Abstract—In the past, researches have addressed this problem as a text classification or categorization problem. In this project, we design an anti-spam filter which uses visual features and parameters for Spam filtering. Spam emails have embedded text messages in images to get around text-based anti-spam filters. We analyzed large collection of data using various features and a few classifiers and compared the accuracies for each classifier.

I. INTRODUCTION

With increasing importance of email, unsolicited commercial email (also known as spam) has become a major problem. Typical approaches, for spam filtering were based on text classification employing various machine learning techniques. These text based approaches have achieved a remarkable accuracy in filtering spam emails.

However, there are two major limitations to these text-based approaches. First, spammers often use various tricks to confuse text-based anti-spam filters. Examples of these tricks are text obfuscation, random space or word insertion, HTML layout, and text embedded in images. Second, as the scale and capacity of the Internet continues to grow, the type of information in emails has become more diverse. The genre of email content has moved from text-only to multimedia-enriched. These limitations greatly reduce the effectiveness of existing text-based anti-spam filters.

The key issue behind these challenges is that the type of content in emails has expanded from text-based to visual-based or combinations of the two. Since visual information is becoming more prevalent in emails, it becomes increasingly necessary to use such information to achieve high accuracy for anti-spam filtering. There are several ways of using visual information, particularly images, in anti-spam filtering. The spam emails are studied to analyze the characteristics of the visual information in spam. One noticeable characteristic is the types of images used in spam emails. These images are usually artificially generated and contain embedded text (i.e. text boxes embedded into image files).

II. THE SPAM DATASET

The SpamArchive dataset was downloaded from the SpamArchive website [1]. Our dataset contains 928 spam images and 810 ham images (Natural Images) of type JPEG.

![Sample images from data set](image)

(a),(b) Spam Images
(c),(d) Ham Images

<table>
<thead>
<tr>
<th></th>
<th>Training</th>
<th>Testing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Spam</td>
<td>520</td>
<td>408</td>
</tr>
<tr>
<td>Ham</td>
<td>450</td>
<td>360</td>
</tr>
</tbody>
</table>

Fig. 2: Dataset
III. THE VISUAL-BASED ANTI-SPAM FILTER

For each email, we extracted a set of features from the images contained in the email. The set of features was then used for classification, with one-class Support Vector Machines (SVM) and K-Nearest Neighbour classifier (KNN) being used as the base classifiers. In the following subsections, we discuss in detail various features used for classification. The features used are:

- Color Histogram with Graphic Features
- Embedded Text Feature
- Gradient Of Histograms

IV. FEATURE EXTRACTION

A. Color Histogram with Graphic Features

Spam images are artificially generated, so the expected image texture statistics are distinguishable from natural images such as sky, mountain, beach, buildings, and human. The observation is that most of spam images are converted from text spams, although they may contain some icons and artifacts. Thus, the color components may be quite limited compared with natural scenes. The color histograms of natural scenes tend to be continuous, while the color histogram of artificial spam image tends to have some isolated peaks. Our interest is only in shape and color so only dominant bins are retained.

Each color can be represented by 2 independent components, so we build 2-D color histograms in certain color space (normalized RG space). We employ normalized RG space in color histogram calculation, which is insensitive to lightings. Since we only care about the shape or color distribution rather than the exact meaning of color bins, we sort the bins in descent order and only keep the top dominant D bins as the feature vector $V_c$. This approach also balances the need for high resolution of color histogram (otherwise similar color will be quantized to the same bin), and the need for efficient training and testing on feature vectors without too high dimensionality. In our experiments, we calculate $32 \times 32 = 1024$, 2-D color histogram and test keeping different top $D = 32, 64, 128$ bins.

Figure 3 and Figure 4 show an example of Color Histograms for Ham and Spam images respectively.

B. Embedded-text Feature Extraction

Spammers are now embedding text messages in images to bypass through text-based anti-spam filters. To detect such devious techniques, it would be helpful to know:

(1) whether there is embedded text in the images,
(2) if so, the area of text regions vs. the total image area.

To derive such information, we have used a text-in-image feature detector which is capable of detecting the text region(s) in an image. The details of the detector will be described later. We use this text-in-image detector to scan through each image in the email and derive the pixel count ratio of the detected text regions to that of the overall image area.
Extraction

In the past, several text detection and text-recognition methods have been used to detect text boxes in grayscale documents, newspapers, and video frames. Previous text-detection methods typically followed a multi-step framework using a combination of image analysis and machine learning techniques.

Since text characters usually have consistent color, regions of similar intensities are found in the image using the MSER (Maximally Stable Extremal Regions) region detector. Some of these regions include extra background pixels. As the written text is typically placed on clear background, it tends to produce high response to edge detection. Furthermore, an intersection of MSER regions with the edges is going to produce regions that are even more likely to belong to text. Since the original MSER regions also contain pixels that are not part of the text. The edge mask along with edge gradients is used to eliminate these regions. Some of the remaining connected components can now be removed by using their region properties. Another useful discriminator for text in images is the variation in stroke width within each text candidate. Characters in most languages have a similar stroke width or thickness throughout. It is therefore useful to remove regions where the stroke width exhibits too much variation. Most non-text regions show a large variation in stroke width. Therefore the non-text region can be filtered using the coefficient of stroke width variation. To compute a bounding box of the text region obtained, the individual characters are first merged into a single connected component. This is carried out using morphological closing followed by opening to clean up any outliers.

Figure 5 and 6 show an example of an email image with embedded text after each step for ham and spam image.

C. Gradient Of Histograms

The distributions of gradient orientation for natural images appear more uniform and noisy than those of spam images. Gradient orientation histograms are particular effective to deal with gray-level images.

To extract gradient orientation histogram $V_g$, the image gradient for each pixel is calculated with Sobel operator, if the gradient magnitude is larger then a threshold $tm = 50$, we quantize its orientation angle $0 ^\circ - 360 ^\circ$ to one of the $D$ (dominant) bins.

Figure 7 and 8 shows an example of ham and spam images with thier respective Gradient of Histograms.

V. Classification

In our anti-spam filter, we define the anti-spam filtering problem as a task of finding whether an unseen email falls into the spam class or ham class.

The dataset is divided into Training and Testing sets. The Training dataset has 520 spam Images and 450 ham Images. Testing dataset has 408 spam images and 360 ham images. The extracted features as described above are analysed using supervised and unsupervised classifiers for spam email filtering. Classifiers that are used to analyse are two-class SVM Classifier and KNN.
Fig. 6: Embedded Text Extraction Ham
(a) original Image, (b) MSER regions, (c) Canny Edges and Intersection of Canny Edges with MSER (d) Text candidates before and after region filtering (e) Extracted Text region

Classifier.

**SVM Classifier:**

Given a positive and a negative dataset, the SVM classifier maps the data from the input space to a higher dimensional space, called the feature space, and constructs a hyperplane in the feature space which separates the data with a maximal margin. We generated models that classify/separate ham/spam images.

For the above extracted features, we implemented a two class SVM classifier to analyse the effectiveness of the system.

**IV. RESULTS**

We trained a two-class SVM classifier using 520 spam images as our positive training set and 450 natural images as our negative training set. The testing dataset has 408 spam images and 360 ham images. The SVM package we used is LIBSVM. The accuracy we achieved is 90 % using SVM Classifier.

Using KNN Classifier for the same training and testing dataset, we got 84 % accuracy. We see that using two-class SVM classification we got a better accuracy compared to KKN classification.

**KNN Classifier:**

Instance-based classifiers such as the KNN classifier operate on the premises that classification of unknown instances can be done by relating the unknown to the known according to some distance/similarity function. Classification using an instance-based classifier can be a simple matter of locating the nearest neighbour in instance space and labelling the unknown instance with the same class label as that of the located (known) neighbour. This approach is often referred to as a nearest neighbour classifier. We used KNN classifier to analyse the effectiveness of the system.

**VI. RESULTS**

Using KNN Classifier for the same training and testing dataset, we got 84 % accuracy. We see that using two-class SVM classification we got a better accuracy compared to KKN classification.
VII. LIMITATIONS

We have encountered a few limitations during our project. Firstly, choosing an appropriate threshold to decide if the image has text regions or not, as the percentage keeps varying with the input image. Secondly, in SVM two-class classification, even though the detection rate is very high, the false-positive rate is also high. One reason for the high false-positive rate could be that the negative sets (for both training and testing) are not representative or diverse enough. In anti-spam filtering, due to privacy issues, a representative negative set is difficult to collect.

VIII. CONCLUSION AND FUTURE WORK

As the spammers techniques become more sophisticated, and the genre of email content continues to evolve, the text-based approaches alone are no longer sufficient for solving the spam problem. In our project, we analyze the spam emails containing images and identify a number of useful visual features that can be efficiently extracted from the emails for anti-spam filtering. The results clearly demonstrate that the proposed anti-spam filter can bring extra filtering power to existing text-based anti-spam filters.

This work focuses on two class classification (ham vs spam), this can be extended to multi-class classification like social, commercial, spam, promotions etc.

ACKNOWLEDGMENT

This project would not have been possible without the continued guidance from our Professor Dr. Anoop M. Namboodri. We would also like to thank our mentor Neha Soorma for her valuable inputs and support.

REFERENCES


[2] Ching-Tung Wu and Kwang-Ting Cheng, USING VISUAL FEATURES FOR ANTI-SPAM FILTERING, University of California, Santa Barbara, CA, USA.


[5] Hrishikesh B. Aradhye, Gregory K. Myers, James A. Herson, Image Analysis for Efficient Categorization of Image-based Spam E-mail, SRI International Menlo Park, CA 94025, USA.