Dog Image Similarity Search using ResNet18 and Annoy

Rushikesh Musale

June 13, 2025

What I learn

- I learn some part of ResNet18.
- Learn how to develop Image similarity model.
- Also,How to use pytorch.

Libraries and Setup

We begin by importing all necessary Python libraries:

```
import os
import torch
from torchvision import models, transforms
from PIL import Image, ImageDraw
import torch.nn as nn
from annoy import AnnoyIndex
```

- `torch`, `torchvision`: For deep learning and pre-trained models.
- `PIL`: For image handling.
- `AnnoyIndex`: For fast approximate nearest neighbor searches.

Image and Model Preparation

We collect all image filenames and initialize the ResNet18 model:

```
image_folder = "PetImages/Dog"
image_files = os.listdir(image_folder)
image_files = [f for f in image_files if f.lower().endswith(('.png', '.
   jpg', '.jpeg'))]
```

We load a pre-trained ResNet18 model with ImageNet weights and modify it to use only its feature extractor:

```
weights = models.ResNet18_Weights.IMAGENET1K_V1
model = models.resnet18(weights=weights)
model.fc = nn.Identity()  # Remove classification layer
model.eval()
```

`model.fc = nn.Identity()` replaces the final classification layer with an identity layer so that we get 512-dimensional feature vectors instead of class predictions.

Preprocessing and Transformation

Before passing images to the model, they must be resized and normalized:

```
transform = transforms.Compose([
    transforms.Resize((224, 224)),
    transforms.ToTensor()
])
```

Annoy Index Setup

We initialize and load the prebuilt Annoy index:

```
annoy_index = AnnoyIndex(512, 'angular')
annoy_index.load('dog_index.ann')
```

`512` is the length of the ResNet18 output vector, and `'angular'` specifies the distance metric.

Visualization Grid Parameters

We define the layout of the final image grid:

```
grid_cols = 5
thumb_size = (200, 200)
grid_width = thumb_size[0] * grid_cols
grid_height = thumb_size[1] * ((24 + 1 + grid_cols - 1) // grid_cols)
```

Main Loop: Similarity Search and Visualization

For each image:

1. Extract feature using ResNet18.
2. Query 24 nearest images using Annoy.
3. Create a 5-column grid.
4. Highlight the query image with a red border.

```
for i, filename in enumerate(image_files):
    img_path = os.path.join(image_folder, filename)
    image = Image.open(img_path).convert('RGB')
    input_tensor = transform(image).unsqueeze(0)

    with torch.no_grad():
        if input_tensor.size(1) == 3:
            output_tensor = model(input_tensor)
            nns = annoy_index.get_nns_by_vector(output_tensor[0], 24)

            image_grid = Image.new('RGB', (grid_width, grid_height), '
               white')

            image_draw = ImageDraw.Draw(image)
            image_draw.rectangle([(0, 0), (thumb_size[0]-1, thumb_size
               [1]-1)],
                                 outline='red', width=8)

            image = image.resize(thumb_size)
            image_grid.paste(image, (0, 0))

            for j, idx in enumerate(nns):
                neighbor_image = Image.open(os.path.join(image_folder,
                   image_files[idx])).convert('RGB')
                neighbor_image = neighbor_image.resize(thumb_size)
                x = thumb_size[0] * ((j + 1) % grid_cols)
                y = thumb_size[1] * ((j + 1) // grid_cols)
                image_grid.paste(neighbor_image, (x, y))
```

Saving the Output

Before saving, we ensure the directory `ImageDump` exists, then we save the grid image.

```
os.makedirs('ImageDump', exist_ok=True)
image_grid.save(f'ImageDump/image_{i}.png')
```

Conclusion

This project demonstrates how to use deep learning features with approximate nearest neighbor search to find visually similar images. It is efficient, scalable, and useful for applications like content-based image retrieval.

Sample Output Image

Below is an example of the output grid showing the query image (with a red border) and its 24 most similar images based on ResNet18 features and Annoy similarity search.

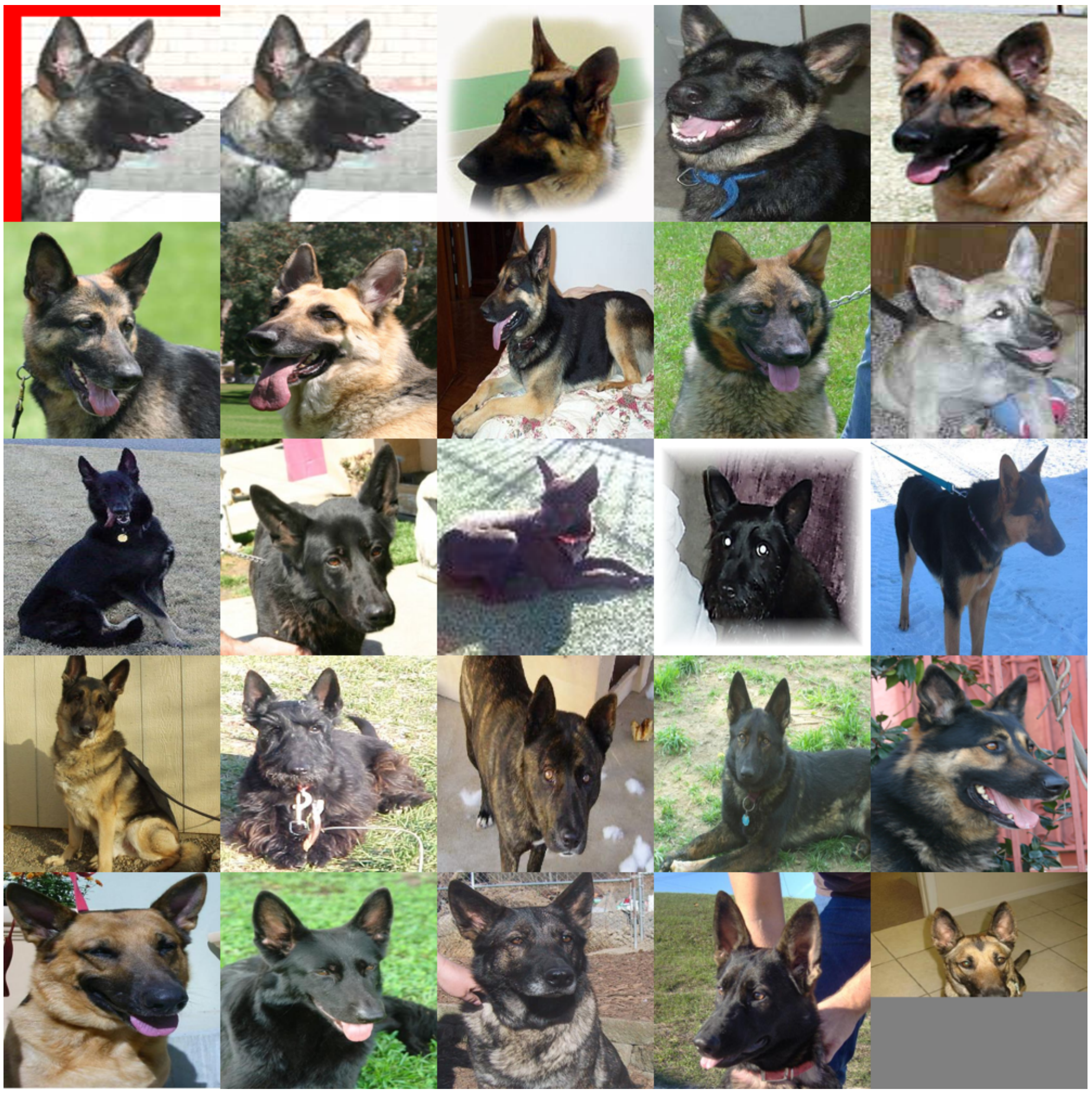

Figure 1: Query image (top-left) and 24 similar dog images.