**Preprocessing:**

import torch

import torchvision.transforms as transforms

from PIL import Image

import matplotlib.pyplot as plt

import os

DATASET\_DIR = '/content/dataset/animals/images'

# Define the transformations

transform = transforms.Compose([

transforms.Resize((64, 64)),

transforms.ToTensor(),

])

def load\_images\_from\_folder(folder, max\_images=30):

images = []

for filename in os.listdir(folder):

if filename.lower().endswith(('png', 'jpg', 'jpeg')):

img\_path = os.path.join(folder, filename)

img = Image.open(img\_path).convert('RGB') # Ensure RGB mode

images.append(img)

if len(images) >= max\_images:

break

return images

def visualize\_images(original, preprocessed, num\_images=5):

plt.figure(figsize=(12, 6))

for i in range(num\_images):

plt.subplot(2, num\_images, i + 1)

plt.imshow(original[i])

plt.title("Original")

plt.axis('off')

plt.subplot(2, num\_images, num\_images + i + 1)

plt.imshow(preprocessed[i])

plt.title("Preprocessed")

plt.axis('off')

plt.tight\_layout()

plt.show()

original\_images\_pil = load\_images\_from\_folder(DATASET\_DIR)

processed\_images = [transform(img) for img in original\_images\_pil]

num\_images\_to\_visualize = 5

visualize\_images(

original\_images\_pil,

[img.permute(1, 2, 0).numpy() for img in processed\_images],

num\_images\_to\_visualize

)