# IMAGE PROCESSING THROUGH TERMINAL USING VOICE AND TEXT COMMANDS


Nafeem Haque- 13201047

Quazi Ahanaf Alif -14101166

Supervisor : Dr. Md. Ashraful Alam

**Department of Computer Science And Engineering**

**Brac University.**

**Submitted on : 14th December,2017**

# DECLARATION

We, hereby declare that this thesis is based on the results found by ourselves. Materials of work found by other researcher are mentioned by reference. This thesis, neither in whole or in part, has been previously submitted for any degree

## Signature of Supervisor

_____

## Signature of Author

_____

## Signature of Author

_____

# ACKNOWLEDGEMENTS

We are eternally grateful to our professors, friends and family who helped us in every step of the way to complete this research.

We would like to express our sincerest gratitude to our supervisor Dr. Md. Ashraful Alam for his continuous support, encouragement and guidance in every step of our research. Without him this research would not have been completed.

# Table of Contents

# List of Figures

# List of Tables

# Abstract

We proposed a new approach of image processing based on terminal using voice and text commands. Any terminal in operating system has a purpose which is doing tasks in a very short time. In the proposed method a terminal is developed to take input as text or voice commands. Where the terminals used in operating systems can be used to modify tasks or implement task or organize them, our terminal is not very different except for it will work on images. The idea we proposed enhances the functionalities of a conventional terminal but focuses on processing images. Also the fact that voice recognition is added as an external feature increases the very idea of image processing and adds to the modern computing world concept of human computer interfacing. The terminal will work as a medium of taking inputs then it will send it to the main program or engine that converts the inputs as commands and implements it over an image. The idea of doing image processing via terminal is more powerful because it gives more control and takes less time than processing images in regular way. Hopefully our work will add a new phase to the current understandings and ways to do image processing. Besides adding voice recognition as human computer interfacing will add up to the modern ways of computer technology.

# Chapter 1

## Introduction

## 1.1 Motivation

Image processing is no longer an uncommon topic in recent computer technology. Image processing can be used in many ways to engineer other technologies and also it can be done or implemented in many ways. On the huge variety under image processing digital image processing is the most common one and almost every computer or mobile user uses this technique to upgrade their photos or images. Conventional way of editing images is using software like Photoshop, Microsoft paint, Ink scape and many other popular programs.

The motivation behind our work is to upgrade the usual ways of processing images and enhance it to the level where it is no longer a complicated thing to do. Al though a lot of people use image editing software and technology but a few of them understand it or can master it. So in simple words it is hard for all the users to get the most out of any image processing program. Now to solve this problem many ways can be implemented. But our idea is to take the image processing to the level where people will use the daily language they use and edit or process image. So when people can use their usual language rather than using the complex functions of a program they can easily take the advantage of image processing in their regular life.

Another motivation that adds up to modern technology is thinking on the perspective of human computer interfacing. HCI is a huge research area, different types of research and development is currently being done on the topic. Our idea of adding voice recognition as an extension to the terminal can easily be used in HCI. We can use voice and convert it to commands to be implemented in image processing.

## 1.2 Literature Review

Several Image Processing techniques have been used in the past. [1] Students of Dept. of Radio electronics, Czech Technical University in Prague proposed an idea to process image using MATLAB. But by this process only the increase and decrease of the brightness of a picture, noise adding to a picture and sharpening a picture can be done. No other filters of features can be done in this application or process.

In 1995, P Lelong, G Dalm, J Klijn Proposed a method on image processing. [2] The invention relates to a method of processing images for constructing a target image from adjacent images having a fixed frame line and referred to as source images, said source and target images having substantially common view points. Though this was a great invention, it had some limitations.

There is also a proposed idea that evaluates the quality of raster images by quantitative and qualitative indicators. [3] Quality of image is formed on the basis of the calculation and correction of brightness, contrast, tone contrast, luminance contrast and tone saturation. Output result of program is the corrected image with quantitative estimates. Quantitative estimates are shown as histograms and tables. This is a new way of processing or editing an image. But, like other ideas, it has some drawbacks. For example- it can't really fulfill all the necessary functions of image processing which the professional photographers or power users need.

An efficient method for precise computation of image-aware geodesic distances for image editing algorithms has also been proposed in 2014 by the students of CS department of Israel Institute of Technology. [4] It exploits the connection between image representation as a mapping from a Cartesian grid and as a collection of its level sets, organized into a tree structure. These distances are usually computed in the image domain, where they can be efficiently approximated using Dijkstra's algorithm or the fast marching or fast sweeping

methods. Despite their wide use, each of the above algorithms introduces approximation errors and inconsistencies that may degrade the quality of the editing.

There is also a method called method of Poisson image editing on FPGA**. [5]** Poisson Image Editing is a powerful method to overlay an image on another image seamlessly. In this method, however, a simple equation is repeatedly applied to each pixel, and this repetition makes its computational complexity very high. But its processing speed is still fast enough for real-time processing of HD images. So, it has been a good way to process image in real time but not in still mode images.

So, we are proposing a new and easy way to process image with voice and text commands through terminal without any hassle. So, we will be able to overcome all the complexity and hardness of image processing. Our proposed idea also promised to make use of the necessary filters in a very efficient way.

## 1.3 Thesis Orientation

We have divided the thesis work in different chapters, and elaborately explained in each chapter to make it easier to understand as a whole. In the next chapter we have discussed the very basics of image processing and then we have given some of the ways to understand terminal technology. Then we have explained our proposed system in chapter 3 and explained the system architecture and the processes. In chapter 4 we have showed experimental setups and the results of our system. At the last chapter we have discussed about the future possibilities of our work and then concluded.

# Chapter 2

## Fundamentals of Image Processing and Terminal Technology

## 2.1 Image Processing

Image processing is a complex feature of modern technology. To put it in simple words, Image processing is nothing but applications of different mathematical models and algorithms over a digital image to create special features or to research. There are different fields where we use image processing.

Different sectors of Image Processing :

- Medical Imaging
- Automation
- Computer Vision
- Robotics
- Space Research
- CGI Effects
- Mobile Applications
- Digital Gaming
- Artificial Intelligence
- Web Applications
- Face Recognition
- Security Mechanisms
- Feature Identification
- Study Analysis

There are huge applications of digital image processing. In any modern imaging system or high graphics system image processing plays a vital role.

### 2.1.1 Types of Image Processing

There are mainly two types of image processing, one is the digital image processing and the other is the analog image processing.

The analog images are the images that we see in real life, for example paintings on a wall. Whenever we use analog images to justify or collect data, that process is known as analog image processing. An example of analog image processing would be X-rays that doctors check and analyze patient's condition.

Digital image processing on the other hand is images that we see in digital format. Whenever we use digital images to add feature to it or extract data from the image by different models then it is called digital image processing.

### 2.1.2 Methods of Digital Image Processing

Digital image processing has a wide area of applications. Different area uses different procedures to do digital image processing. But to scale it overall, image processing can be done by two procedures, one is the pixel based image processing and the other is the vector based image processing.

Pixel based or raster graphics works on the image's pixels which can cause a lot of problem. Like scaling, it is hard to scale an image with pixel based processing.

Vector image processing is convenient because it transforms any given image to a mathematical model or a vector so that easily different types of processes can be done. But this way of image processing is way more complicated and it also needs powerful and expensive machines to process.

### 2.1.3 Importance of Digital Image Processing

In one simple sentence modern technology era massively depends on digital image processing. What we research today or the complicated systems we have developed so far needs image processing. Although image processing algorithms have been arriving from the very start of the computer technology, it is at present that we use it in our everyday life. Image processing is not anymore a rare concept in our life. We use it in our cell phones, computers other portable devices. Gaming and other digital media uses image processing over a wide span. Not just for entertainment many of the advanced research highly depends on the image processing.

Some most common implementations of digital image processing :

1. Face Recognition : Five years ago we had to tag people on social media, but now a days social networking platform like Facebook, Instagram uses face recognition technology. Whenever we upload an image we can see this happening.
2. Gaming : All of computer or mobile device users have played at least one game in their life time. Games are a common example of image processing.
3. Movies : Most of the movies we watch today uses high graphics and special effects. Although these sort of movie production use highly advanced technology but at the core it needs image processing.
4. Simulation : To make concepts or theories we use computer generated simulations. Simulations are done by image processing.
5. Investigation : Many of the accident or events are analyzed by image processing. CSI, CID and other departments that investigate crime scenes uses image processing technology to find out clues of the occurrence.
6. Weather Prediction : Weather forecast are done by processing satellite image processing.
7. Statistics Creation : Many of the statistics like crop production, poverty investigation and other statistics can be created using image processing. In this case satellite images plays a vital role.
8. Space research : Image processing is widely used in space research. Calculation of distance, type of objects, properties and other information can be collected by applying image processing.

9. War Technology : War technology is depended on image processing. For example, Driving a drone needs high resolution image processing

There are far more ways to use image processing than the mentioned ones. So we understand the importance of image processing.

## 2.2 Terminal Technology

Old days computers ran on operating systems like DOS. These sort of computer technology vastly depended on terminal technology. Former and popular way of computation was depended on terminals.

Terminal is a program that can run different commands. It was popular in the first generation of computer technology because of the lack of high graphics processing machines. It was simplest form of technology that gave tremendous power to the user. Although present day technology has higher graphics and most of the people use graphical operating systems, still the experts use terminal to make the best out of computer technology. The terminal is a really amazing technology or program that is able to interact with hardware directly and do things that a graphical user program is not able to do.

### 2.2.1 Terminals in Operating Systems

Terminal in any computer operating system is a really common program. Any type of operating system uses terminal. Although the versions and the work capability is different from each other but all of the OS has their own terminals

Windows has command prompt as terminal. A lot of the features can be accessed by using the command prompt. Mac has its own terminal. The terminal that is a lot powerful than any other operating system's terminal is the terminal of Linux. Linux user uses mainly the terminal for

advanced work, and its very popular. Linux terminal is so powerful that it is possible to hack big networks from it. But for that, one needs to have extensive knowledge.

### 2.2.2 Terminals in Networking

Recently we don't hear much about SSH technology because a lot of advanced tool has arrived. But SSH was very popular earlier to remotely access computers. SSH is a form of terminal and it was widely used previously.

The routers we use today at home has graphical user interface where user can setup the network but the major routers used widely still uses CLI (Command Line Interface) to setup paths and connections, and this is done by using terminal.

Databases and other systems still use queries which are executed in terminal like technology.

As we can see that terminal is a very old technology but for the strength of this program it has still the dominance over computer technology.

# Chapter 3

## Proposed System

The system takes image as an input, extracts commands from the terminal then sends it all to control engine. The control engine checks for errors and processes the commands, then it decides the edit class in which the image will go through processing and the processed image will come out as an output and then the system will display the output image.

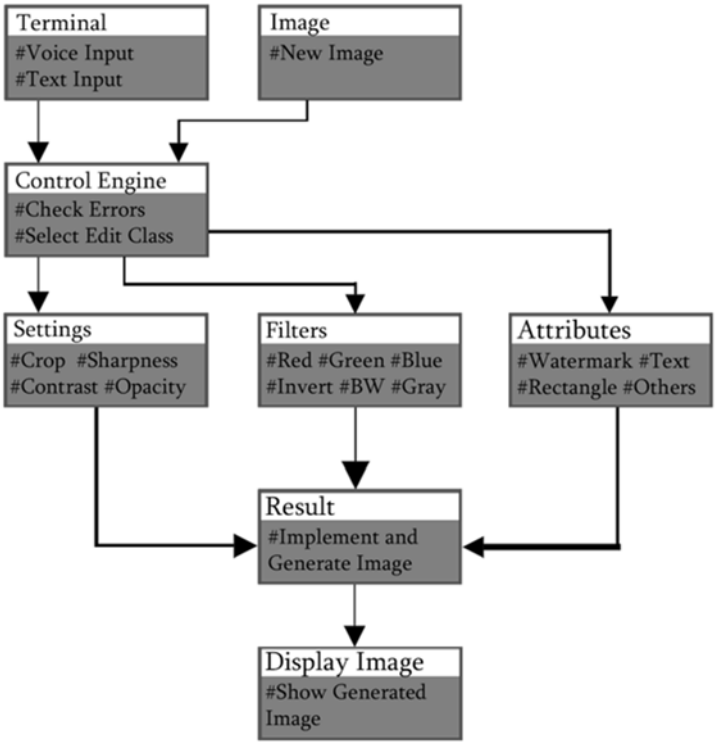



Fig. 1 Architecture of proposed system

## 3.1 Terminal

The terminal we built is a graphical user interface. It is a medium that is used to take inputs either as text or as voice of the user. The input is then passed to the control class.

### 3.1.1 Interface


Fig. 2 Interface of the terminal

The graphical user interface is mainly made up of two parts. At the top region the menu bar and just below that the rest of the region is covered with a text area for writing commands.

The top region or the menu bar has menu items accordingly File, Settings, Learn, Voice Command, Tools, About.

File header has 3 sub-sections which are open, renew terminal and exit. Open is used to locate images on hard drive and open it for the program to use. Renew terminal is for clearing the terminal. Exit is used for closing the entire program.

Settings have font color, size and background color to adjust the interface according to the user's preferences.

Learn is basically for the user to understand the program. It teaches how to use the commands and describes generally about the software.

Voice Command is used when the user wishes to speak the commands instead of writing in the terminal.

About header is for providing the information about the project and developers.

### 3.1.2 Menu Bar

The Menu bar in the interface is like typical menu bar in any software. The menu bar gives the control over the interface, individual user can customize interface according to their taste. They can change the interface's color, text, font according to their taste. The menu bar also activates the voice command procedure.


Fig.3 Top menu bar

### 3.1.3 Text Area

Below the menu bar there is a text area which is shown in the previous interface section. This part of the terminal is used to give commands to the program, it also takes the command and passes to the control engine.

### 3.1.4 Scripting

In the top menu section, under the Tools there is a sub-menu called Scripting Panel. This works almost like a text editor, but the difference is we use it for scripting, any scripts can be written, saved, opened and run using this panel

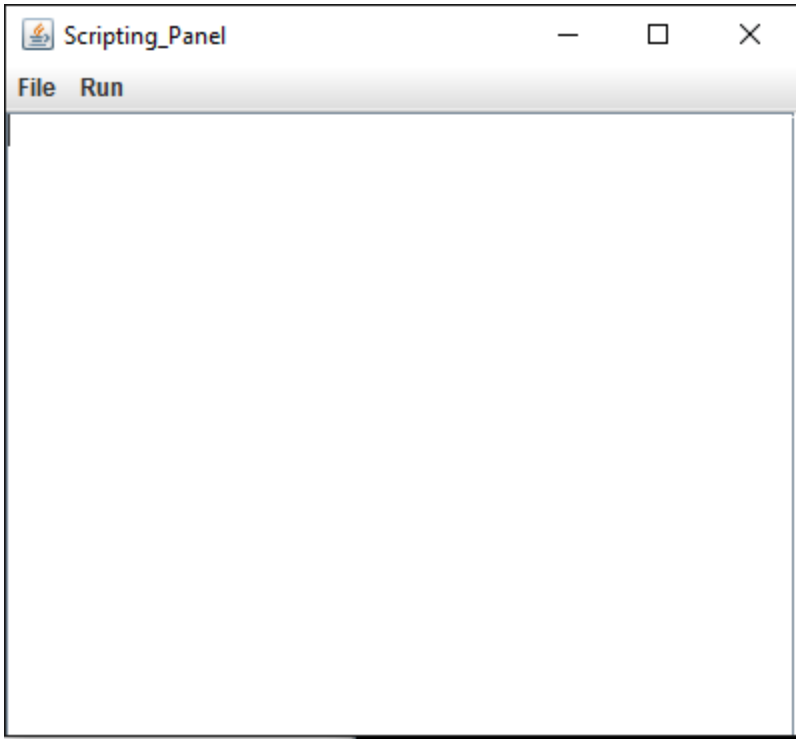



Fig. 4 Scripting Panel

## 3.2 Data Inputs

### 3.2.1 Images

New image can be opened using the terminal. There are two ways by which the images can be opened. One is by using the open under File menu and other is by using command in the terminal.

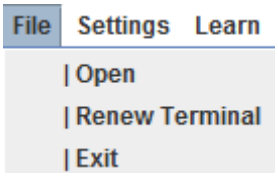



Fig. 5 Using Open under File to open image


Fig. 6 Using command to open image

### 3.2.2 Text

The text is used to create commands, the texts are given as streams of text then later converted to useful commands. The text area of the terminal is used for taking text streams as input.

### 3.2.3 Voice

We can also use voice as a commanding or input source. To use voice commands the menu bar of the program is used to activate voice listener program. After taking the inputs it is matched with grammar file and then passed to the control engine as commands.

## 3.3 Control Engine

### 3.3.1 Error Checking

Whenever a command is passed from terminal, the engine matches the text and determines if the command is valid or not. Now the way it matches is, it checks the command given or passed by the terminal with the commands that are allowed. If it finds matching problem it will generate error.

For the voice, the engine checks and also matches with the grammar file. If match is found it is passed accordingly to the command otherwise error is generated.

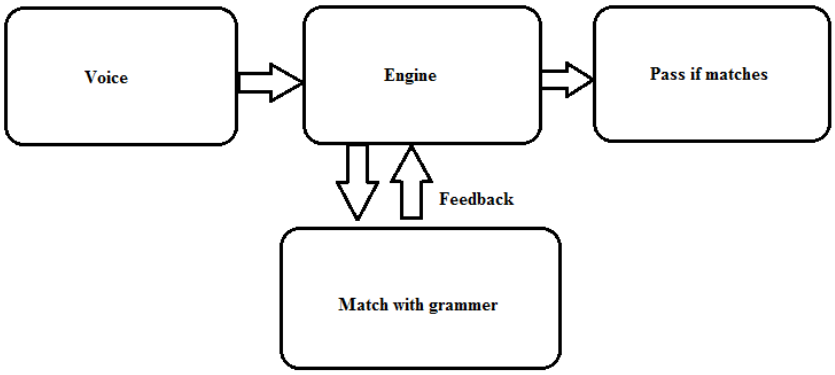



Fig. 7 Engine checking with grammar file

### 3.3.2 Editing Decision Making

After the error checking phase, when the command is approved then comes decision making process. The engine investigates the command and decides whether the command is about settings, filter or attribute changes, then sends it to the selected class accordingly to get the image processed.

## 3.4 Settings Class

Settings class is the class that controls different settings of an image, whenever this class is called or knocked according to the engine's determination it implements functionality to the image. Setting class has mainly four function they are crop, opacity, sharpness, contrast.

### 3.4.1 Crop

Crop is a function that takes two points and height and width as parameters then crops the original image according to the parameters.

Algorithm 3.4.1 :

***Procedure Crop (x, y, height, width)***

***Start :***

***Mark pixel (x. y)***

***Form Area (height, width)***

***Cut Image***

***End***

### 3.4.2 Opacity

Opacity function gets a value from the user and then implements the transparency.

**Algorithm 3.4.2 :**

***Procedure Opacity ( value )***

***Start :***

***For [every pixel in width]***

***For [every pixel in height]***

***Apply opacity ( value )***

***End***

### 3.4.3 Sharpness

This function implements sharpness to an image according to user's desire. The function gets the value from user and then appends it to the image.

**Algorithm 3.4.3 :**

***Procedure Sharpness ( value )***

***Start :***

***For [every pixel in width]***

***For [every pixel in height]***

***Apply Sharpness ( value )***

***End***

### 3.4.4 Contrast

Contrast function takes value from the user and then changes contrast of an image.

**Algorithm 3.4.4 :**

***Procedure Contrast ( value )***

***Start :***

***For [every pixel in width]***

***For [every pixel in height]***

***Apply Contrast ( value )***

***End***

## 3.5 Filter Class

Filter class is used to implement different filters on the image. It can convert the image to different filters such as black and white, gray, invert and others. Whenever the engine assigns task taken from terminal to the filter class it works accordingly to the command given or passed.

### 3.5.1 Red Scale

This function usually takes an image and overrides the image's every pixel with only the red values and nulls values of blue and green.

**Algorithm 3.5.1 :**

***Procedure Red Scale ()***

***Start :***

***For [every pixel in width]***

***For [every pixel in height]***

***Get RGB values of pixel***

***Select Red***

***Null Green, Blue***

***Set RGB value to pixel***

***End***

### 3.5.2 Blue Scale

This function works just like red scale except for the fact it overrides with blue values and nulls the value of green and red.

**Algorithm 3.5.2 :**

***Procedure Blue Scale ()***

***Start :***

***For [every pixel in width]***

***For [every pixel in height]***

***Get RGB values of pixel***

***Select Blue***

***Null Green, Red***

***Set RGB value to pixel***

***End***

### 3.5.3 Green Scale

The green scale function is exactly like blue and red scale but instead it implements only the green value.

**Algorithm 3.5.3 :**

***Procedure Green Scale ()***

***Start :***

***For [every pixel in width]***

***For [every pixel in height]***

***Get RGB values of pixel***

***Select Green***

***Null Red, Blue***

***Set RGB value to pixel***

***End***

### 3.5.4 Invert

The maximum value for red, green and blue can be 255, the invert function subtracts the value of red, green and blue from 255 and implements it over the image.

**Algorithm 3.5.4 :**

***Procedure Invert ()***

***Start :***

***For [every pixel in width]***

***For [every pixel in height]***

***Get RGB values of pixel***

***Red=255-Red, Green=255-Green, Blue=255-Blue***

***Set RGB value to pixel***

***End***

### 3.5.5 Gray

The gray function takes all the values of red, green and blue and averages the values then appends the value to the pixel

**Algorithm 3.5.5 :**

***Procedure Gray ()***

***Start :***

***For [every pixel in width]***

***For [every pixel in height]***

***Get RGB values of pixel***

***Average=(Red + Green + Blue)/3***

***Red=Average, Green=Average, Blue=Average***

***Set RGB value to pixel***

***End***

**3.5.6 Black And White**

This function works almost like gray but checks whether the average value is greater than (255/2) or less. If greater than sets red, green and blue to maximum(255) else sets to minimum(0).

Algorithm 3.5.6 :

***Procedure Black And White ()***

***Start :***

***For[every pixel in width]***

***For[every pixel in height]***

***Get RGB values of pixel***

***Average=(Red + Green + Blue)/3***

***If Average>(255/2)***

***Red=255, Green=255, Blue=255***

***Else***

***Red=0, Green=0, Blue=0***

***Set RGB value to pixel***

***End***

## 3.6 Attribute Class

Attribute class creates the opportunity to add extra features like text, drawings, watermarks on the image. The idea of making this class is to extend the general functionality of the image. We didn't have to develop any algorithms for this class. We directly used built in functions of Java environment.

### 3.6.1 Write Text

Write text, function is used for writing text over an image, this is helpful to add captions or description to the image.

### 3.6.2 Fill Rect and Draw Rect

These two functions are used to draw the outline or fully filled rectangles over the image. These are really simple functions. The idea from these two functions can be extended to create circles, triangles and other shapes on an image.

### 3.6.3 Watermark

Sometimes we need to add watermarks to an image, this function helps to put watermark logo or signatures to the image.

# Chapter 4

## Experimental Setups and Results

## 4.1 Experimental Setups

The text streams didn't need any extra setup we just converted the text streams to useful commands. But for the voice command process we have used extra libraries to support the system and to recognize the voice or commands spoken by the user.

### 4.1.1 Text To Command Conversion

Whenever a user gives a command, at first we have extracted the words from the sentence and then created an array of words from the given sentence.

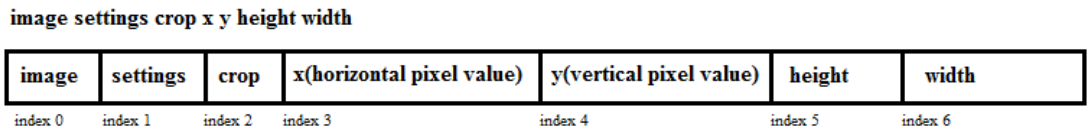



Fig. 8 Array of words from Text Streams

Then from index 1, we can say the type or the category of edit or image processing the user want. After that we look up further to index 2 and check whether the category determined from index 1, has the function or sub-category. If it has the function then we look up further if not then the program generates error. As we go along we keep matching the array index with the functions and given data ( for example height, width and pixel information ) . The array is traversed until it ends, if any of the index is not matched while traversing then error is generated otherwise the words from the index keep going to a specific function with data. Fig 4.1.2 shows

different category and their sub-category or functions. From the starting index 1, it goes through the options then after entering to one it sees index 2 and then again goes to the option for index 1 after that it checks index 3 and proceeds and accordingly gets all the data to process image and generates the desired output.

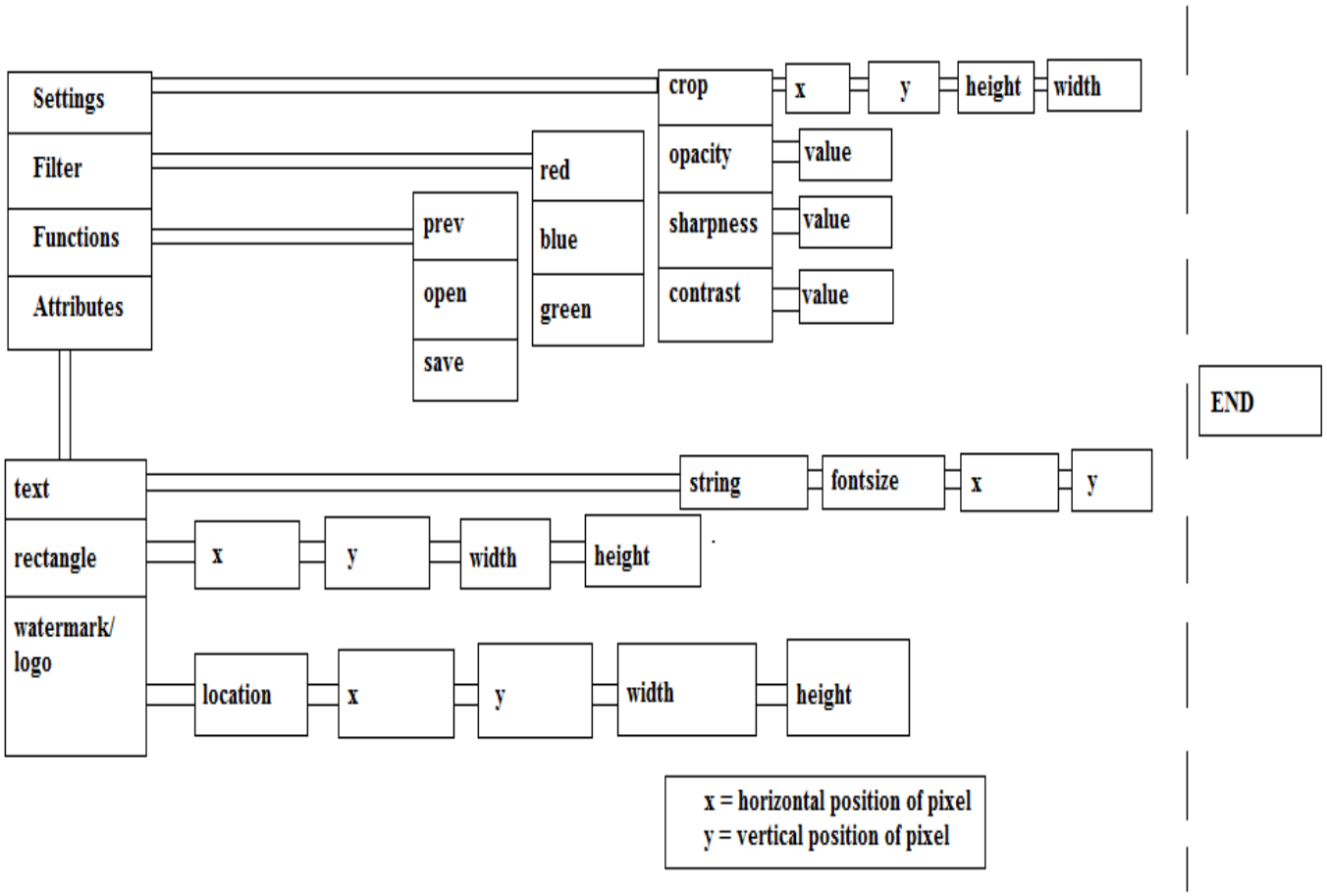



Fig. 9 Understanding the process of image from array

For example, if the users give “image filter green”, the program will look up index 1, which is filter, if found then it will look at the sub-category of filter, then if green found then it will apply the green filter to the image. If any of the category is not found the program will generate error.

### 4.1.2 Voice To Text Conversion

Whenever the user wishes to use voice as a source of command he can easily navigate to the menu bar and activate voice listener. The voice listener will turn on the microphone of a computer then use library to recognize voice, then the library will check if the spoken words are in grammar file, if found then the voice will be converted to text streams.

The libraries used for building this application are :

1. JSAPI ( Java speed API).
2. Sphinx ( Speech recognizing engine ).
3. Tidigits ( Sound Range determination )
4. WSJ ( Sound Range and Microphone controlling ).

The format for grammar file :

```
#JSGF V1.0;  //version declaration

grammar name;  //name of the grammer file

public<name of function> = (words) (words | words) (words)  // this is just a format for any function or sentence type
                                                            // many function can be given for different sentences formation


//example
//public <filter> = (Image)(Filter)(Black and White | Blue | Green | Red | Invert | Gray);
//public <draw> = (Image)(draw | fill)(Rectangle);
```

"//" is used for comments

Fig. 10 Example of grammar file

After all the process of voice to text conversion, the text is then send to terminal and from there the terminal passes the text stream to the engine and from there the text is converted. The conversion is explained at [section 4.1.1].

## 4.2 Results

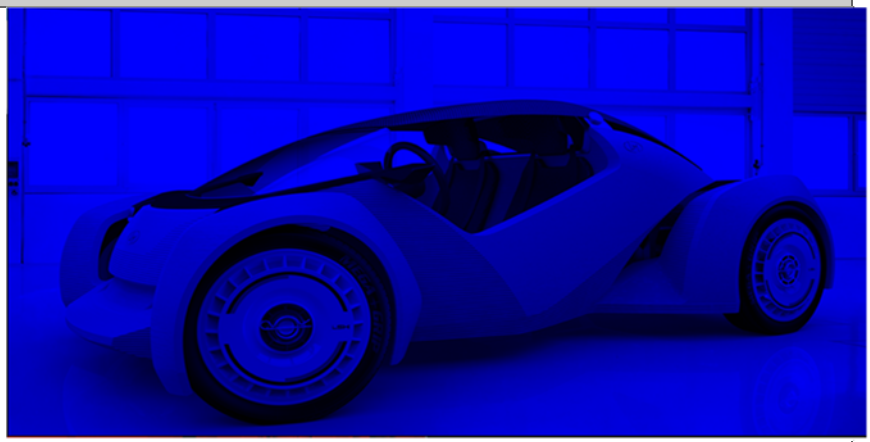
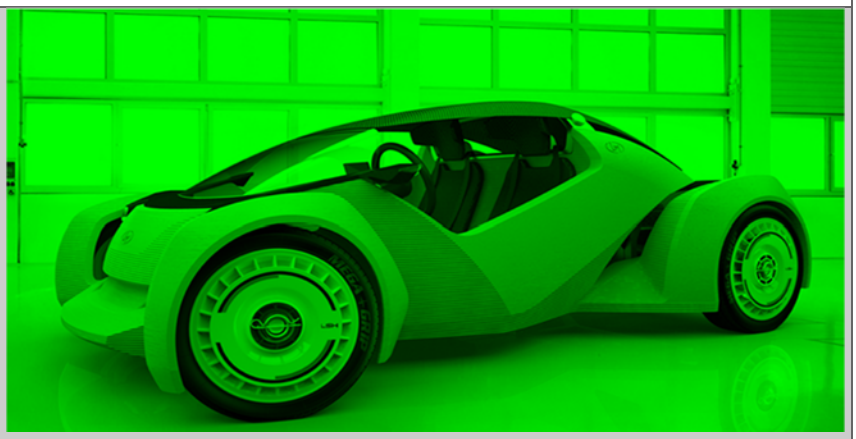
The results are shown with respect to the commands. Every command gives different output. Every command calls different algorithms to write pixels or change them.

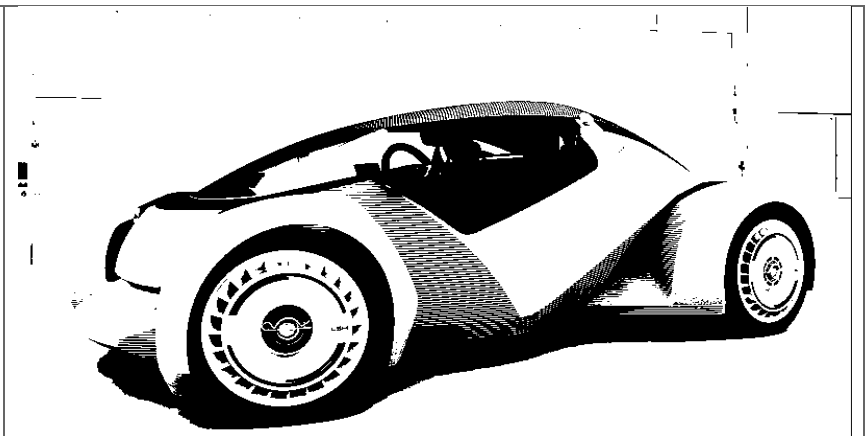
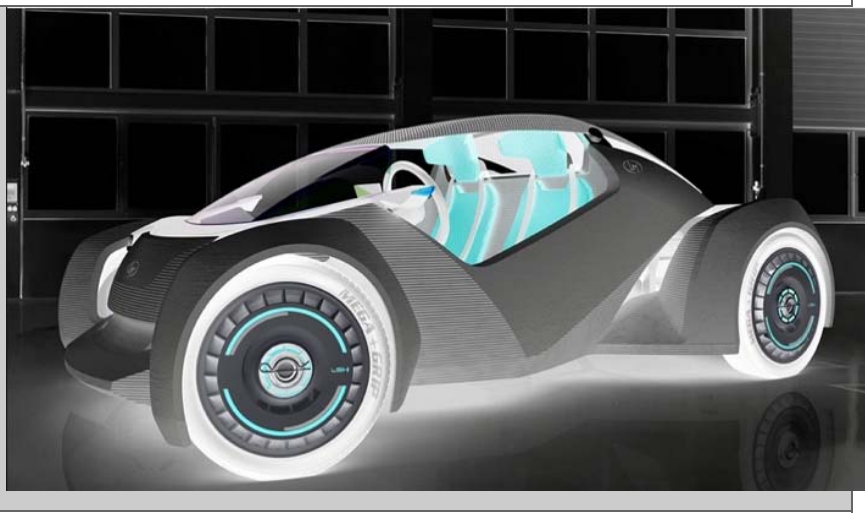
### 4.2.1 Original Image

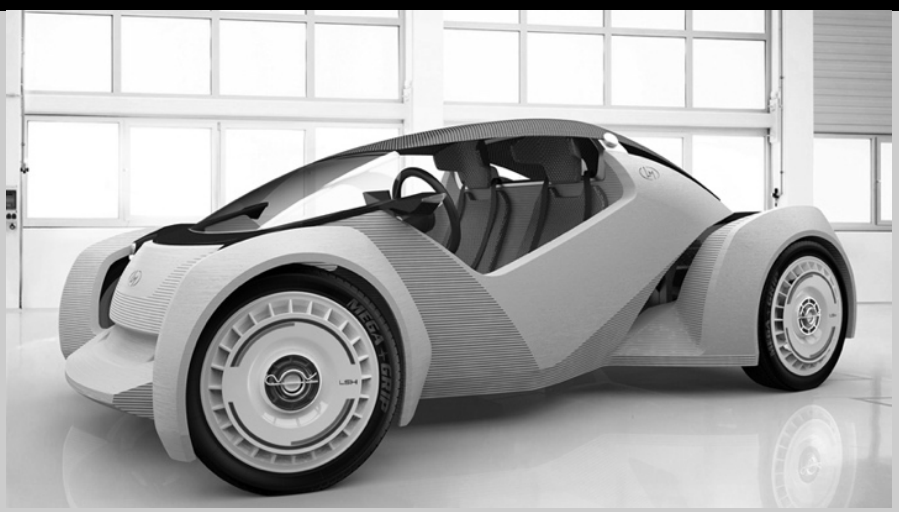
Fig. 11 Original Image

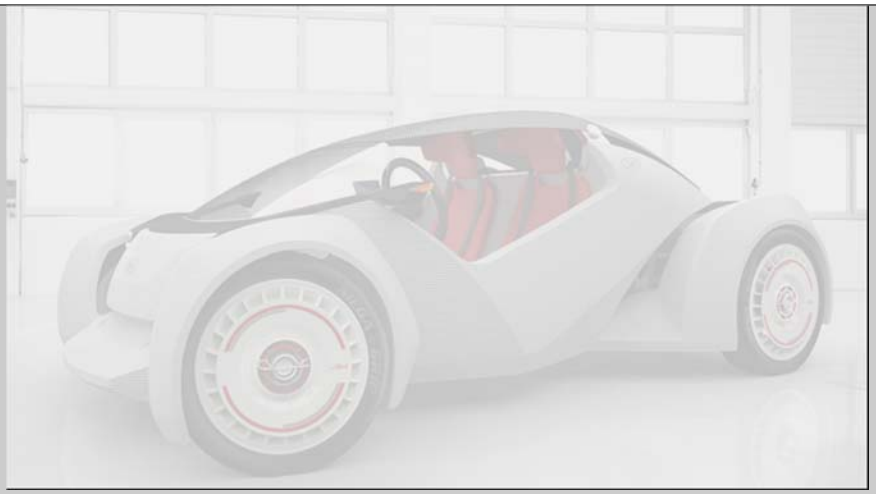
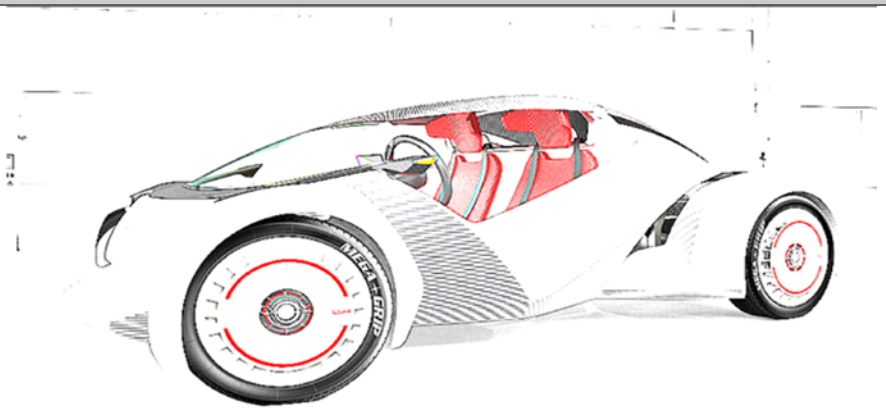
### 4.2.2 Processed Image

A table containing original images and processed images after using different types of filters is given below :

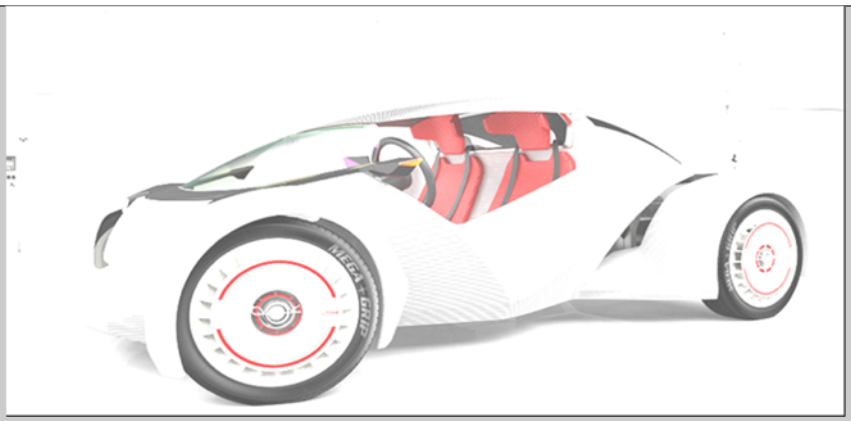
| Filter Name | Command | Processed Picture |
|---|---|---|
| Red | **image filter red** |  |
| Blue | **image filter blue** |  |
| Green | **image filter green** |  |

| | | |
|---|---|---|
| Black and White | **image filter bw** |  |
| Invert | **image filter invert** |  |

| Filter Name | Command | Processed Picture |
|---|---|---|
| Gray | **image filter gray** |  |

| | | |
|---|---|---|
| Crop | **image settings crop 100 100 100 100** |  |
| Opacity (Dropped To 50%) | **image settings opacity 50** |  |
| Sharpness | **image settings sharpness 10** |  |

| Filter Name | Command | Processed Picture |
|---|---|---|

| | | |
|---|---|---|
| Contrast (Increased) | **image settings cont 10** |  |
| Text | **image attr wtext car 12 100 100** |  |
| Watermark | **Image attr wmark D:\\art_gal\\bee.png 100 100 100 100** |  |

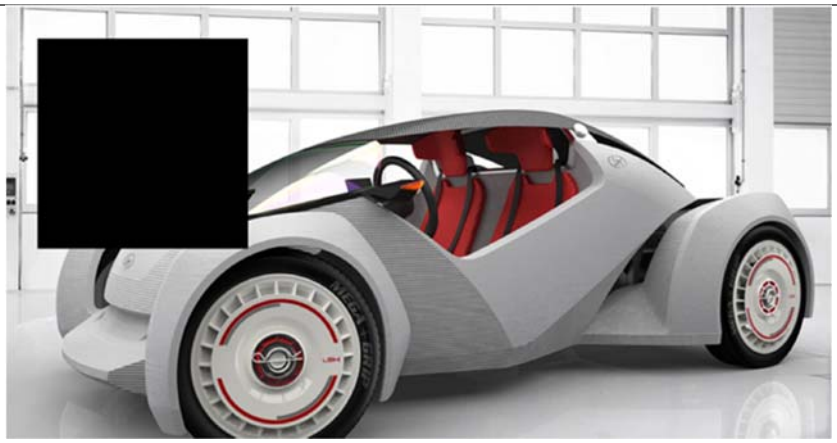
| | | |
|---|---|---|
| Rectangle Drawing | **Image attr frect 30 30 200 200** |  |

Table 1. Original Image and Processed Image

## 4.3 Scripting Result

Scripting is an important part of any terminal. Scripting add power to the terminal and makes it easy for the user to do a lot of tasks by adding commands. The advantage of scripting in our terminal is, we can add multiple functions over an image, also use the same script to edit multiple images.

### 4.3.1 Example

Let's consider that we have a white image in our 'D' directory. This directory also has a folder that contains some image file and the name of the folder is **art_gal** [art gallery]. The images are given below.

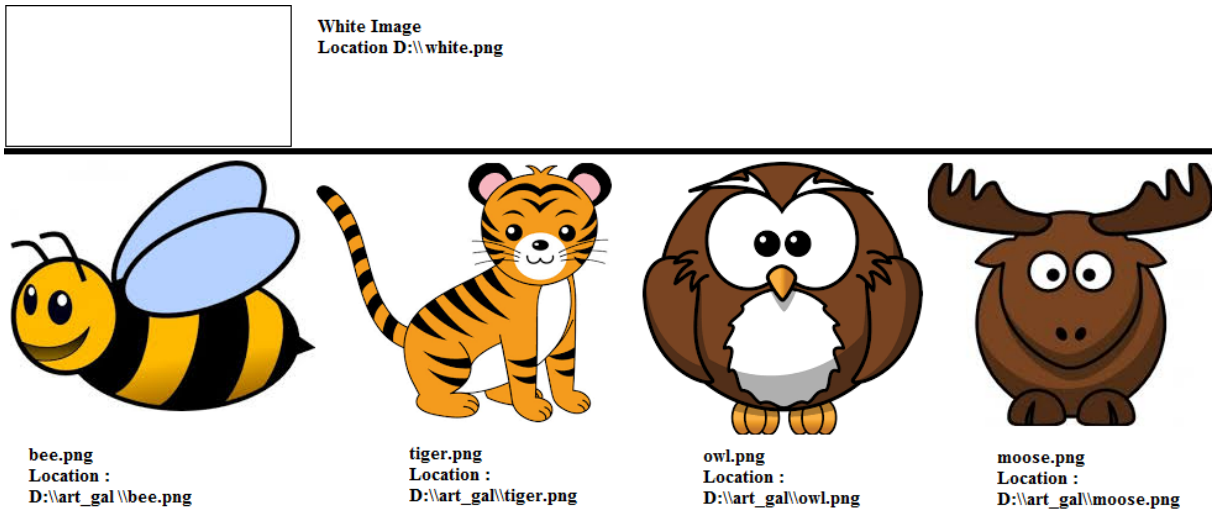



Fig. 12 Directory wise images

Let us now write a script and run it.

Script :

*image attr wmark D:\\art_gal\\bee.png 100 100 100 100*

*image attr wmark D:\\art_gal\\tiger.png 220 100 100 100*

*image attr wmark D:\\art_gal\\moose.png 340 100 100 100*

*image attr wmark D:\\art_gal\\owl.png 460 100 100 100*

*image attr wtext bee 12 100 220*

*image attr wtext tiger 12 220 220*

*image attr wtext moose 12 340 220*

*image attr wtext owl 12 460 220*

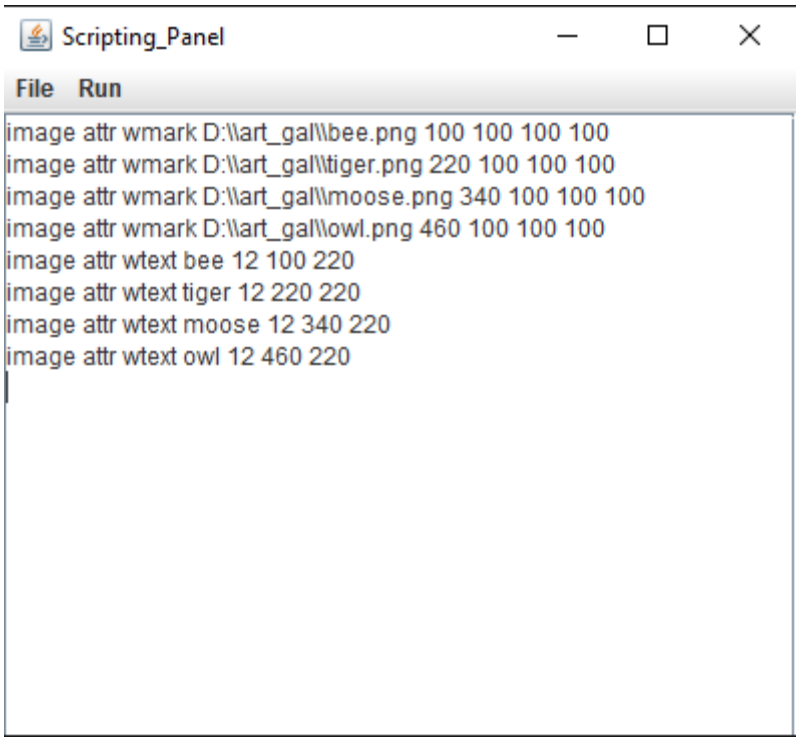



Fig. 13 Script Writing

Image before running the script :

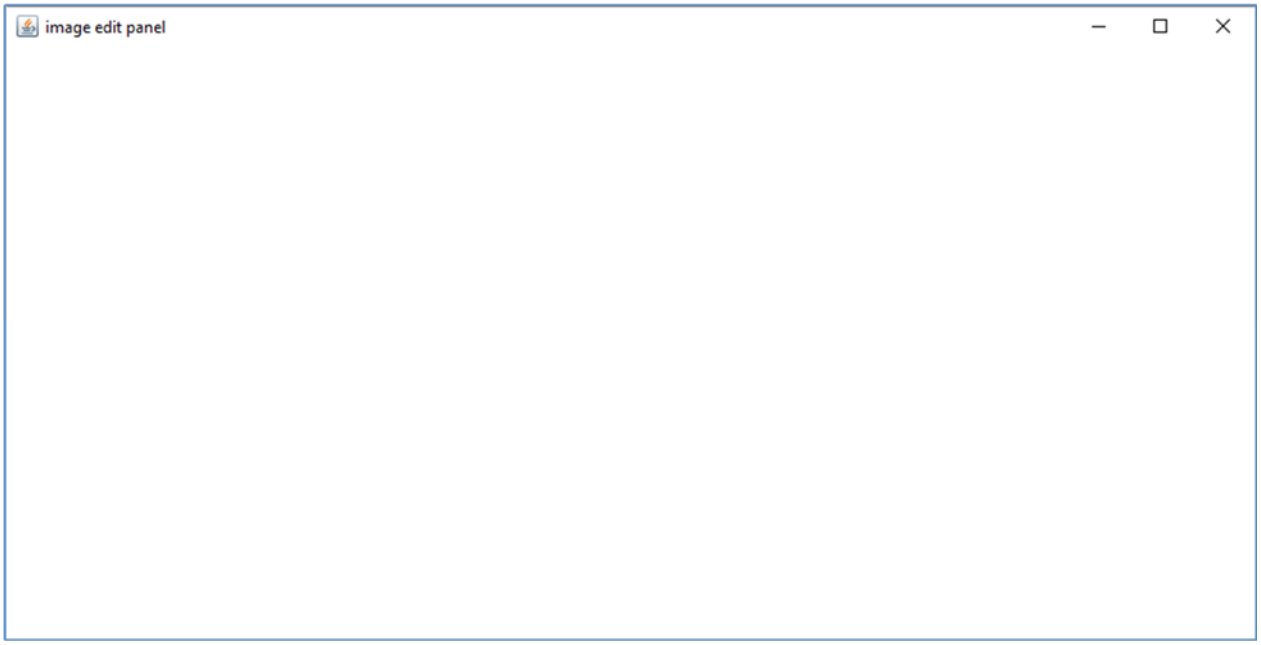



Fig. 14 White Image

Image After running the script :

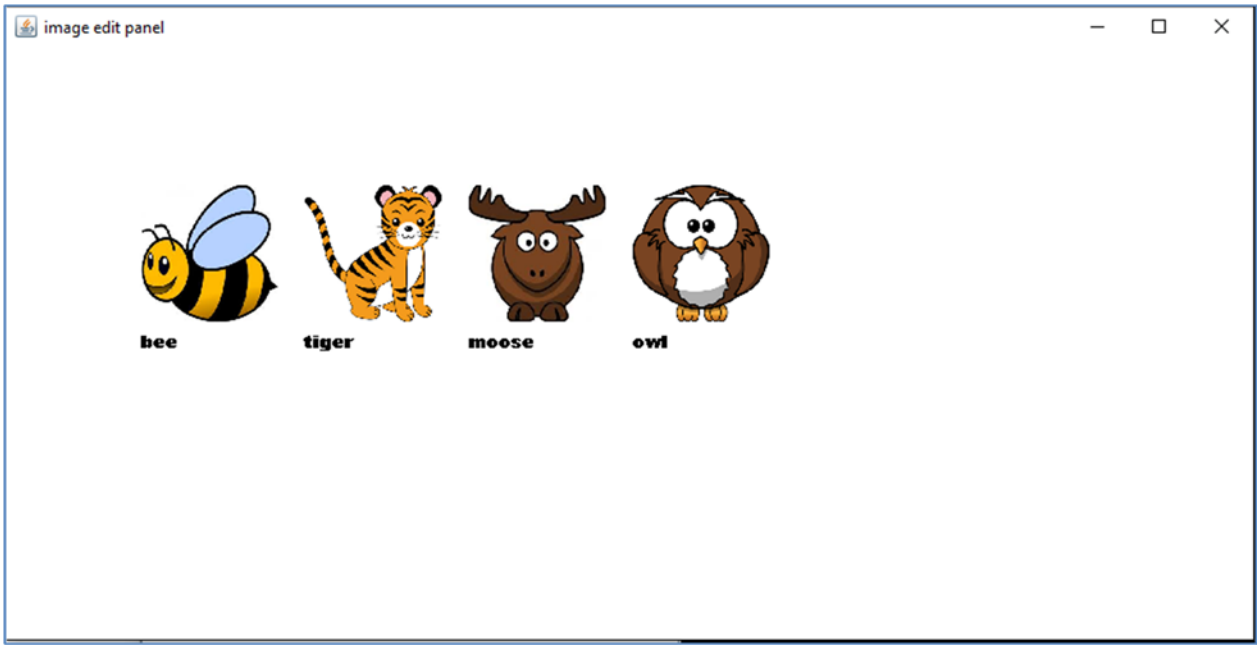



Fig. 15 Image After Scripting

The demonstration was just a simple work, but scripting can be used to make powerful and complicated image processing.

For experimenting the terminal, a few techniques have been added. A various range of image processing can be added to the terminal in future.

# Chapter 5

## Future Work and Conclusion

## 5.1 Future Work

Image processing through terminal using voice and text commands is a unique idea that has been developed to reduce the complexity of image processing systems. The proposed method takes normal human language as input and converts it to useful commands to process images. So far the method has been implemented with a very few examples. The concept has far reaching implementations that might not be visible yet. In this section we are going to explain how and where this idea can be used.

### 5.1.1 Internet Technology

Our conventional web system uses html as a markup language, but if we add systems like scripting to the terminal it will be possible to create an extended version of the script and use it as markup language. Best suitable use for this program will be creating simple page websites and designs.

### 5.1.2 Graphics Creation

We have shown results on images in the previous chapter, but we can also create high graphics from scratch, alternatively merging different graphics to a single layer graphics. Using the terminal will allow to create logos, posters and other graphical effects on images, movie files. We will also be able to create animations by using framing technology.

### 5.1.3 Research

Using the terminal is easy and simple. So far a few features have been added to demonstrate the idea, but in future after adding many more features it will be possible to use the terminal as an extensive tool for research.

### 5.1.4 Live Image Broadcast

The terminal can be used as a live image or video streaming station if we can merge it with internet. The idea is now well popular as we use it in Facebook and Youtube live. But the terminal will give it a lot more power and it will be easier to create stations and really inexpensive.

### 5.1.5 Robotics

If we add more intelligence to the program it will essentially work as a robot. Which will take instructions from humans and then interpret it to useful commands. For example, if a person asks the robot to pick up an apple it will be able to understand the language and get the job done by simply getting information from its camera.

### 5.1.5 Automation

Automation of cars and other vehicles is also possible if we extend the terminal and modify it accordingly, whereas in the terminal now takes natural language as input it will take information from the image and take decision accordingly.

### 5.1.6 Parallel Computing

The terminal can be extended to form a program like SSH, we can use the same image processing over a huge computers connected to a network. Also in space research or other area

that uses or works with high resolution computers, rather than using the super computers we can crop the original image and pass it to different computers and process with same algorithms, which will lower the cost of research and development.

### 5.1.7 Voice Controlled Operating Systems

Just like the earlier terminal has evolved to an operating system, the terminal developed, can also be used to create a voice controlled operating system.

There are other ways we can use the terminal, as image processing is a very highlighted and useful mechanism, the terminal makes the concept easier. Thus it has a lot of potential to power the future world image processing and other technology.

## 5.2 Conclusion

Terminal technology is a very old technology but extra powerful. Since modern image processing is very much complex, we have tried to make it easier by adding terminal technology. Although the working procedure of the terminal is almost like other terminals available in operating systems but our terminal takes natural language either by voice or text as input which is an advantage for the general user. As we are marching towards complex digital systems, it can be understood that image processing holds so much importance at present and in the upcoming technology. The focus of this terminal is to make a very complex part of computer science known as image processing, simple and easy. Adding voice and simple language of user makes it easier. Image processing through terminal is a very small idea for now but it is unique in its nature. This idea can be extended to use in other fields such as AI, machine learning, human computer interfacing, robotics, automation and much more. Terminal is powerful itself when it comes to comparing with graphical user programs, so far engineers have used the terminal to

create and use complicated programs. The proposed system makes the terminal even more powerful since it works with graphics and image processing.

At last, the terminal might sound a very complex program, but it is really very easy since it works on simple human language. It also reduces the stress of image processing. By upgrading the terminal it is possible to solve more complicated problems of image processing. The development so far thus have so much potential that will be revealed in the future.

# References

[1] Lukáš KRASULA, Miloš KLÍMA, Eric ROGARD, Edouard JEANBLANC , MATLAB-based Applications for Image Processing and Image Quality Assessment Part II: Experimental Results

Available : https://www.radioeng.cz/fulltexts/2012/12_01_0154_0161.pdf

[2] P Lelong, G Dalm, J Klijn(1995) - US Patent 5,444,478,

Image processing method and device for constructing an image from adjacent images

[3]Maria Nazarkevych,Bogdan Yavourivskiy,Ivanna Klyuynyk,Editing raster images and digital rating with software

[4] Anastasia Dubrovina,Rom Hershkovitz,Ron Kimmel,Image editing using level set trees

[5]Ryouhei Maeda,Tsutomu Maruyama, An implementation method of poisson image editing on FPGA

[6] C. Ballester, M. Bertalmio, V. Caselles, G. Sapiro, J. Verdera (2001), IEEE, Transactions on Image Processing, vol. 10, no. 8, pp. 1200–1211.

Filling-In by Joint Interpolation of Vector Fields and Gray Levels.

[7] F. Meyer, J.Histoch.Cytochem (1979), vol.27, pp. 128–135,

Iterative image transforms for an automatic screening of cervical smears.

[8] R.Klevansand R.Rodman,(1997),

Voice Recognition, Artech , Boston, London,

[9] H.Sakoe and S.Chiba, (1978) IEEE Trans. ,Acoustics, Speech, Signal Proc.,ASSP-26(1),

Dynamic programming algorithm optimization for spoken word recognition .

Availabe:https://www.researchgate.net/publication/49587837_A_Review_on_Speech_Recognition_Technique [accessed Dec 12 2017].

[10] .M.A.Anusuya , S.K.Katti, Information Security (2009),

Speech Recognition by Machine: A Review International journal of computer science and A Review on Speech Recognition Technique

Available:https://www.researchgate.net/publication/49587837_A_Review_on_Speech_Recognition_Technique [accessed Dec 12 2017].

[11] Ehsani, F., & Knodt, E. (1998) Language Learning &Technology, 2(1), 46-60,

Speech technology in computer-assisted language learning: Strengths and limitations of a new CALL paradigm.