1. **Introduction**

***1.1* *Challenges***

**1.1.1 Retrieving HTML Code**

The first challenge was to retrieve the HTML script of the Google page which was loaded after searching for images. The HTML script was retrieved using shell script.

**1.1.2 Retrieving URLs**

Once the HTML script was retrieved, we then had to fetch the URLs of the images from those pages. The project takes images from the internet. So from the whole HTML script, we had to find the specific lines where URLs were located.

**1.1.3 Downloading Images**

After finding the particular images, we had to download them so as to modify them. Thus, downloading was a challenge faced. Write how we downloaded.

**1.1.4 Resizing Images**

This part was the major and most challenging part of the project. After searching for the images, retrieving the HTML page of search results, finding the URLs of the images, and downloading them, the final task was to resize the images. The process of resizing has been explained in the later part of the report.

***1.2 Motivation For The Work***

Machine Learning is a very popular field right now, but obtaining enough data for experimentation is very complicated. But there are a large number of resources available on the internet which can be used as a source of data, for the datasets. This was one motivation to build a tool which could simplify the process of gathering data. Another motivation for this project was to apply on the knowledge gained throughout the Unix Programming and Practice (IT202) course, about various Unix Commands and concepts of Shell Programming. This was an excellent opportunity for solving a real world problem using the concepts learned during the course. And the final motivation was to build upon the knowledge gained throughout the course and learn more about UNIX.

**2. Methodology and Framework**

***2.1 System Requirements***

**2.1.1 ImageMagick**

ImageMagick is a software suite to create, edit, compose or convert bitmap images. The ‘convert’ command is used to reformat images. This command could you be used only after installing the ImageMagick library on the system. The functionality of ImageMagick is typically utilised from command line. ImageMagick used multiple computational threads to increase performance and can read, process or write mega-, giga- or tera- pixel image sizes. ImageMagick is free software delivered as a ready-to-run binary distribution or as source code that you may use, copy, modify, and distribute in both open and proprietary applications.

**2.1.2 OS Supporting Bash Terminal**

Since the code uses Linux Commands, system should have an Operating System which supports Bash Terminal. This can be done by installing Ubuntu, or softwares like Cygwin or Bash on windows. We can then run the commands through the command prompt. The version of Ubuntu on which we had tested this is 16.04, with the Kernel version 4.4.0-101.

**2.1.2 Active Internet Connection**

The images which are to be modified are first searched on Google and then being downloaded from the URLs. For this, an active internet connection is required, without which the images would not be downloaded.

**2.1.3 Memory to Store Images**

After searching the images, they are downloaded and hence stored in the internal memory of our system. For this, a sufficient amount of disk space is required. The user can download as many images as he/she wants and then modify them. The modified images will be stored in the system, hence the system should have enough space to store the images.

***2.2 Algorithms and Techniques***

The images were scraped as follows:

● Select crawler based on user choice, wget or curl

● Read the list of categories from query\_list.txt

● For each category:

○ Fetch the Google Image Search Results page using the crawler

○ Match the image links using grep and awk

○ Store the list of image links to a file using cat

○ Download images from the file list of URLs using the crawler

○ Resize the images to the desired resolution using convert

**3. Implementation**

***3.1. List of Main Unix Commands***

The following shell commands are used in implementing the Image Scrapper: -

* **awk**
* **convert** (**ImageMagick)**
* **curl**
* **egrep**
* **which**
* **wget**

***3.2. Uses and its Syntax***

**1)** **awk:**-

awk is one of the most powerful tools in Unix used for processing the rows and columns in a file. It is used for manipulating data and generating reports. It searches files for text containing a pattern. When a line or text matches, awk performs a specific action on that line/text.

awk reads from a file or from its standard inut, and outputs to its standard output. awk does not get along with non-text files.

***Syntax: -***

Awk ‘/search pattern 1/{Actions}/search pattern 2/ {Actions}’ filename

One of the most commonly flags as options is numeric sort. -n, --numeric-sort

**2) convert (ImageMagick): -**

Convert command in UNIX is use to perform operations on an image like to resize it, make it blur, crop, despackle, dither, draw on ,flip, join, re-sample and many more. It also help in conversion between image formats. This command is a member of the ImageMagick suite of tools.

***Syntax: -***

convert input-file [options] output-file

**3) curl: -**

Curl command is used to transfer from or to a server, using one of the protocols: HTTP, HTTPS, FTP, FTPS, SCP, SFTP,TFIP, DICT, TELNET, LDAP or FILE

***Syntax: -***

curl [option] [url…]

**4) egrep: -**

Egrep command is used to search for a pattern extended regular expressions. Egrep is essentially the same as running grep with the option -E option. This version of grep is efficient and fast.

In case of egrep, even if you do not escape the meta-characters,it would treat them as special characters and substitute them for their special meaning instead of treating them as a part of string.

***Syntax: -***

egrep [option] pattern [file…]

**5.)** which: -

Which is used to return the pathnames of the files ( or links ) which would be executed in the current environment, had the filename been given as a command in a strictly POSIX-conformant shell. It does this by searching the paths in the path environment variable for executable files matching the names of arguments.

Which doesn’t allow symbolic links.

***Syntax: -***

which -a [filename] …..

**6.) wget:-**

wget command stands for “web get”. It is a command line utility for downloading files from the Internet. It supports downloading multiple files, downloading in the background, resuming downloads, limiting the bandwidth used for downloads and viewing headers. It can also be used for taking a mirror of a site and can be combined with other UNIX tools to find out things like broken links on a site.

***Syntax: -***

wget [options] [url…..]

**4. Results and Analysis**

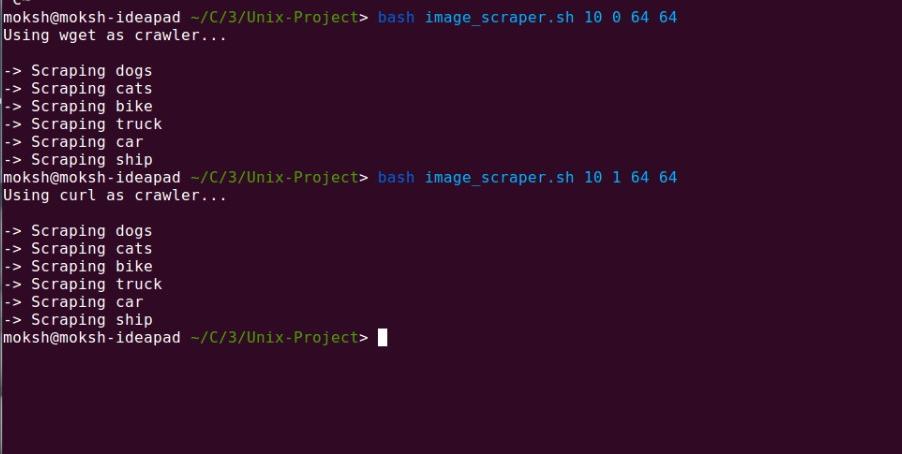
****

Figure 4.1

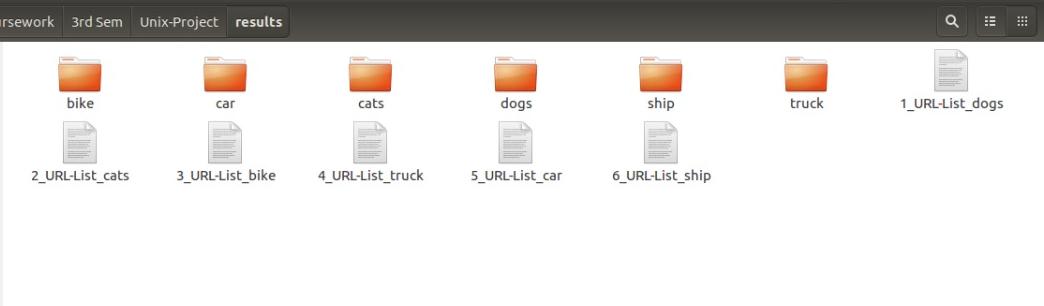


Figure 4.2



Figure 4.3

* The crawler is properly selected based on the command line arguments given.
* The list from the file is properly read and the appropriate pages are parsed.
* Image links are properly matched and stored in the appropriate file.
* The images are downloaded in their respective folders.
* The images are properly resized to the specified dimensions.

**5. Conclusion**

The aim of this project was to search for required images, fetch URLs of those images, download the images and then resize the images using UNIX commands. The resized images were then stored in our system.

As detailed in the preceding sections, we have successfully implemented the model to search, download and then modify images. Various functions were used in this project code, thus we got to know new concepts such as declaring functions, defining functions, calling them and passing parameters to these functions. Along with the normal commands such as cat, mkdir, rm, rmdir, touch, etc. we also learned the uses and applications of commands such as curl, awk, egrep, which, etc.

The main concept behind modifying such images was to create a data set which could be easily imported into Python programs and also be helpful for beginners to obtain and use the data in various Machine Learning applications.

The future improvements could be searching and downloading the images from various other sources. This project searches and downloads images based on a Google Search. We could also take the images from some other platforms such Flickr API, thus increasing the scope of the project, making it practically possible to use in many real life scenarios.

This project has been very helpful in the leaning process, and have a new appreciation for shell scripting. On completion of the project, the major thing realized was that the knowledge of shell scripting and major UNIX commands can be as useful, helpful and interesting as the applications of any other programming language, and perform some operations easily and in a much less complex manner.

**References**

[1] Google Image Search, [https://images.google.com](https://images.google.com/)

[2] wget Manual Page, <https://www.gnu.org/software/wget/manual/>

[3] curl Manual Page, <https://curl.haxx.se/docs/manpage.html>

[4] egrep Manual Page, <https://linux.die.net/man/1/egrep>

[5] awk Manual Page, <https://www.gnu.org/software/gawk/manual/gawk.html>

[6] convert: ImageMagick Manual, <https://www.imagemagick.org/script/convert.php>