AUTOMATIC IMAGE TAGGING USING MACHINE LEARNING

Dissertation submitted to Shri Ramdeobaba College of Engineering and Management, Nagpur in partial fulfilment of requirement for the award of degree of

Bachelor of Engineering

in

Computer Science and Engineering

BY

ABHILASH MANDLEKAR[BE13CSU020]

CHINMAY DEGWEKAR [BE13CSU033]

NISHANT KASHIV [BE13CSU043]

PRATIK DEVIKAR [BE13CSU047]

SAARTHAK PANDE [BE13CSU055]

UNDER THE SUPERVISION OF

PROF. V. RATHOD

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SHRI RAMDEOBABA COLLEGE OF ENGINEERING AND MANAGEMENT, NAGPUR-13

(An Autonomous Institution of Rashtrasant Tukadoji Maharaj Nagpur University)

NOVEMBER-2016

SHRI RAMDEOBABA COLLEGE OF ENGINEERING AND MANAGEMENT

(An autonomous institute affiliated to Rashtrasant Tukdoji Maharaj Nagpur University Nagpur)

Department of Computer Science and Engineering

CERTIFICATE

This is to certify that the Thesis on "**AUTOMATIC IMAGE TAGGING USING MACHINE LEARNING**" is a bonafide work of

ABHILASH MANDLEKAR
CHINMAY DEGWEKAR
NISHANT KASHIV
PRATIK DEVIKAR
SAARTHAK PANDE

submitted to the to Rashtrasant Tukdoji Maharaj Nagpur University, Nagpur in fulfilment of the award of a Degree of Bachelor of Engineering. It has been carried out at the Department of Computer Science and Engineering , Shri Ramdeobaba College of Engineering and Management, Nagpur during the academic year 2016-17.

Date: 11/11/2016

Place: NAGPUR

Prof. V. Rathod
Project Guide

Dr. M.B.Chandak
H.O.D
Department of Computer Science and Engineering

Dr. R.S.Pande
Principle

DECLARATION

I hereby declare that the thesis titled "**AUTOMATIC IMAGE TAGGING USING MACHINE LEARNING**" submitted herein has been carried out in the Department of Computer Science and Engineering of Shri Ramdeobaba College and Management , Nagpur. The work is original and has not been submitted earlier as a whole or part for the award of any degree/diploma at this or any other institution/ University.

Date: 11/11/2016

Place: Nagpur

Name of the Student	Roll No	Signature
Abhilash Mandlekar	41	
Chinmay Degwekar	54	
Nishant Kashiv	65	
Pratik Devikar	69	
Saarthak Pande	77	

APPROVAL SHEET

This report entitled "AUTOMATIC IMAGE TAGGING USING MACHINE LEARNING" by

ABHILASH MANDLEKAR
CHINMAY DEGWEKAR
NISHANT KASHIV
PRATIK DEVIKAR
SAARTHAK PANDE

is approved for the degree of Bachelor of Engineering.

Name and Signature of the Supervisor

Name and Signature of External Examiner

Name and Signature of RRC Members

Name and Signature of HOD

Date: 11/11/2016

Place: NAGPUR

ACKNOWLEDGEMENTS

The success and final outcome of this project required a lot of guidance and assistance from many people and we are extremely fortunate to have got this all along the completion of our project work. Whatever we have done is only due to such guidance and assistance and we will not forget to thank them.

We respect and thank Dr. M.B.Chandak , for giving us an opportunity to do the project work and providing us all support and guidance which made us complete the project on time . We are extremely grateful to him for providing such a nice support and guidance though he had busy schedule managing the college affairs.

We owe our profound gratitude to our project guide Prof. V. Rathod who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system.

We are thankful to and fortunate enough to get constant encouragement, support and guidance from all Teaching staffs of Department of computer science which helped us in successfully completing our project work. Also, we would like to extend our sincere regards to all the non-teaching staff of department of computer science for their timely support.

NAME OF THE PROJECTEES-

- PRATIK DEVIKAR
-ABHILASH MANDLEKAR
-CHINMAY DEGWEKAR
-NISHANT KASHIV
-SAARTHAK PANDE

ABSTRACT

At present, Image Recognition is one of the most important research areas in the field of Artificial Intelligence. Mobile phones are becoming the convergent platform for personal sensing, computing, and communication. This project attempts to exploit this convergence towards the problem of automatic image tagging. We envision our Android Application, a mobile phone based collaborative system that senses the people, activity, and context in a picture, and merges them carefully to create tags on-the-fly. We deploy a prototype of our application on Android phones, and demonstrate its effectiveness through various pictures, taken in various settings. While research in face recognition continues to improve image tagging, our application is an attempt to embrace additional dimensions of sensing towards this end goal. Performance comparison with Tensorflow's Android Application shows that such an out-of-band approach is valuable, especially with increasing device density and greater sophistication in sensing/learning algorithms. The application accomplishes this feat using a bundled machine learning model running which is located on a server. The model is trained against thousands of images so that it can look at the photos the camera feeds it and classify the object into its best guess (from the 24 object classifications it knows). Along with its best guess, it shows a confidence score to indicate how sure it is about its guess. The ultimate goal of this project is to help user identify the appropriate tags or annotations of the uploaded images.

Keywords - Machine Learning, Android App, Tagging, Annotations, Images, Recognition, Classification.

List Of Figures

TABLE OF CONTENTS:

1. INTRODUCTION

With continuously increasing amounts of images available on the Web and elsewhere, it is important to find methods to annotate and organize image databases in meaningful ways. Tagging images with words describing their content can contribute to faster and more effective image search and classification. In fact, a large number of applications, including the image search feature of current search engines (e.g.,Yahoo!, Google) or the various sites providing picture storage services (e.g., Flickr, Picasa) rely exclusively on the tags associated with an image in order to search for relevant images for a given query. However, the task of developing accurate and robust automatic image annotation models entails daunting challenges. First, the availability of large and correctly annotated image databases is crucial for the training and testing of new annotation models. Although a number of image databases have emerged to serve as evaluation benchmarks for different applications, including image annotation (Duygulu et al., 2002), content-based image retrieval(Li and Wang, 2008) and cross language information retrieval (Grubinger et al.,2006), such databases are almost exclusively created by manual labeling of keywords, requiring significant human effort and time. The content of these image databases is often restricted only to a few domains, such as medical and natural photo scenes (Grubinger et al., 2006), and specific objects like cars, airplanes, or buildings (Fergus etal., 2003). For obvious practical reasons, it is important to develop models trained and evaluated on more realistic and diverse image collections. This project attempts to exploit this convergence towards the problem of automatic image tagging. We envision our Android Application, a mobile phone based collaborative system that senses the people, activity, and context in a picture, and merges them carefully to create tags on-the-fly. If done well, automatic image tagging can enable a variety of applications. One may imagine improved image search in the Internet, or even within one's own computer – Bob may query his personal photo collection for all pictures of Alice and Eve playing together in the snow.

Another application may tag videos with important event/activity markers; a user of this application may be able to move the video slider to the exact time-point where President Obama actually walks up to the podium, or starts speaking. Today, such functionalities may be available in select images and videos, where some humans have painstakingly tagged them [2]. TagSense aims to automate this process via sensor-assisted tagging.
Given the immense practical applications of a means of automatic image tagging, along with the deep academic challenges associated with recognising real world objects within images, it is not surprising to find that there has been great interest amongst the computer vision and information retrieval community in the development of robust and efficient automatic image tagging systems. The main purpose of tagging images in this manner is to allow for the retrieval of images based on natural language keywords as opposed to alternative content based image retrieval (CBIR) techniques such as query by sketch or query by example.Automatic image annotation technology will be at the forefront of this revolution in enabling users to use familiar natural language search interfaces to retrieve images of relevance.

1.1 *Objectives:*

- To create an android application to display the automatically extracted tags from the images related to everyday household objects.
- Automatic Image Annotation is a process by computer system which automatically assigns metadata in the form of keywords to a digital image.
- The automatic annotations of objects is done using Machine Learning and Deep Learning techniques, especially 'Convolutional Neural Networks'.
- This application of computer technique can be employed in image retrieval systems to organize and locate images of interest from a database and also in image categorization.

- This application can also be useful in Robotics and Artificial Intelligence.

2. REVIEW OF LITERATURE

Automatic image classification and recognition is the research focus of the pattern recognition method in the field of image processing. Commonly used image features are color, shape and texture. With the advent of time, the number of images being captured and shared online has grown exponentially. The images which are captured are later accessed for the purpose of searching, classification and retrieval operation. Hence these images must be labeled with appropriate words, phrases or keywords so that the requisite operation can be performed efficiently. Automatic Image Tagging is such a technique which associates an appropriate keyword from a given set of words or phrases based on the relevance to the content of the image. Web images consist of valuable contextual information located in nearby region or within the image itself. This contextual information is nothing but the information related to or in context of image that can be used for indexing the images and also in image retrieval system. The current existing system involves human interference and is time consuming besides inconsistency. The current existing system also deals with the problem of detecting overlapped contextual information accurately which leads to incorrect assigning of contextual information and also decreases the accuracy and efficiency of the system. All this drawbacks of existing system are removed in proposed system by automatic extraction of contextual information and by text processing. With this rapid growth, arises the need to perform effective manipulation (like searching, retrieval etc...) on images. Several search engines

retrieve relevant images by text-based searching without using any content information. However, recent research shows that there is a semantic gap between content based image retrieval and image semantics understandable by humans. Semantic gap can be described as “the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation”. As a result, research in this area has shifted to bridge the semantic gap between low level image features and high level semantics. Thus, assigning relevant keywords is significant and can improve the image search quality. This is known as Image Annotation. It is defined as technique of assigning semantically relevant keywords to an image. When images are retrieved using these annotations, such retrieval is known as annotation based image retrieval. Annotation-Based Image Retrieval systems are an attempt to incorporate more efficient semantic content into both text-based queries and image captions. As can be seen in many of today's image retrieval systems, ABIR is considered more practical.

3. THE BODY OF THESIS

This section discusses in detail all the components involved in the application. It also explains the basic functioning of the application. System architecture talks about the technologies and platform used in the application.

3.1 Working

An overview of how the application works described in the following steps:

- ❏ Home page will be displayed with a welcome message.
- ❏ Three options will be displayed for users:-
    1. Choose an image from Gallery
    2. Upload an image via Camera
    3. Send the image to the Server for processing.
- ❏ Once an image is selected, it will be sent to the server for further processing, where the trained machine learning model is kept.
- ❏ The model will analyze the image and give appropriate annotations as output.

- ❑ The result will be displayed on the android app.

3.2 Technologies Used

★ Artificial Neural Networks:

Neural Networks (also referred to as **connectionist systems**) are a computational approach which is based on a large collection of neural units loosely modeling the way the brain solves problems with large clusters of biological neurons connected by axons. Each neural unit is connected with many others, and links can be enforcing or inhibitory in their effect on the activation state of connected neural units. Each individual neural unit may have a summation function which combines the values of all its inputs together. There may be a threshold function or limiting function on each connection and on the unit itself such that it must surpass it before it can propagate to other neurons. These systems are self-learning and trained rather than explicitly programmed and excel in areas where the solution or feature detection is difficult to express in a traditional computer program.

★ Convolutional Neural Network

In machine learning, a **convolutional neural network** (**CNN**, or **ConvNet**) is a type of feed-forward artificial neural network in which the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex. Individual cortical neurons respond to stimuli in a restricted region of space known as the receptive field. The receptive fields of different neurons partially overlap such that they tile the visual field. The response of an individual neuron to stimuli within its receptive field can be approximated mathematically by a convolution operation. Convolutional networks were inspired by biological processes and are variations of multilayer perceptrons designed to use minimal amounts of preprocessing. They have wide applications in image and video recognition, recommender systems and natural language processing.Convolutional neural networks are often used in image recognition systems because of their high accuracy. They have achieved an error rate of **0.23 percent** on the MNIST database, which as of February 2012 is the lowest achieved on the database. Other deep learning

algorithms are not good at classifying objects into fine-grained categories such as the particular breed of dog or species of bird, whereas convolutional neural networks handle this with ease.

★ Tensorflow

TensorFlow is an open source software library for numerical computation using data flow graphs. Nodes in the graph represent mathematical operations, while the graph edges represent the multidimensional data arrays (tensors) communicated between them. The flexible architecture allows you to deploy computation to one or more CPUs or GPUs in a desktop, server, or mobile device with a single API. TensorFlow was originally developed by researchers and engineers working on the Google Brain Team within Google's Machine Intelligence research organization for the purposes of conducting machine learning and deep neural networks research, but the system is general enough to be applicable in a wide variety of other domains as well.

★ Graphical Processing Unit

Traditional machine learning uses handwritten feature extraction and modality-specific machine learning algorithms to label images or recognize voices. However, this method has several drawbacks in both time-to-solution and accuracy.

Today’s advanced deep neural networks use algorithms, big data, and the computational power of the GPU to change this dynamic. Machines are now able to learn at a speed, accuracy, and scale that are driving true artificial intelligence.

Deep learning is used in the research community and in industry to help solve many big data problems such as computer vision, speech recognition, and natural language processing. Practical examples include:

- Vehicle, pedestrian and landmark identification for driver assistance
- Image recognition
- Speech recognition and translation
- Natural language processing
- Life sciences

The NVIDIA Deep Learning SDK provides high-performance tools and libraries to power innovative GPU-accelerated machine learning applications in the cloud, data centers, workstations, and embedded platforms.

★ Python Language

Python if a popular scientific language and a rising star for machine learning. I'd be surprised if it can take the data analysis mantle from R, but matrix handling in NumPy may challenge MATLAB and communication tools like IPython are very attractive and a step into the future of reproducibility. I think the SciPy stack for machine learning and data analysis can be used for one-off projects (like papers), and frameworks like scikit-learn are mature enough to be used in production systems.

Libraries used in Python are:-

- Tensorflow
- SVM
- Scipy
- Numpy
- Matplotlib
- Pickle
- Sklearn

★ PHP

PHP is a server-side scripting language designed primarily for web development but also used as a general-purpose programming language. Originally created by Rasmus Lerdorf in 1994, the PHP reference implementation is now produced by The PHP Development Team.PHP originally stood for *Personal Home Page*, but it now stands for the recursive acronym *PHP: Hypertext Preprocessor*.

PHP code may be embedded into HTML code, or it can be used in combination with various web template systems, web content management systems and web frameworks. PHP code is usually processed by a PHP interpreter implemented as a module in the web server or as a Common Gateway Interface (CGI) executable. The web server combines the results of the interpreted and executed PHP code, which may be any type of data, including images, with the generated web page. PHP code may also be executed with a command-line interface (CLI) and can be used to implement standalone graphical applications.

★ XML

In computing, **Extensible Markup Language** (**XML**) is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. The W3C's XML 1.0 Specification and several other related specifications,—all of them free open standards—define XML.

The design goals of XML emphasize simplicity, generality, and usability across the Internet. It is a textual data format with strong support via Unicode for different human languages. Although the design of XML focuses on documents, the language is widely used for the representation of arbitrary data structures such as those used in web services.

★ Android and Android Studio

Android is a mobile operating system(OS) currently developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets. Android's user interface is based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input. In addition to touchscreen devices, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics. As of 2015, Android has the largest installed base of all operating systems. It is the second most commonly used mobile operating system in the Unites States, while iOS is the first.

Android software development is the process by which new applications are created for the Android operating system. Applications are usually developed in Java programming language using the Android software development kit(SDK), but other development environments are also available.

Android Studio is the official IDE for Android application development, based on IntelliJIDEA. On top of the capabilities you expect from IntelliJ, Android Studio offers:

- Flexible Gradle-based build system
- Build variants and multiple apk file generation
- Code templates to help you build common app features
- Rich layout editor with support for drag and drop theme editing
- lint tools to catch performance, usability, version compatibility, and other problems
- ProGuard and app-signing capabilities
- Built-in support for Google Cloud Platform, making it easy to integrate Google Cloud Messaging and App Engine

★ Android Volley

Volley is an HTTP library that makes networking for Android apps easier and most importantly, faster. Volley is available through the open AOSP repository.

Volley offers the following benefits:

- Automatic scheduling of network requests.
- Multiple concurrent network connections.
- Transparent disk and memory response caching with standard HTTP cache coherence.
- Support for request prioritization.
- Cancellation request API. You can cancel a single request, or you can set blocks or scopes of requests to cancel.

- Ease of customization, for example, for retry and backoff.
- Strong ordering that makes it easy to correctly populate your UI with data fetched asynchronously from the network.
- Debugging and tracing tools.

★ Dataset

ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images. ImageNet is an image dataset organized according to the WordNet hierarchy. Each meaningful concept in WordNet, possibly described by multiple words or word phrases, is called a "synonym set" or "synset". There are more than 100,000 synsets in WordNet, majority of them are nouns (80,000+). In ImageNet, we aim to provide on average 1000 images to illustrate each synset. Images of each concept are quality-controlled and human-annotated. In its completion, we hope ImageNet will offer tens of millions of cleanly sorted images for most of the concepts in the WordNet hierarchy. The ImageNet project is inspired by a growing sentiment in the image and vision research field – the need for more data. Ever since the birth of the digital era and the availability of web-scale data exchanges, researchers in these fields have been working hard to design more and more sophisticated algorithms to index, retrieve, organize and annotate multimedia data. But good research needs good resource. To tackle these problem in large-scale (think of your growing personal collection of digital images, or videos, or a commercial web search engine’s database), it would be tremendously helpful to researchers if there exists a large-scale image database. This is the motivation for us to put together ImageNet. We hope it will become a useful resource to our research community, as well as anyone whose research and education would benefit from using a large image database.

3.3 Methodology

1) Convolutional Neural Network

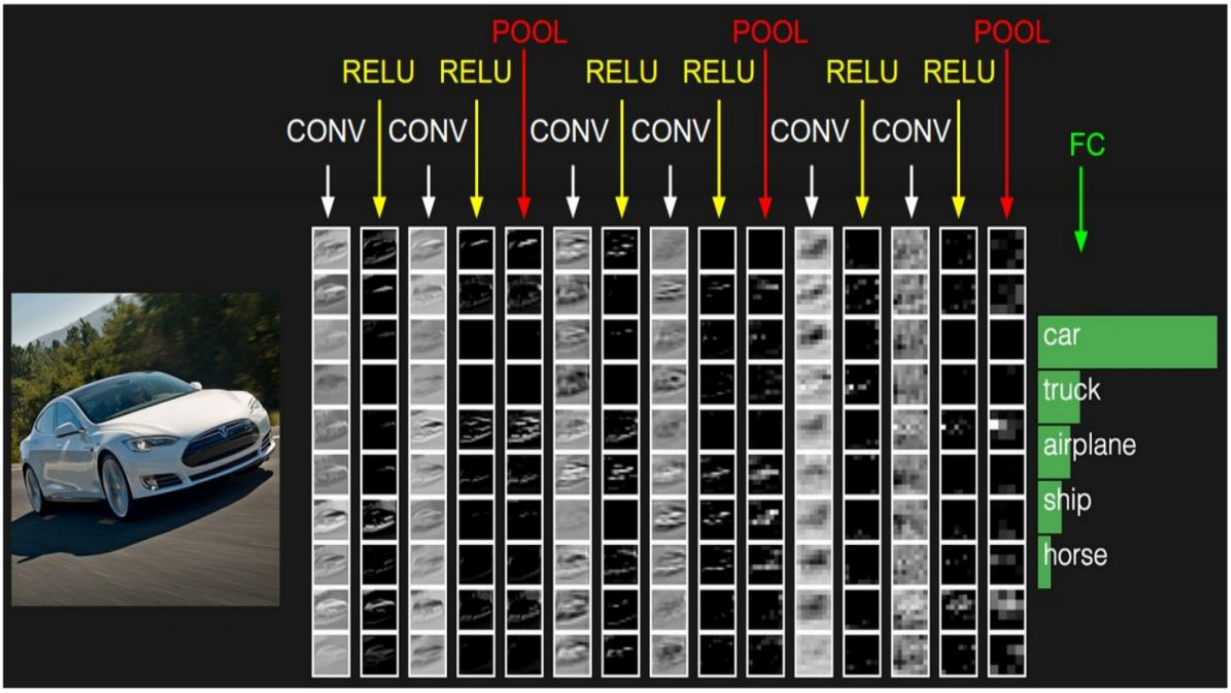
In machine learning, a convolutional neural network (CNN, or ConvNet) is a type of feed-forward artificial neural network in which the connectivity pattern between its neurons is inspired by the organization of the animal visual cortex, whose individual neurons are arranged in such a way that they respond to overlapping regions tiling the visual field.Convolutional networks were inspired by biological processes and are variations of multilayer perceptrons designed to use minimal amounts of preprocessing. They have wide applications in image and video recognition, recommender systems and natural language processing.

There are 4 main layers in CNN:

1. Convolutional - The primary purpose of Convolution in case of a ConvNet is to extract features from the input image. Filters act as Feature detectors.The value of the filter, is in fact, not manually provided but the machine chooses the suitable value by training and changing its weights.
2. Non-Linear (ReLU {Rectified Linear Unit}) - ReLU is an element wise operation (applied per pixel) and replaces all negative pixel values in the feature map by zero. The purpose of ReLU is to introduce non-linearity in our ConvNet, since most of the real-world data we would want our ConvNet to learn would be non-linear (Convolution is a linear operation – element wise matrix multiplication and addition, so we account for non-linearity by introducing a non-linear function like ReLU). Ex: Output = Max (zero, input)
3. Pooling or Sub-Sampling - Spatial Pooling (also called subsampling or downsampling) reduces the dimensionality of each feature map but retains the most important information. Spatial Pooling can be of different types: Max, Average, Sum etc

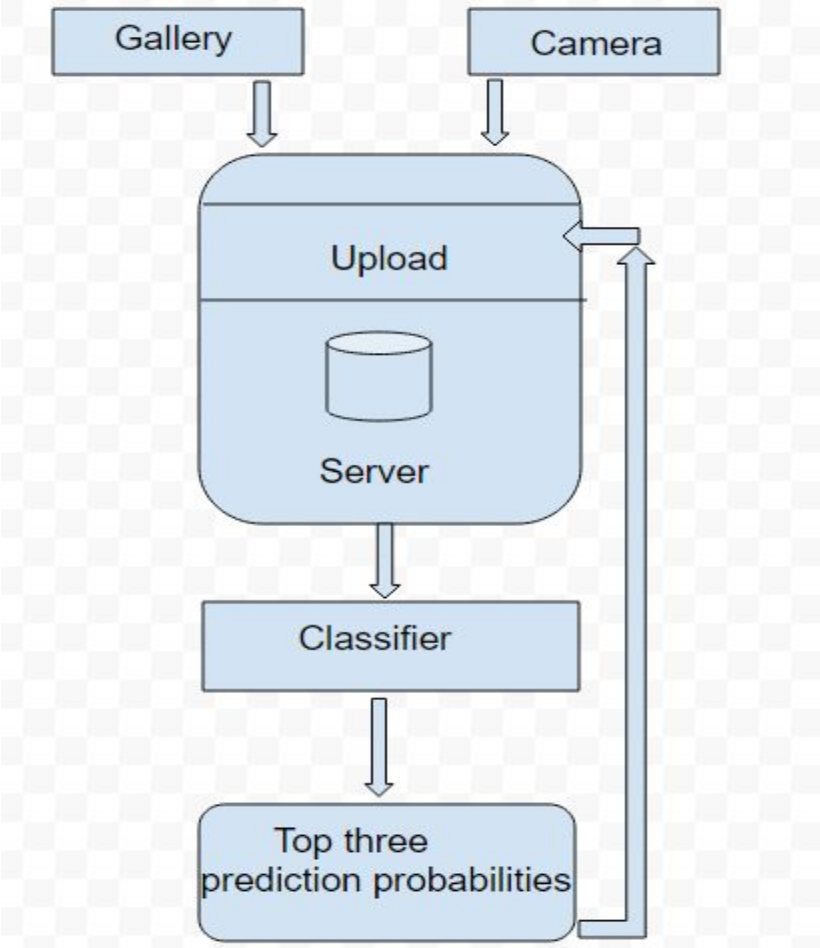
- It makes the input representations (feature dimension) smaller and more manageable.
- Reduces the number of parameters and computations in the network, therefore, controlling overfitting.
- Makes the network invariant to small transformations, distortions and translations in the input image (a small distortion in input will not change the output of Pooling – since we take the maximum / average value in a local neighborhood).
- Helps us arrive at an almost scale invariant representation of our image (the exact term is “equivariant”). This is very powerful since we can detect objects in an image no matter where they are located .

4. Classification (Fully-Connected Layers) - Predicts the classes. Finally, after several convolutional and max pooling layers, the high-level reasoning in the neural network is done via fully connected layers. A fully connected layer takes all neurons in the previous layer (be it fully connected, pooling, or convolutional) and connects it to every single neuron it has. Fully connected layers are not spatially located anymore (you can visualize them as one-dimensional), so there can be no convolutional layers after a fully connected layer.

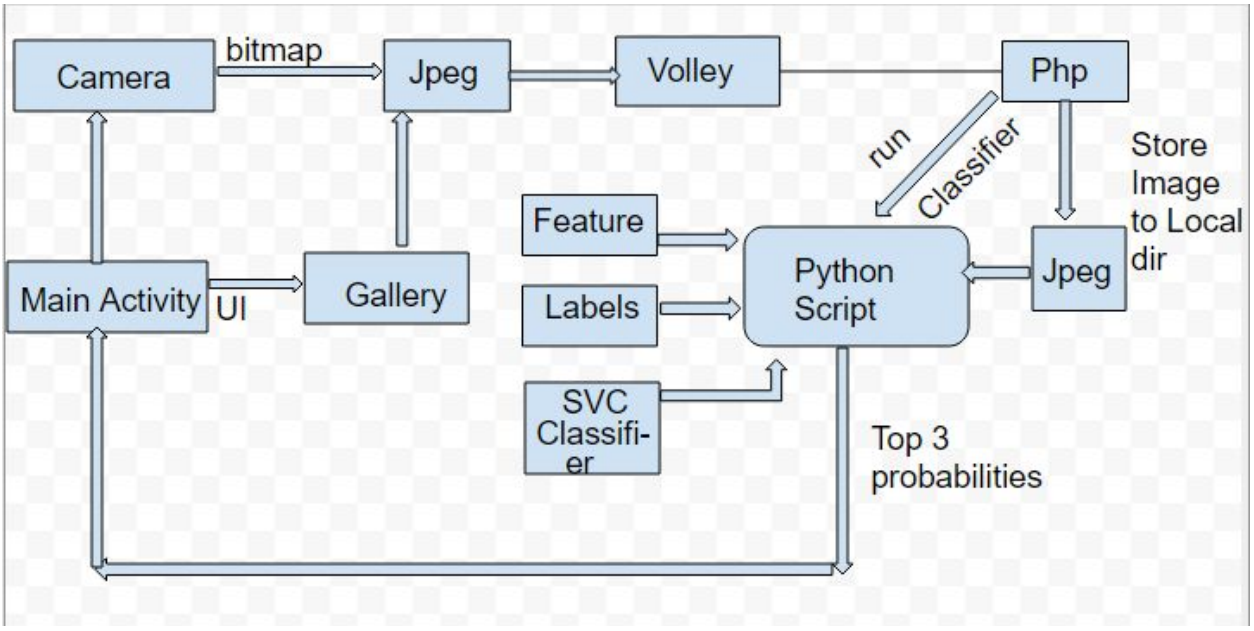


4. PROCESS MODELS

i)Control flow diagram

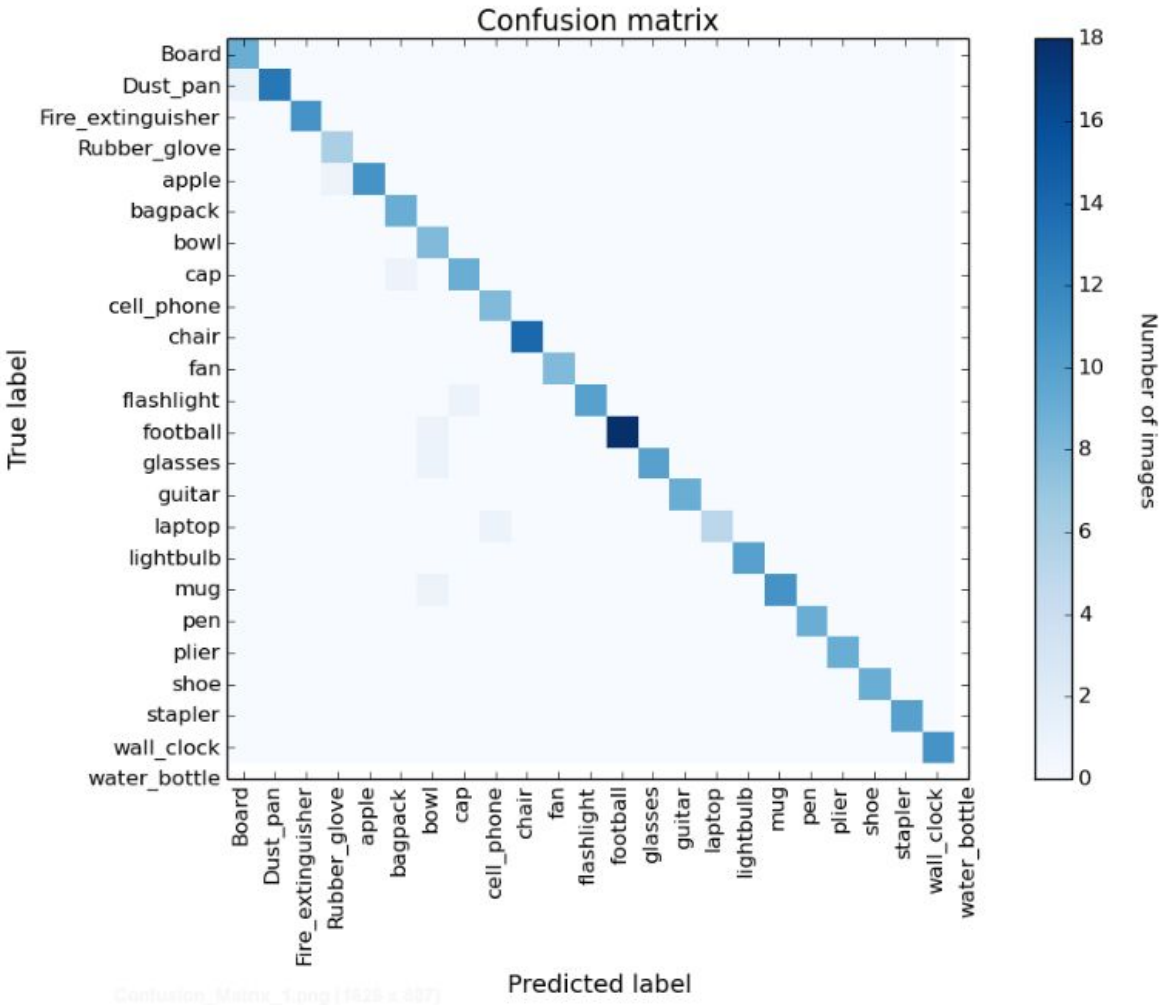


ii) Data Flow Diagram

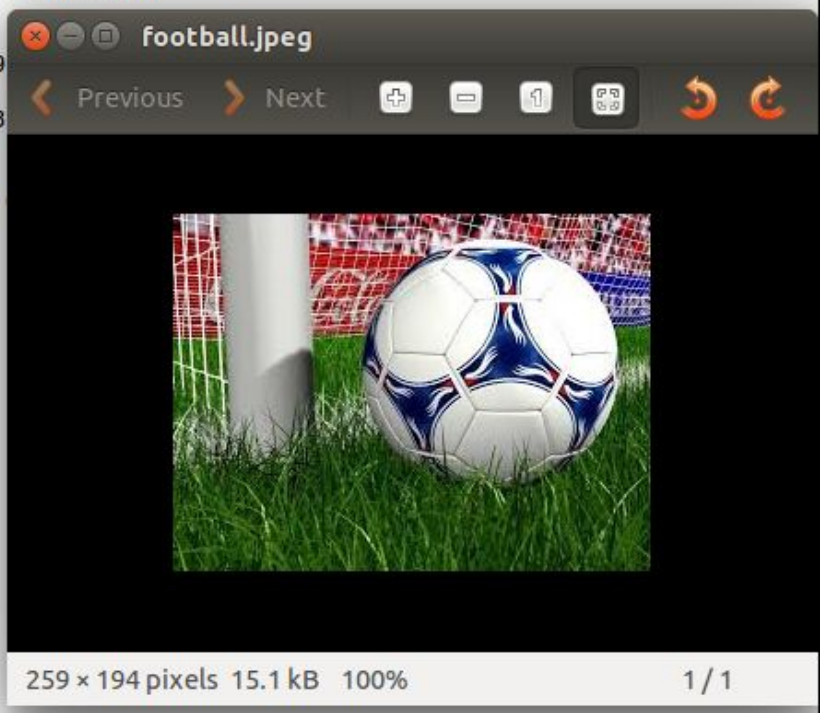


5. IMPLEMENTATION

PROJECT SCREENSHOTS:



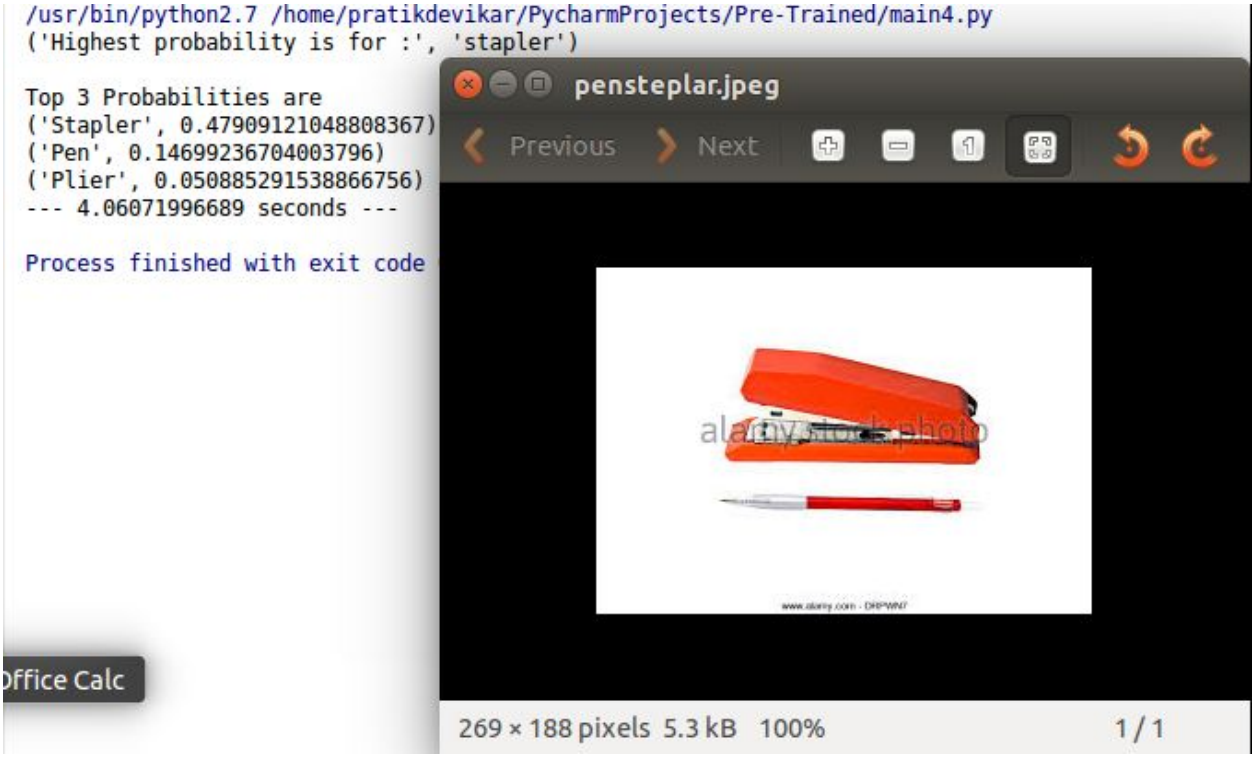
```
/usr/bin/python2.7 /home/pratikdevikar/PycharmProjects/Pre-Trained/main4.py
('Highest probability is for :', 'football')
()
Top 3 Probabilities are
('Football', 0.74940013411586959
('Bowl', 0.029318269964116985)
('Stapler', 0.019306125557070373
--- 14.6773660183 seconds ---

Process finished with exit code
```



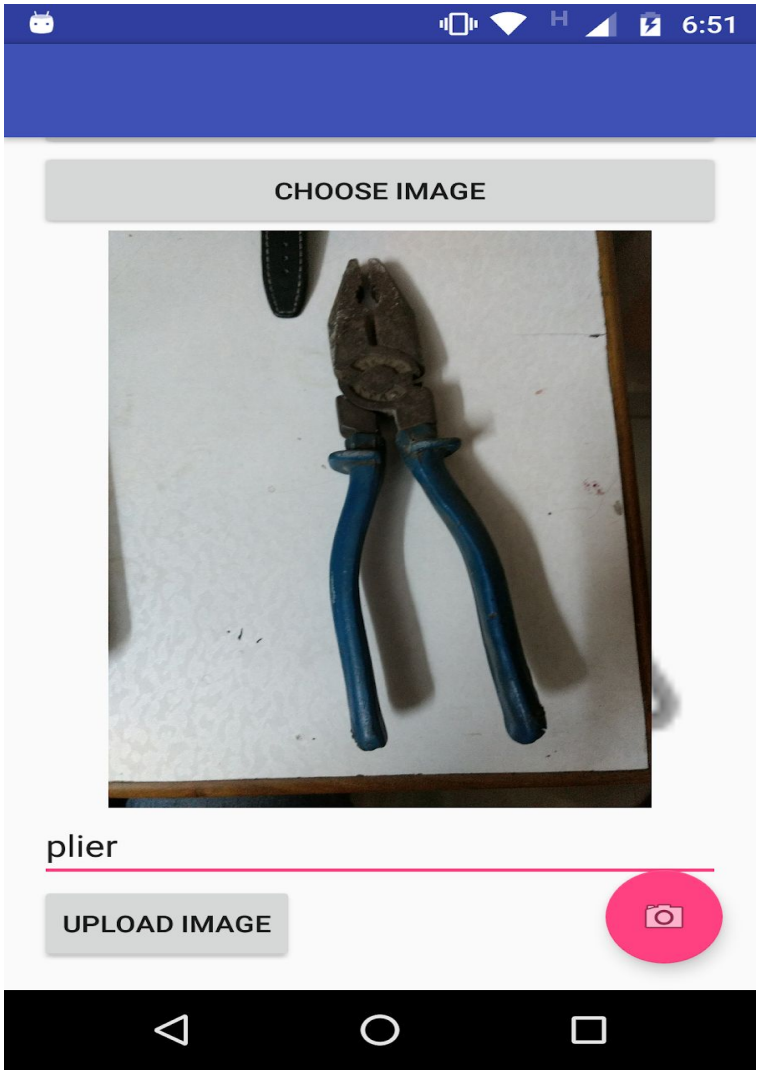
Test case 1

```
/usr/bin/python2.7 /home/pratikdevikar/PycharmProjects/Pre-Trained/main4.py
('Highest probability is for :',

Top 3 Probabilities are
('Laptop', 0.40665448593212133)
('Cell Phone', 0.114127605651977
('Bowl', 0.050379409467486674)
--- 10.7551121712 seconds ---

Process finished with exit code
```

Test case 2


/usr/bin/python2.7 /home/pratikdevikar/PycharmProjects/Pre-Trained/main4.py
('Highest probability is for :', 'stapler')
Top 3 Probabilities are
('Stapler', 0.47909121048808367)
('Pen', 0.14699236704003796)
('Plier', 0.050885291538866756)
--- 4.06071996689 seconds ---
Process finished with exit code
pensteplar.jpeg
Previous
Next
269 × 188 pixels 5.3 kB 100%
1 / 1
Office Calc

Test case 3



Main Activity

5.Conclusion

- Implemented a classifier to extract tags from images.
- Created a server side script to run the classifier,extract tags,and respond to request.
- For user interface,developed an android and to provide image to server.
- Classifier has obtained accuracy around 80%
- Classifier can also perform multi-class classification.

6.Future Work

- This architecture can be extended to implement image retrieval system.
- Android app can be extended to do client side tagging.
- Classifier can be extended to tag videos.
- Generated Tags can also be used for Automatic Image Captioning.

7.References

- "TagSense: A Smartphone-based Approach to Automatic Image Tagging" - Chuan Qin, Xuan Bos, Romit Roy Choudhury and Shrihari Nelakuditi

- "Automatic Image Tagging' - Sean Moran, University of Edinburgh

- "Comprehensive Study on Automatic Image Annotation" - International Journal of Emerging Technology and Advanced Engineering - Dhatri Pandya, Prof. Bhumika Shah

- https://www.tensorflow.org

- https://developer.android.com