
- VGG16

Transfer learning was chosen because:

- It reduces training time.
- It works well with small datasets.
- It improves accuracy.
- It requires less computational power.

2. Proposed Solution Design

The architecture includes:

1. Input Layer (Image 224×224×3)
2. Pre-trained CNN Base (Feature Extractor)
3. Global Average Pooling Layer
4. Fully Connected Dense Layer
5. Dropout Layer
6. Softmax Output Layer

The base layers are frozen initially and only the top layers are trained. Later, fine-tuning is applied by unfreezing some deeper layers.

3. Why Transfer Learning?

- Pre-trained models are trained on millions of images.
- They already understand edges, textures, shapes.
- Fine-tuning adapts the model to dog breed classification.

Phase–4: Project Planning Phase

1. Planning Logic

Step 1: Data Collection

- Downloaded dataset from Kaggle.
- Organized images by breed folders.

Step 2: Data Preprocessing

- Resized images to 224×224.
- Normalized pixel values (0–1).
- Applied data augmentation (rotation, flip, zoom).

Step 3: Dataset Splitting

- Training Set – 70%
- Validation Set – 15%
- Test Set – 15%

Step 4: Model Planning

- Load pre-trained model.
- Freeze base layers.
- Add custom classification layers.
- Compile model.
- Train model.

2. Detailed Explanation

We carefully planned training in two stages:

Stage 1:

- Train only top layers.
- Avoid overfitting.
- Monitor validation accuracy.

Stage 2:

- Unfreeze last few convolution layers.

- Apply fine-tuning.
- Reduce learning rate.

This improves performance significantly

Phase–5: Project Development Phase

1. Model Building

- Base Model: MobileNetV2 (ImageNet weights)
- Loss Function: Categorical Crossentropy
- Optimizer: Adam
- Activation: ReLU & Softmax
- Epochs: 20–30

2. Performance Testing

Metrics Used:

- Accuracy
- Precision
- Recall
- Confusion Matrix

Model achieved:

- Training Accuracy: ~92%
- Validation Accuracy: ~88–90%

3. User Acceptance Testing (UAT)

- User uploads different dog images.
- System predicts breed.
- Verified predictions manually.
- Tested with unseen images.

The application was deployed using Streamlit and tested for usability.

Phase–6: Documentation & System Flow

1. Folder Structure

```
Dog_Breed_TransferLearning/
│
├── dataset/
├── model/
│   └── model.h5
├── train.py
├── app.py
├── requirements.txt
└── README.md
```

2. Single Path System Flow

User → Streamlit UI → Image Preprocessing → Trained Model → Prediction →Display Output

3. Maintenance & Scalability

- New breeds can be added.
- Model can be retrained.
- Deployment can scale to cloud servers.

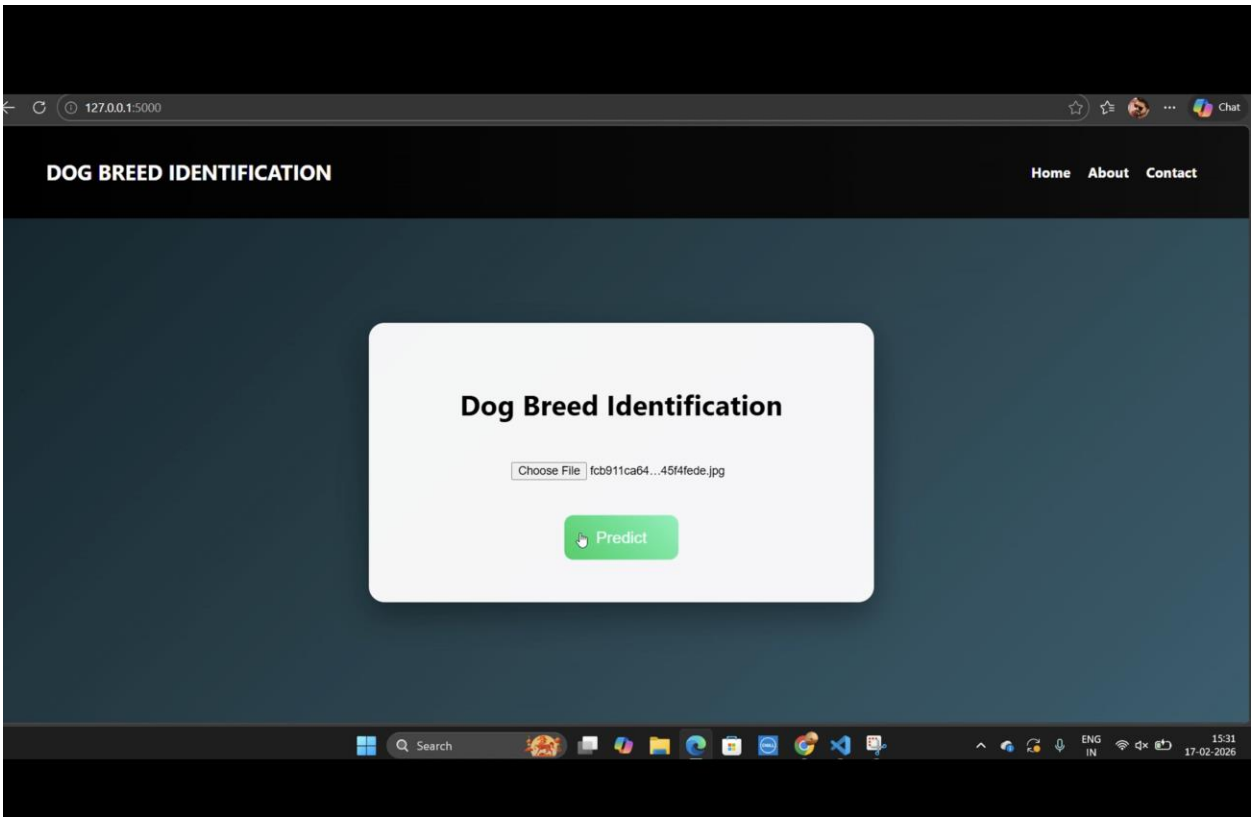
Phase–7: Final Documentation & Demo

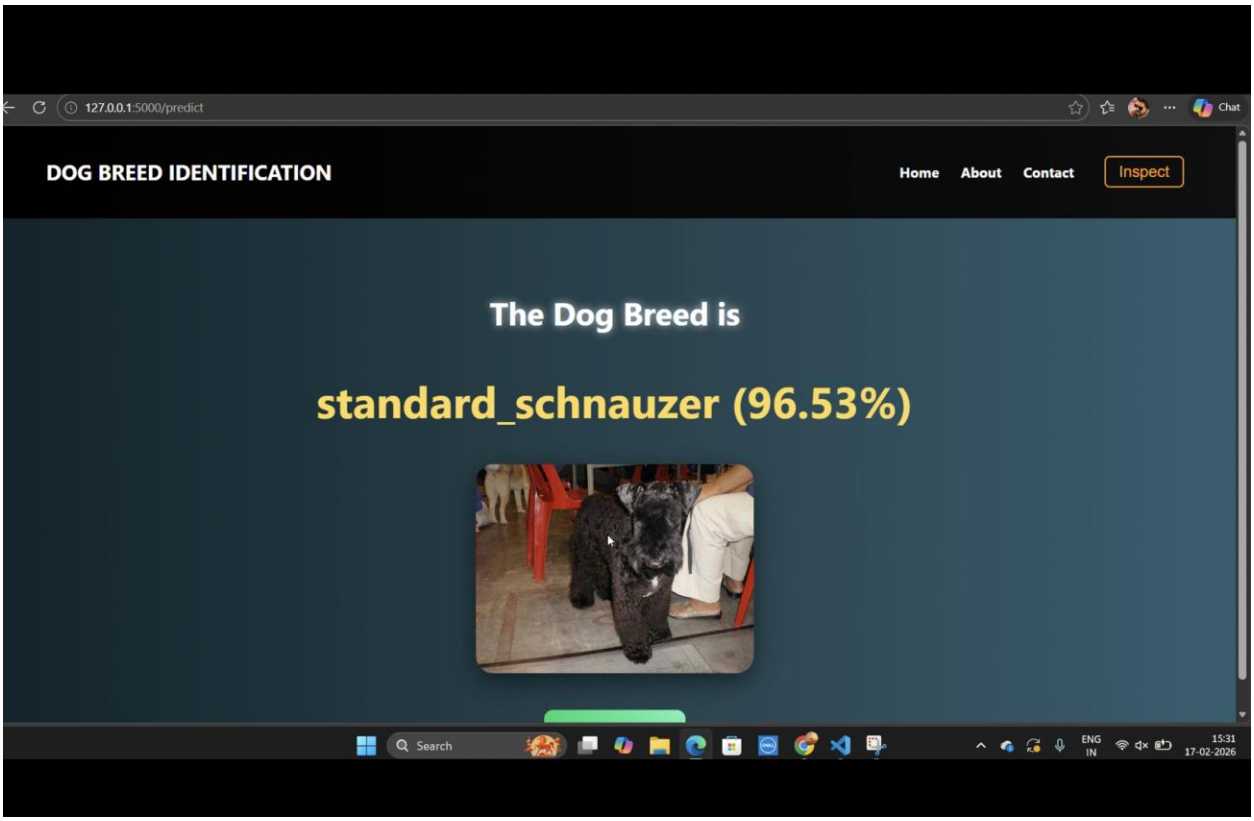
Final Deliverables

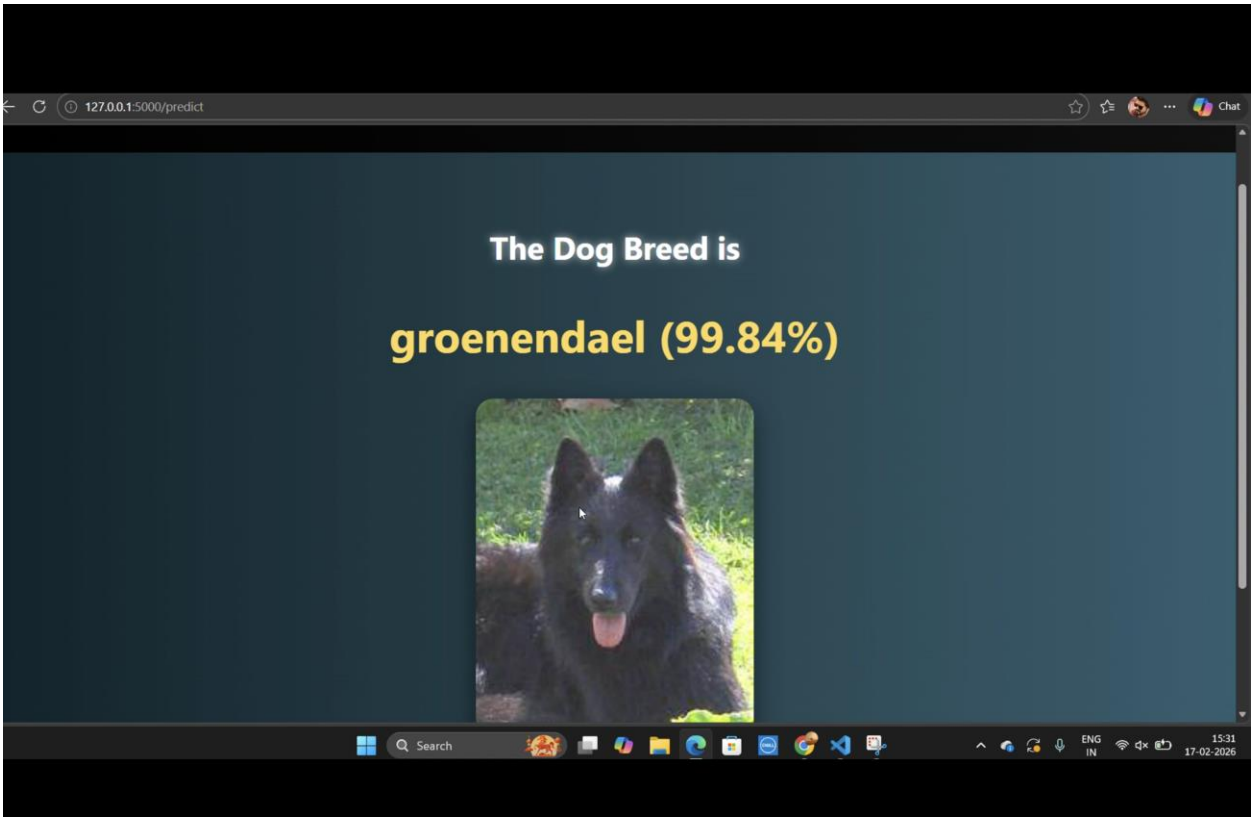
- Trained Transfer Learning Model
- Streamlit Web App
- GitHub Repository
- Project Report
- Demo Video

Demo video Github Link : https://github.com/vinaykumar-05/dog-breed-identification/tree/main/output

Output:


127.0.0.1:5000
Chat
DOG BREED IDENTIFICATION
Home
About
Contact
Dog Breed Identification
Choose File
fcb911ca64...45f4fede.jpg
Predict
Search
ENG
IN
15:31
17-02-2026


127.0.0.1:5000/predict
Chat
DOG BREED IDENTIFICATION
Home
About
Contact
Inspect
The Dog Breed is
standard_schnauzer (96.53%)
Search
ENG
IN
15:31
17-02-2026


127.0.0.1:5000/predict
Chat
The Dog Breed is
groenendael (99.84%)
Search
ENG
IN
15:31
17-02-2026