**Abstract:**

Image recognition refers to technologies that identify places, logos, people, objects, buildings, and several other variables in digital images. It may be very easy for humans like you and me to recognise different images, such as images of animals. We can easily recognise the image of a cat and differentiate it from an image of a horse. But it may not be so simple for a computer.

A digital image is an image composed of picture elements, also known as pixels, each with finite, discrete quantities of numeric representation for its intensity or grey level. So the computer sees an image as numerical values of these pixels and in order to recognise a certain image, it has to recognise the patterns and regularities in this numerical data.

How does Image recognition work?

Typically the task of image recognition involves the creation of a neural network that processes the individual pixels of an image. These networks are fed with as many pre-labelled images as we can, in order to “teach” them how to recognize similar images.

**Introduction:**

Using pertained model Resnet50 for Image recognition ResNet-50 is a convolutional neural network that is 50 layers deep. load a pretrained version of the network trained on more than a million images from the ImageNet database . The pretrained network can classify images into 1000 object categories, such as keyboard, mouse, pencil, and many animals.

Resnet50 architecture:

Diagram

Description automatically generated

**Dataset:**

ImageNet is a large dataset of annotated photographs intended for computer vision research.

The goal of developing the dataset was to provide a resource to promote the research and development of improved methods for computer vision.

Based on statistics about the dataset recorded on the ImageNet homepage, there are a little more than **14** million images in the dataset, a little more than **21** thousand groups or classes (synsets), and a little more than 1 million images that have bounding box annotations (e.g. boxes around identified objects in the images).

Here is how the data looks like:

A picture containing shop

Description automatically generated

**Pre-processing the full ImageNet dataset**

The ImageNet dataset consists of three parts, training data, validation data, and image labels.

The training data contains 1000 categories and 1.2 million images.

The validation and test data consists of 150,000 photographs, collected from Flickr and other search engines, hand labeled with the presence or absence of 1000 object categories. The 1000 object categories contain both internal nodes and leaf nodes of ImageNet, but do not overlap with each other. A random subset of 50,000 of the images with labels has been released as validation data along with a list of the 1000 categories. The remaining images are used for evaluation and have been released without labels.

**Code:**

**import tensorflow**

**import keras**

**from keras.applications.resnet50 import ResNet50**

**from keras.preprocessing import image**

**from keras.applications.resnet50 import preprocess\_input, decode\_predictions**

**import numpy as np**

**model = ResNet50(weights='imagenet')**

**img\_path = '44.png'**

**import matplotlib.pyplot as plt**

**%matplotlib inline**

**from skimage.io import imread**

**img = imread(img\_path)**

**plt.imshow(img)**

**img = image.load\_img(img\_path, target\_size=(224, 224))**

**x = image.img\_to\_array(img)**

**x = np.expand\_dims(x, axis=0)**

**x = preprocess\_input(x)**

**preds = model.predict(x)**

**print('Predicted:', decode\_predictions(preds, top=3)[0])**

Graphical user interface, text, application, email

Description automatically generated**Test:**



Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated



**References:**

Deep Residual Learning for Image Recognition

CVPR 2016 · Kaiming He, Xiangyu Zhang, Shaoqing Ren, Jian Sun ·

https://keras.io/api/applications/resnet