TABLE OF CONTENTS

Topics **Page No.**

INTRODUCTION

Spam can be defined as unsolicited bulk email. According to Symantec , during 2015 spam accounted for approximately 60% of all inbound email and, consequently, spam filtering is essential for the continued viability of email communications. We use the term “ham” to distinguish legitimate messages from spam. Thus the problem of detecting image spam is that of distinguishing between ham and spam images.

TEXT SPAM:

In this project for text spam detection we have detect polarity of sentence to detect if it is spam or ham**[1]**.Using sentiment analysis we detect the polarity of a sentence and if the polarity of a particular sentence is less than 0, then that particular sentence will consider as spam, if the polarity is greater than 0 we will consider it ham.

IMAGE SPAM:

In this project for image spam detection we are basically considering a naked image and trying to flag it as spam image.We have done sobel and canny edge detection analysis to get the edges of a image.On the other side we are cropping the face of the human shown in the image and applying sobel and canny edge detection**[2]**.Now we are calculating the pixel intensity of the cropped face after edge detection as well as the pixel intensity of the original image after edge detection.Then we are comparing the intensity of cropped image and original image,If the the amount of pixel intensity in cropped image is more than 70% in the original image then we flag the original image as spam.

PROPOSED WORK

We have used python language to implement both text and image spam.

TEXT SPAM:

We have used textblob**[3]** library to text processing.we managed to detect the polarity of a sentence,for example we took a csv file consist of spam and ham messages.we use a special function to detect polarity and if the polarity is greater than 0 ,calculation of spam message will increase by 1 and if the polarity is less than 0,calculation of ham message will increase by 1.At the end the we will get the result in a pie chart.

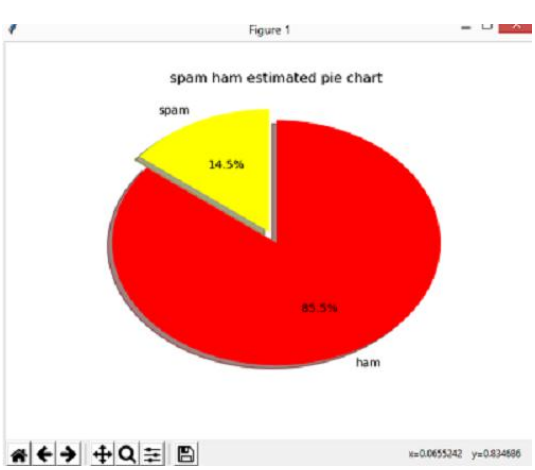
IMAGE SPAM:

We have used openCV**[4]** library to image processing.first we r converting a naked image to gray scale to maintain the equal pixel intensity throughout the image.Then we are applying sobel and canny edge detection to determine the edges of the original image.

On the side we have used haar cascade**[5]** tools to detect the face of the humans shown in the image and crop the detected face of the humans.

EXPERIMENTAL RESULT

TEXT SPAM:



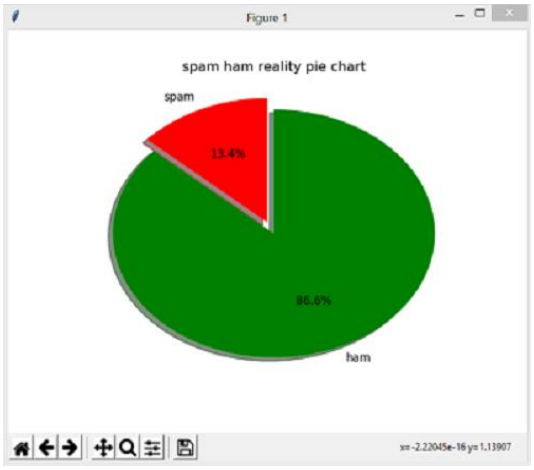
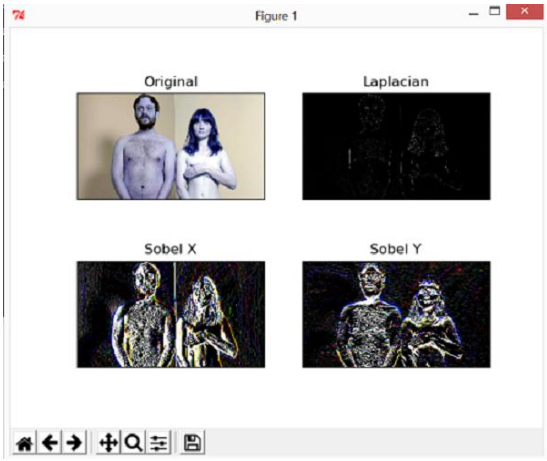
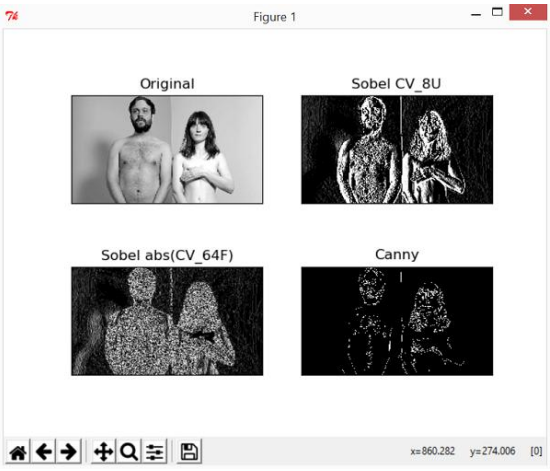
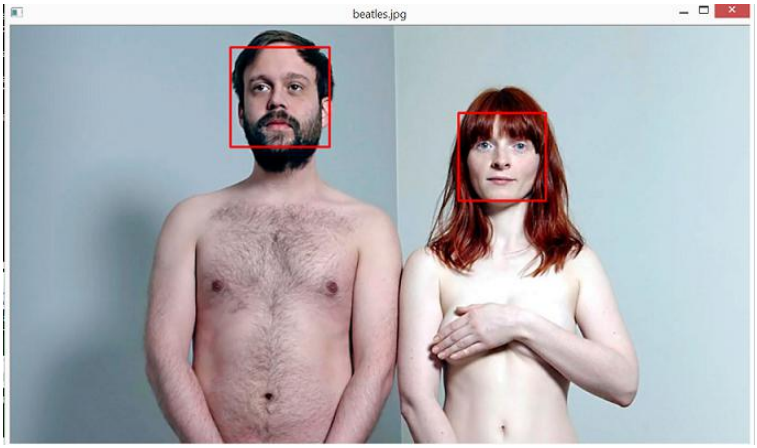


IMAGE SPAM:







CONCLUSION

On the whole, by the corner point density in images to filtering the spam images, the detection rate is relatively high. Our proposal method can be effective against spam images from spam mails. Combined with pre existing methods which have been used into practical application, we can effectively filter spam mails. Effectively identify the corner and conduct corner statistics is the key point in this experiment. In future researches, we strive to optimize the algorithm design of corner detection. We also hope to remove the noise around the outline of the binary image as much as possible in order to provide better image quality for corner point detection in next step. Because spam image`s style changes fast, newer modifications of spam image are already found in the span of our research. Consequently, anti-spam technology should be improved continuously. In next step, our work will focus Journal of Advances in text image spam detection.

REFERENCES

[1]*Spam and Ham* retrieved from http://www.openkod.com/wp-content/uploads/2015/06/Anti-spam.pdf

[2]*Edge Detection* retrieved from https://www.tutorialspoint.com/dip/concept_of_edge_detection.htm

[3]*Textblob* retrieved from https://textblob.readthedocs.io/en/dev/

[4]*OpenCV* retrieved from http://opencv-python-tutroals.readthedocs.io/en/latest/py_tutorials/py_tutorials.html

[5]*Haar Cascade* retrieved from https://pythonprogramming.net/haar-cascade-face-eye-detection-python-opencv-tutorial/