# USING VISUAL FEATURES FOR ANTI-SPAM FILTERING

*Ching-Tung Wu*[†]*, Kwang-Ting Cheng*[†]*, Qiang Zhu*[†] *and Yi-Leh Wu*[§]

[†]University of California, Santa Barbara, CA, USA
[§]VIMA Technologies, Inc., Santa Barbara, CA, USA
tonywu@cs.ucsb.edu, {timcheng,qzhu}@ece.ucsb.edu, ywu@vimatech.com

## ABSTRACT

Unsolicited Commercial Email (UCE), also known as spam, has been a major problem on the Internet. In the past, researchers have addressed this problem as a text classification or categorization problem. However, as spammers' techniques continue to evolve and the genre of email content becomes more and more diverse, text-based anti-spam approaches alone are no longer sufficient. In this paper, we propose a novel anti-spam system which utilizes visual clues, in addition to text information in the email body, to determine whether a message is spam. We analyze a large collection of spam emails containing images and identify a number of useful visual features for this application. We then propose using one-class Support Vector Machines (SVM) as the underlying base classifier for anti-spam filtering. The experimental results demonstrate that the proposed system can add significant filtering power to the existing text-based anti-spam filters.

## 1. INTRODUCTION

With the increasing importance of email and the incursions of Internet marketers, unsolicited commercial email (also known as spam) has become a major problem on the Internet.

Junk email has been recognized as a problem since 1975 [1]. It was not a serious concern until marketers began to flood the system, overtaxing the resources of Internet Service Providers (ISPs). Since the late '90s, several anti-spam filtering solutions have been proposed. In general, these approaches treat the email spam filtering problem as a text classification or categorization problem, employing various machine learning techniques to solve the problem. In [5], the authors proposed using a Naive Bayesian classifier to filter junk emails. Some researchers have also suggested using Support Vector Machines (SVM) [7] and decision trees [8] for this task. These text-based approaches have achieved a remarkable accuracy in filtering spam e-mails.

However, there are two major limitations to these text-based approaches. First, spammers often use various tricks to confuse text-based anti-spam filters [2]. 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. For one example, legitimate message-senders have added multimedia content, particularly images, to text-based emails to enrich the message. Even worse, spammers have learned to use images to mask deceptive email messages, as well as to confound text-based anti-spam filters by employing HTML-based tricks. The raw content of these spam emails may not make sense to a filtering program, but the hidden messages can be rendered visible when recipients open them.

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. Our research team has investigated ways of using visual information, particularly images, in anti-spam filtering. We studied the spam emails 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). In this work, we propose a novel anti-spam filter, which classifies multimedia-enriched emails based on their visual information. Specifically, the proposed anti-spam filter extracts image features and uses a one-class SVM classifier to decide whether an unseen email is in the spam category. Experimental results show that, for emails containing image data, the proposed anti-spam filter can achieve a detection rate of 80% or more with less than 1% false positives. In addition, the proposed anti-spam filter can work with existing text-based filters. In comparison with the Bayesian text-based filter used in Thunderbird (a mail client of Mozilla [3]), our proposed filter can improve the spam detection rate from 47.7% to 84.6% for the validation set derived from the SpamArchive dataset [4].

The main contributions of this paper are summarized as follows:

- We introduce the use of visual information for anti-spam filtering. By thoroughly analyzing a large collection of spam emails, we demonstrate that useful features and parameters can be derived from images in spam emails for the purpose of anti-spam filtering.
- Based on such visual features and parameters, we propose a novel visual-based anti-spam filter. The proposed filter, used in conjunction with existing text-based filters, can improve the filtering accuracy.
- We have successfully integrated the proposed anti-spam filter with Thunderbird and demonstrated very promising results.

The rest of the paper is organized as follows: In Section 2, we present the statistics of some visual parameters from a thorough analysis of more than 120K spam emails downloaded from SpamArchive [4]. In Section 3, we present the proposed filtering system in detail. Then, in Section 4, we present experimental results to show the effectiveness of the proposed anti-spam filter.

## 2. THE SPAM DATASETS AND ANALYSIS

We prepared the following datasets for analysis and experiments:

- SpamArchive dataset: 122,877 spam emails.
- Ling-Spam dataset: 2,412 legitimate emails from Linguist mailing list and 481 spam emails [6].

0-7803-9134-9/05/$20.00 

The SpamArchive dataset was randomly downloaded from the SpamArchive website. The total number of downloaded and processed spam messages is 122,877. The Ling-Spam dataset is a public anti-spam filtering corpus [6]. For this work, we used the Ling-Spam dataset to train the Bayesian anti-spam filter in Thunderbird. Using these two datasets and the trained Bayesian anti-spam filter, we planned to show experimentally the limitations of text-based anti-spam filters, and also to demonstrate the additional power they can gain with our visual-based anti-spam filter.

In the SpamArchive dataset, 37.76% (46,395/122,877), of the emails contained images. Among those emails containing images, only 43.72% (20,283/46,395) contained accessible images. Many emails did not have images explicitly attached, but they provided links to the images. A fraction of these links were no longer accessible at the time we processed them. We further analyzed these 20,283 spam emails. The statistics are shown in Table 1.

**Table 1**. Statistics of emails containing accessible images

| Type | no. of Emails | Percentage |
|---|---|---|
| w/ image with embedded-text | 16,849 | 83.07% |
| w/ banner or graphics | 19,868 | 97.95% |
| w/ external image | 19,813 | 97.68% |
| blocked by Bayesian filter | 9,420 | 46.44% |

The results shown in Table 1 clearly indicate that if a spam email contains images, the images are likely to be artificially generated and external (i.e. not explicitly attached to the email). These images may be banners/graphics or text boxes embedded within the images. Figure 1 shows an example of a computer-generated image with embedded-text regions. In the following section, we introduce the ideas behind our visual-based anti-spam filter.

## 3. 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) being used as the base classifier. In the following subsections, we discuss the two major components, the features and the classifier, respectively.

### 3.1. Feature Description

Based on the observations summarized in Section 2, we define three sets of features as follows:

- Embedded-text features
- Banner and graphic features
- Image location features

More and more frequently, spam emails are embedding text messages in images to get around 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 developed a text-in-image 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 following embedded-text features: (1) the total number of text regions detected in all images in the email, (2) the percentage of images with detected embedded-text regions, and (3) the pixel count ratio of the detected text regions to that of the overall image area. Figure 1 shows an example of an image with identified text regions.

Many of the images in spam emails are banners and computer-generated graphics which are part of advertisements. We have developed a banner detector and a graphics detector. Banner images are usually very narrow in width or height. Also, banner images usually have a large aspect ratio vertically or horizontally. Graphic images, however, usually contain homogeneous background and very little texture.

Using these detectors, we can extract the following banner and graphic features: (1) the ratio of the number of banner images to the total number of images, and (2) the ratio of the number of graphic images to the total.

Spammers usually put their images behind web servers and create references in the emails to save server and network resources. This is in contrast to personal emails, where images are usually attached to the emails. We define the image location feature to be the ratio of the number of external images to the total number of images in the email.

### 3.2. Feature Extraction

**Banner and graphic feature extraction**. Since banner images carry certain geometric characteristics, they can be detected by using a very simple rule-based detector. In extracting banner features, we first use the rule-based detector to check the size and aspect ratio of images; then, we calculate the corresponding features according to the detected number of banner images.

Because computer-generated graphics usually contain homogeneous color patterns, they contain almost no texture in fine resolution. To extract graphics features, we first apply wavelet transformation on the input images. Then, we extract texture features in three orientations (vertical, horizontal, and diagonal) at fine resolution. If any of these extracted texture features falls below a predefined threshold, the image is likely to be a computer-generated graphic. We calculate the graphic features based on the detected number of graphic images.

**Embedded-text feature extraction**. In the past, several text-detection and text-recognition methods have been proposed to detect text boxes in grayscale documents, newspapers, and video frames [13, 14, 15]. Previous text-detection methods typically followed a multi-step framework using a combination of image analysis and machine learning techniques. Potential text regions were further refined as they passed through each step of the detection framework.

In [16], Viola and Jones proposed a face-detection framework using rectangle features and a boosting algorithm to train the cascade detector. In [12], we extended their idea and proposed a novel embedded-text detector for the task of text detection particularly in web and email images. We further enhanced the embedded-text detector by approaching the problem from the angle of object detection. We developed a unified detection framework, which was similar to Viola and Jones' work with two major differences: (1) We defined three sets of Position-Independent Features (PIF) to capture the essence of characters. Viola and Jones use *position-dependent* rectangle features, which can capture the differences in intensity among facial regions at particular positions. However, in the case of text detection, due to different types and sizes of fonts, the shapes of text are diverse and somewhat random. The three sets of PIF features are Local Edge-Pattern (LEP), Local Edge-Density (LED), and Global Edge-Density (GED). The LEP features are extracted by using a 3x3 mask and a binary coding scheme to translate edge patterns into decimal values. The LED and GED features capture the local and global distribution of edge points in images. The LED features are extracted by measuring the edge density around an edge point within

a 3x3 to 7x7 sub-window, whereas the GED features are extracted by measuring the edge density in different parts of the detection sub-window. The total number of PIF features is 509. These features are then selected and trained using the boosting algorithm to form the cascade detector. (2) We adopted a smart scanning algorithm to trace text lines using their spatial and geometrical regularities. Unlike faces, text objects appear as regions in images. Text objects in the same text line are close to each other (spatial regularity) and of similar size (geometrical regularity). These regularities can be used to locate text lines and to further improve the detection accuracy. The smart scanning algorithm starts with an initial sub-window size (12x10). It uses a sub-window to slide through the image vertically. Once a sub-window is detected as text, its neighboring sub-windows to the left and to the right are examined. This process is repeated until the entire image has been scanned. After each full image scan, the sub-window size is increased by 10% until it reaches the stopping size (72x60) or the image size. The raw detection results are post-processed and grouped to form the final text regions.

Figure 1 shows an example of an email image with embedded-text. The rectangles around the text lines show the detection result by our embedded-text detector. We calculate the embedded-text feature based on the detected text regions, such as the number and area of regions.


**Fig. 1**. Text embedded in image and detected regions

### 3.3. The Classifier

In previous approaches, the anti-spam filtering problem has typically been treated as a two-class or multiple-class classification problem. In the two-class case, researchers were trying to determine whether or not an unseen email was spam, whereas in the multiple-class case, the unseen emails were divided into several categories (such as commercial, financial, objectionable, health, spiritual, etc.).

One difficulty with the two-class and multiple-class classification is the need for multiple sets of training samples. For example, in the Naive Bayesian approach, one set of spam emails and one set of legitimate emails are required to train the classifier. While spam datasets are easily accessible, a representative set of legitimate emails is difficult to collect. 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. We propose using the one-class SVM [11] as the base classifier.

**One-class SVM**. The basic model of SVM[10] is a maximal margin 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. In [11], the authors extend the SVM to support one-class classification. Since there are no negative training samples, the basic idea is to find the hyperplane between the positive samples and the origin. In this case, the support vectors construct a probability-dense region which encompasses the training data in the input space. For training a one-class SVM classifier, we need only positive training examples (i.e. spam emails in this target application). There is no need to obtain a representative set of legitimate emails for training. In training a one-class SVM classifier, the user can adjust a parameter to allow a certain percentage of training samples to be outside the defined region (called the outlier percentage). The larger the outlier percentage, the smaller the defined region, and thus the lower the detection rate as well as the false-positive rate in the testing phase.

To evaluate the effectiveness of the one-class classifier, we have also implemented a classifier using the standard two-class SVM. Our analysis of the trade-offs between using one-class and two-class classifiers will be presented in Section 4.

## 4. EXPERIMENTS

We have designed and conducted experiments to investigate the following issues:

- The trade-offs between using the one-class and two-class classifiers for the filtering task.
- The added value of the proposed visual-based anti-spam filter beyond existing text-based filter.

### 4.1. Datasets

In the experiments, the statistics of the datasets used for training and validation were as follows:

- 8,500 spam emails as the positive training set.
- 1,500 spam emails as the positive validation set.
- 428 legitimate emails as the negative validation set.
- 10,000 artificial emails as the negative training set.

The positive training and validation set were created from the SpamArchive dataset introduced in section 2. From the spam emails which contain accessible images, we randomly selected 10,000 of them as the positive set. We reserved 15% of the positive set for validation and the rest for training.

We collected 428 legitimate emails which contained images to use as the negative validation set. These emails were volunteered by several people. We also generated 10,000 artificial emails using a large collection of photo images from Corel CDs. Each email contained a random number (from 1 to 10) of images. Every image was assumed to be local (i.e. explicitly attached in the email).

### 4.2. Experimental Results

In this experiment, we trained several one-class SVM classifiers using a different percentage of outliers for each classifier. We also trained a two-class SVM classifier using 8,500 spam emails as our positive training set and 10,000 artificial emails as our negative training set. The SVM package we used in this experiment was LIBSVM [9]. Table 2 compares the performance of these classifiers. In the case of one-class classification, it is clear that the tighter the positive region (i.e. the larger the outlier percentage), the lower the false-positive rate will be. In two-class classification, even though the detection rate is very high, the false-positive rate will also be high. One reason for the high false-positive rate could be that the negative sets (for both training and validation) are not representative or diverse enough. In anti-spam filtering, due to privacy issues, a representative negative set is difficult to collect. Moreover, low false-positive rates are more important than high detection rates. Therefore, we

chose the one-class SVM classifier as the base classifier with 20% outlier in our anti-spam filter.

**Table 2**. Classifier comparison

| Classifier | Detection Rate | False Positive |
|---|---|---|
| 1-class SVM, 5% outlier | 95.27 | 6.37 |
| 1-class SVM, 10% outlier | 90.80 | 2.69 |
| 1-class SVM, 20% outlier | 81.40 | 0.98 |
| 2-class SVM | 99.93 | 13.97 |

Table 3 shows the comparative performances of the anti-spam filters. The text-based Bayesian filter, trained using Ling-Spam dataset, detected only 47.73% of the spam emails. The proposed visual-based anti-spam filter, however, detected 81.40% of the spam emails (using 1-class SVM with 20% outlier) and brought an additional 36.87% improvement in detection rate to the text-based Bayesian filter.

**Table 3**. Filter comparison

| Filter | Detection Rate |
|---|---|
| Text-based | 47.73 |
| Visual-based | 81.40 |
| Text-based + Visual-based | 84.60 |

**Table 4**. Blocked/Missed email analysis

| Filter | Blocked | Missed |
|---|---|---|
| Text-based | 265 tokens | 96 tokens |
| Visual-Based | 68,517 pixels | 49,853 pixels |

Table 4 shows the statistics of emails blocked and those missed by the text-based Bayesian and the proposed anti-spam filters. Most of the spam emails that were missed by the Bayesian filter contained very few text tokens. Some of them contained no text at all. The average number of text tokens in these emails was 96, while the average number of text tokens in emails that were blocked by the Bayesian filter was 265. Moreover, the average number of image pixels per email in those spam emails blocked by the proposed anti-spam filter was 68,517 whereas the average number for the missed spam emails was 49,853.

The computation time for our proposed anti-spam filter depends on the number of images in the email and the number of text regions in each image. For a 384x256 photo image, it takes around 200-250 ms to extract features. For a text-rich image, it may take around 500-700 ms. To classify an email with four images, it takes roughly 2-3 seconds. Since the proposed anti-spam filter is a client-side filter, and email clients do not need real-time performance, the additional overhead caused by the proposed anti-spam filter is not a concern.

## 5. CONCLUSIONS AND FUTURE WORK

Previous work in content-based anti-spam filtering relies primarily on the text content of the emails. As the spammers' techniques become more sophisticated, and the genre of email content continues to evolve, these text-based approaches alone are no longer sufficient for solving the spam problem. In this paper, 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. Based on these features, we propose a novel anti-spam filter based on a one-class SVM classifier.

We have integrated the proposed anti-spam filter with Thunderbird, a mail client from Mozilla. The experimental results show that the proposed anti-spam filter are both effective and efficient. The results clearly demonstrate that the proposed anti-spam filter can bring extra filtering power to existing text-based anti-spam filters.

## 6. ACKNOWLEDGMENT

The authors would like to thank Mei-Chen Yeh for her contributions and support in this work.

## 7. REFERENCES

[1] J. Postel. On The Junk Mail Problem. RFC706. November 1975.

[2] J. Graham-Cumming. The Spammer's Compendium. http://www.jgc.org/tsc/index.html

[3] The Mozilla Project. http://www.mozilla.org

[4] SpamArchive. http://www.spamarchive.org

[5] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz. A Bayesian approach to filtering junk email. AAAI Workshop on Learning for Text Categorization, Jul. 1998, Madison, Wisconsin. AAAI Technical Report WS-98-05

[6] I. Androutsopoulos, J. Koutsias, K. V. Chandrinos, G. Paliouras, and C. D. Spyropoulos. An Evaluation of Naive Bayesian Anti-Spam Filtering. In Proc. of the workshop on Machine Learning in the New Information Age, 2000.

[7] A. Kolcz and J. Alspector. SVM-based filtering of e-mail spam with content-specific misclassification costs. ICDM Workshop on Text Mining (TextDM 2001), Nov. 2001.

[8] X. Carreras and L. Mrquez. Boosting trees for anti-spam email filtering. In Proceedings of RANLP-01, 4th International Conference on Recent Advances in Natural Language Processing, Tzigov Chark, BG, 2001

[9] C.-C. Chang and C.-J. Lin. LIBSVM: a library for support vector machines, 2001. Software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm

[10] V. Vapnik. The Nature of Statistical Learning Theory. Springer Verlag, New York, 1995.

[11] B. Scolkopf, J. Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson. Estimating the support of a high-dimensional distribution. Technical Report 99-87, Microsoft Research, 1999.

[12] C.-T. Wu, et al. A Novel Embedded-Text-in-Image Detector and Its Applications. UCSB Technical Report, January 2005.

[13] L.A. Fletcher and R. Kasturi. A Robust Algorithm for Text String Separation from Mixed Text/Graphics Images. IEEE Trans. on Pattern Analysis and Machine Intelligence, vol. 10, no. 6, pp. 910-918, Nov. 1988

[14] R. Lienhart and A. Wernicked. Localizing and segmenting text in images and videos. IEEE Trans. Circuits Syst. Video Technol., vol. 12, pp. 236-268, Apr. 2002.

[15] D. Chen, H. Bourlard, and J-Ph. Thiran. Text identification in complex background using SVM. In Proc. IEEE CVPR, p.621-626, Dec. 2001

[16] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In CVPR, 2001.