Active Learning Image Spam Hunter

Yan Gao and Alok Choudhary

Dept. of EECS, Northwestern University, Evanstion, IL, USA
ygao@cs.northwestern.edu, choudhar@eecs.northwestern.edu

Abstract. Image spam is annoying email users around the world. Most previous work for image spam detection focuses on supervised learning approaches. However, it is costly to get enough trustworthy labels for learning, especially for an adversarial problem where spammers constantly modify patterns to evade the classifier. To address this issue, we employ the principle of active learning where the learner guides the user to label as few images as possible while maximizing the classification accuracy. Active learning is more suited for online image spam filtering since it dramatically reduces the labeling costs with negligible overhead while maintaining high recognition performance. We present and compare two active learning algorithms, based on an SVM and a Gaussian process classifier respectively. To the best of our knowledge, we are the first to apply active learning for the task of spam image filtering. Experimental results demonstrate that our active learning based approaches quickly achieve $> 99\%$ high detection rate and $< 0.5\%$ low false positive rate with small number of images being labeled.

1 Introduction

Global spam volumes increase very fast over the past five years. Email spam accounted for 96.5% of incoming emails received by businesses by June 2008 [1], and costed more than $70 billion management expenses for US government annually. Among all spam emails, approximately 30% are image spams, which embed the spam messages in image attachments, as reported by McAfee [2] in 2007.

Detecting image spams is a typical image content recognition problem. In the arms race with anti-spam technology, spammers constantly employ different image manipulation technologies, such as all the tricks used in CAPTCHA (Completely Automated Public Turing Test to Tell Computers and Humans Apart), to embed spam messages into images. These different tricks include adding speckles and dots in the image background, varying borders, randomly inserting subject lines, and rotating the images slightly and so on. Figure 1 shows some examples of image spams.

Previous work has leveraged OCR techniques and text classifier for image spam detection. However, the appearance of CAPTCHA technologies easily degrades the recognition rate of an OCR system, which in turn affects the accuracy of the text classifier. As an improvement, many recent works have been targeting

G. Bebis et al. (Eds.): ISVC 2009, Part II, LNCS 5876, pp. 293–302, 2009.

Fig. 1. Spam image examples

on automated and adaptive content based image spam detection, e.g., Gao et al.'s image spam hunter [3], Dredze et al's fast image spam classifier [4], and near duplicate image spam detection [5]. Most of them employ supervised statistical machine learning algorithms to build a classifier for filtering spam images using discriminative image features.

Although supervised learning algorithms have achieved good accuracy for the task of image spam detection, getting sufficient labeled images for robust training is always expensive, especially for the adversarial problem that re-training model needs to be done quite often. By leveraging the principle of active learning [6,7,8,9,10], we can drastically reduce the labeling cost by identifying the most informative examples for users to label. Hence in this paper we propose a system prototype of an active learning image spam hunter to solve the adversarial spam detection problem.

Our goal is to create a strong classifier while requesting as few labels as possible. We present and compare two active learning algorithms, which are based on an SVM and a Gaussian process classifier respectively. These two algorithms are tested on an image spam dataset collected from Jan 2006 to Mar 2009, which contains both positive spam images collected from our email server, and negative natural images downloaded from Internet. Our approaches on average requires very few images to be labeled in a corpus to achieve >99% detection rate and <0.5% false positive rate.

The remainder of this paper is organized as follows. Section 2 presents an overview of system design and operation flow of the active learning image spam hunter. Then in Section 3, we describe two active learning algorithms. One is based on an SVM classifier, and the other is based on a Gaussian process classifier. We present the image statistic features adopted to discriminate natural and spam images in Section 4. In Section 5, we use extensive experiments to validate and compare the effectiveness of the proposed system and algorithms. Finally, we conclude and summarize our future work in Section 6.

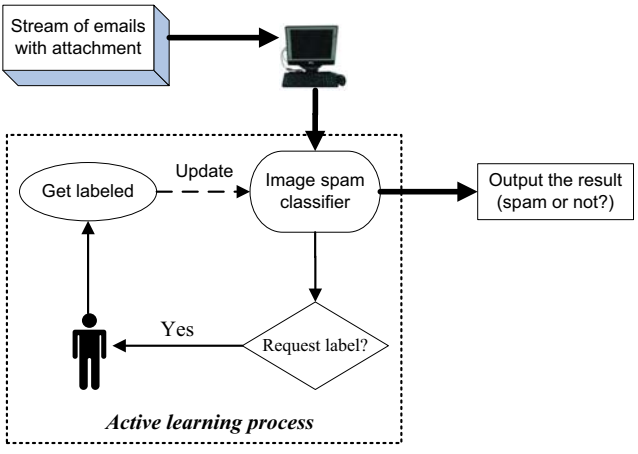

Fig. 2. Prototype system diagram

2 System Framework

In this section, we present an active learning system prototype of image spam hunter, as shown in Figure 2, to differentiate spam images from normal image attachments. The whole dataset is splitted into two: the labeled dataset and unlabeled dataset. The labeled dataset is denoted as $\mathcal{X}_L = \{\mathbf{x}_i | i \in L\}$, with labels $\mathcal{Y}_L = \{y_i \in \{-1, +1\} | i \in L\}$, where 1 represents spam image and -1 represents non-spam image, respectively. The unlabeled dataset is denoted as $\mathcal{X}_U = \{\mathbf{x}_i | i \in U\}$. We assume $L = [1, n]$ and $U = [n + 1, N]$. Let $\mathcal{X} = \mathcal{X}_L \cup \mathcal{X}_U$. When the system is firstly used, $\mathcal{X}_L$ is an empty set ϕ and $\mathcal{X}_U$ may cover the full dataset $\mathcal{X}$. We randomly choose a few (< 10) spam images and non-spam images to label and take them as the initial labeled dataset for training the first round classifier.

The core of this prototype system is an active learning algorithm with a data sample choosing criterion $AL(y(\mathbf{x}))$, where $y(\mathbf{x})$ is the classifier induced from the learning algorithm. As long as an appropriate mathematic quantity $AL(y(\mathbf{x}))$ is defined, we can make any supervised learning algorithm to be an active learning algorithm. The active learning criterion $AL(y(\mathbf{x}))$ efficiently guides the users to label as few images as possible while maximizing the recognition performance of the classifiers.

More formally, at each step of the active learning algorithm, we first perform the supervised learning algorithm with the current $\mathcal{X}_L$, and build the image spam classifier $y(\mathbf{x})$. Next we select

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{X}_U} AL(y(\mathbf{x})) \tag{1}$$

to label and get

$$\mathcal{X}_L \Leftarrow \mathcal{X}_L + \mathbf{x}^* \tag{2}$$

$$\mathcal{X}_U \Leftarrow \mathcal{X}_U - \mathbf{x}^*. \tag{3}$$

With the new $\mathcal{X}_L$, the above active learning step is iterated until the recognition accuracy of the classifier reaches a satisfactory level. We will discuss the selection of iteration times in Section 5. In this way, the continuously adaptive classifier is generated and ready to filter the incoming batch of new emails with image attachment.

3 Active Learning Algorithms

We present two different active learning algorithms in this section. One is adapted from the probabilistic output of an SVM(Support vector machines) [11,12]. The other is built on top of a Gaussian process (GP) classifier [13,14].

3.1 Active Learning SVM

Given the labeled data set $\mathcal{X}_L = \{\mathbf{x}_i, y_i\}_{i=1}^n$, the primal problem of a linear SVM solves the following quadratic program for obtaining the maximum margin linear classifier [15,16], i.e.,

$$\min_w \ \frac{\|\mathbf{w}\|^2}{2} + C\sum_i \xi_i \tag{4}$$

$$s.t. \ \ y_i(\mathbf{x}_i \cdot \mathbf{w} + b) \leq 1 - \xi_i \ and \ \xi_i \leq 0 \ \ \forall i. \tag{5}$$

The solution of the above constrained optimization problem is usually obtained by solving the Wolfe dual problem,

$$\max \ \sum_i \alpha_i - \frac{1}{2}\sum_{i,j} \alpha_i \alpha_j \mathbf{x}_i \cdot \mathbf{x}_j \tag{6}$$

$$s.t. \ \ 0 \leq \alpha_i \leq C \ \ \forall i \ and \ \sum_i \alpha_i y_i = 0. \tag{7}$$

It shows that the solution is given by

$$\mathbf{w} = \sum_{i=1}^{N_s} \alpha_i y_i \mathbf{x}_i, \tag{8}$$

where N_s indicates the number of support vectors for the classifier. Therefore, the classification result of a new data vector $\mathbf{x}$ is

$$y = sign\left\{\sum_{i=1}^{N_s} \alpha_i y_i \mathbf{x}_i \cdot \mathbf{x} + b\right\}. \tag{9}$$

It is easy to observe that in both the Wolfe dual problem Equation 6 and the final classifier Equation 9, the data vectors only present in the form of dot

product. This enables us to construct nonlinear SVM by leveraging the kernel tricks [16], i.e., to solve the following problem

$$\max \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \tag{10}$$

$$s.t.\ 0 \leq \alpha_i \leq C \quad \forall i. \tag{11}$$

$$\sum_i \alpha_i y_i = 0, \tag{12}$$

where $k(\mathbf{x}_i, \mathbf{x}_j)$ is a kernel function which defines the dot product of the nonlinear transformed data vectors $\phi(\mathbf{x}_i)$ and $\phi(\mathbf{x}_j)$ in a reproducing kernel Hilbert space (we use Gaussian radial basis kernel in our experiments), i.e.,

$$k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j). \tag{13}$$

Similarly, the final nonlinear SVM classifier is

$$y = sign \left\{ \sum_{i=1}^{N_s} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b \right\}. \tag{14}$$

Note that we do not need to explicitly define the nonlinear transformation $\phi(\mathbf{x})$ since both the optimization problem in Equation 10 and the solution in Equation 14 only involves the kernel function. As shown by Madevska-Bogdanova et al. [11], we could transform the function output from a support vector machine to be a posterior distribution by using a Sigmoid function, i.e.,

$$p(y = 1|\mathbf{x}) = \frac{1}{1 + \exp\{k(\sum_{i=1}^{N_s} \alpha_i y_i k(\mathbf{x}_i, \mathbf{x}) + b)\}}, \tag{15}$$

where k is a constant quantity which could be estimated from the training data. With this posterior probability of the predicted label given the data point, a natural active learning criterion would be based on the uncertainty of the predicted label given a data point. Let $p_1 = p(y = 1|\mathbf{x})$. The uncertainty is naturally defined by an entropy term,

$$H(y(\mathbf{x})) = -p_1 \log p_1 - (1 - p_1) \log(1 - p_1). \tag{16}$$

Therefore, for this active learning SVM, we define

$$AL(f(\mathbf{x})) = H(y(\mathbf{x})). \tag{17}$$

The rationale behind the criterion is that the active learning algorithm should guide the users to label the image for which the classifier are least confident to recognize.

3.2 Active Learning Gaussian Process Classifier

Given the labeled dataset $\mathcal{X}_L$, an unlabeled data $\mathbf{x}_u$, and $\mathcal{X}_{Lu} = \mathcal{X}_L + \mathbf{x}_u$, we introduce a latent variable z_i, which is the soft label of the data point $\mathbf{x}_i$. We denote $\mathcal{Z}_{Lu} = \{z_i | i \in L + u\}$. In a GP classifier, the joint distribution of $\mathcal{Z}_{Lu}$ is assumed to be a joint Gaussian with zero mean and covariance defined by a kernel function $k(\cdot, \cdot)$ applied to $\mathbf{x}_i$ and $\mathbf{x}_j$, i.e.,

$$p(\mathcal{Z}_{Lu}|\mathcal{X}_{Lu}) \sim \mathcal{N}(\mathbf{0}, \mathbf{K}), \tag{18}$$

where $\mathbf{K}$ is a $N \times N$ matrix with the element $k_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$. We denote K_{LL} be the sub-matrix of $\mathbf{K}$ that is induced by $\mathcal{X}_L$. Following Kapoor et al. [14], we assume $p(y|z)$ be a Gaussian distribution $\mathcal{N}(y, \sigma^2)$. We immediately have

$$p(\mathcal{Z}_{Lu}|\mathcal{X}_{Lu}, \mathcal{Y}_L) \propto p(\mathcal{Z}_{Lu}|\mathcal{X}_{Lu}) p(\mathcal{Y}_L|\mathcal{Z}_{Lu}) \tag{19}$$

$$= p(\mathcal{Z}_{Lu}|\mathcal{X}_{Lu}) \prod_{i \in L} p(y_i|z_i) \tag{20}$$

Denote y_u be the label of $\mathbf{x}_u$ we would like to predict, we are interested in inferring the following quantity

$$p(y_u|\mathcal{X}_{Lu}, \mathcal{Y}_L) = \int_{\mathcal{Z}_{Lu}} p(y_u|\mathcal{Z}_{Lu}) p(\mathcal{Z}_{Lu}|\mathcal{X}_{Lu}, \mathcal{Y}_L) d\mathcal{Z}_{Lu}. \tag{21}$$

Denote $\mathbf{k}(\mathbf{x}_u) = [k(\mathbf{x}_u, \mathbf{x}_1), k(\mathbf{x}_u, \mathbf{x}_2), \ldots, k(\mathbf{x}_u, \mathbf{x}_n)]^T$, and $\mathbf{I}$ be the identity matrix, by following [14], we have

$$p(y_u|\mathcal{X}_{Lu}, \mathcal{Y}_L) = \mathcal{N}(\bar{y}_u, \bar{\sigma}_u^2) \tag{22}$$

where

$$\bar{y}_u = \mathbf{k}(\mathbf{x}_u)^T (\sigma^2 \mathbf{I} + \mathbf{K}_{LL})^{-1} \mathcal{Y}_L \tag{23}$$

$$\bar{\sigma}_u^2 = k(\mathbf{x}_u, \mathbf{x}_u) - \mathbf{k}(\mathbf{x}_u)^T (\sigma^2 \mathbf{I} + \mathbf{K}_{LL})^{-1} \mathbf{k}(\mathbf{x}_u) + \sigma^2. \tag{24}$$

Denote $p_1 = p(y_u = 1|\mathcal{X}_{Lu}, \mathcal{Y}_L)$, we can define the entropy by using Equation 16, and the active learning criterion would exactly take Equation 17. It is worth noticing that Kapoor et al. [14] defined their active learning criterion with this GP classifier to be

$$AL(y(\mathbf{x})) = -\frac{|\bar{y_u}|}{\bar{\sigma}_u}. \tag{25}$$

In this binary classification problem, it is easy to verify that this is equivalent to our entropy uncertain measure.

4 Image Features

We extract 23 discriminant image statistical features [17] for our active learning image spam hunter. They cover the properties of color, texture, shape and appearance.

For color statistics, we first build a 10^3-dimension color histogram in the joint RGB space by quantizing each color band into 10 different levels. The *entropy* of this histogram is computed as the first statistics. We further set up one 100-dimension histogram for each of the 3 color channels. Then the *discreteness*, *mean*, *variance*, *skewness*, and *kurtosis* for each of the three histograms are calculated, which adds another $5 \times 3 = 15$ statistics. Here the discreteness is the summation of all the absolute differences between any two consecutive bins. So altogether we collect 16 color statistics.

Local binary pattern (LBP) [18] is used to analyze the texture statistics. We extract 59-dimension texture histogram, including 58 bins for all the different uniform local binary patterns, i.e., the pattern of at most two $0 \sim 1$ transitions in a 8-bit stream, and an additional bin for all other non-uniform local binary patterns. The *entropy* of the LBP histogram is calculated as 1 texture statistics.

Shape information is also considered as the important features in our system. A $40 \times 8 = 320$ dimensional gradient magnitude-orientation histogram is built to describe the shape information. The *entropy* of the histogram is the first shape feature, and the second feature is the difference between the energies in the lower frequency band and the higher frequency band. Then we use the total amount of edges and the average length of the edges as another two shape features by running a Canny edge detector [19]. Thus there are 4 shape statistics in total.

Last but not least, we use the spatial correlogram [20] of the grey level pixels within 1-neighborhood to represent appearance information. The first feature is the *average variance ratio* of all the slices, which is the ratio between the variance of the slice and the radius of the symmetric range over the mean of the slice that accounts for 60% of the total counts of the slice. Then histograms are built from each slice of the correlogram, and the *average skewness* of the histograms is calculated as the second feature.

5 Experiment

In our experiments, we report the recognition accuracy on both the *active learning pool* $\mathcal{X} = \mathcal{X}_L \cup \mathcal{X}_U$, and the *hold-out data-set* $\mathcal{X}_h$. We keep track of the recognition accuracy with the progress of active learning. We also compare with a baseline setting where at each step we randomly choose an image sample from $\mathcal{X}_u$ for the users to label. We call the active learning process to be *active supervision* and the baseline setting to be *random supervision*. We adopt the Gaussian radial basis kernel for both the SVM and the GP classifier. In the following, we present our data collection first followed by the detailed experimental results.

5.1 Data Collection

We collected an image dataset which contains 1190 spam images and 1760 normal images. The spam images are extracted from real spam images received by 10 graduate students in our department between Jan. 2006 and Mar. 2009. These spam images were extracted from the original spam emails and all of them are converted to JPEG format. For normal image attachments, we collect photo

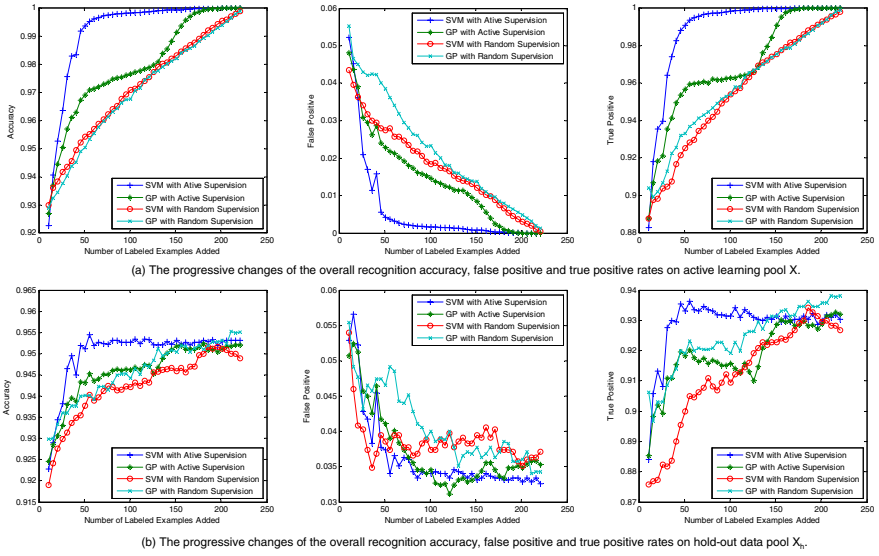

Fig. 3. The changes of the accuracy with the progress of active learning

images by either downloading from photo sharing site Flikr.com, or fetching the photo images from popular image search engines such as Microsoft live image search (http://www.live.com/?scope= images).

5.2 Results Comparison

Since typical users usually deal with hundreds of emails in a one-day batch, we randomly extract a subset of 10% images from the whole data corpus as the test subset in each experiment. To test the generalization performance of the classifiers induced from active learning, each time we randomly sample 20% data from the test subset as a hold-out dataset $\mathcal{X}_h$. The rest 80% is adopted as the active learning pool $\mathcal{X}$. We randomly select 10 samples from active learning

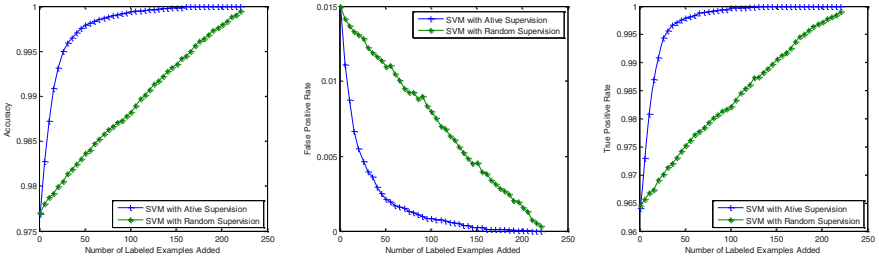

Fig. 4. The recognition accuracy of running active learning SVM on an initialized classifier

pool to initialize the system. Figure 3 presents the experimental results averaged over 100 runs. Part (a) in Figure 3 presents the progressive changes of the overall recognition accuracy, false positive and true positive rates on $\mathcal{X}$ with the human adding more and more labels, while part (b) shows the results on $\mathcal{X}_h$.

In general, the classifiers induced form active supervision achieve much better results than those from random supervision, in other words, much less amount of labels are needed for active supervision to achieve the same recognition accuracy as random supervision. In particular, the active learning SVM only requires to label less than 50 images in $\mathcal{X}$ to achieve over 99% recognition accuracy. This is also observed in the holdout dataset where the recognition accuracy quickly approach to the saturation point than the algorithms with random supervision. Moreover, with our feature setting and the selected kernel function, the active learning SVM consistently shows better performance than the active learning GP classifier.

The recognition performance on $\mathcal{X}_h$ also shows that the induced classifier generalize well so that it may be employed for fully automated image spam filtering. But it is preferred to always run in the active learning mode as we can ensure more than 99% accuracy by the end of the learning process. If not considering the initialization process of the system, the amount of labels required to adapt the classifier to next batch of emails is even less. Figure 4 presents the recognition performance of continuously running the active SVM algorithm on a second subset of data, initialized from the SVM classifier obtained from the first subset. The reported results are also averaged over 100 different runs. As we can clearly observe, with a well-trained initial SVM, the active learning SVM only requires to label 20 ($<$7%) images in order to achieve over 99% recognition accuracy. That is to say, our active learning image spam hunter system only needs $<$7% label data to get the ideal high detection rate. This ratio may further reduce with the increase of the dataset .

6 Conclusion

In conclusion, we propose to employ active learning for online image spam email filtering. The design of a prototype system is presented and two different active learning algorithms are evaluated. Our extensive comparative experiments manifests that the active learning SVM is a better choice for this task, given the image statistic features we adopted.

Acknowledgements

This work was supported in part by DOE FASTOS award number DE-FG02-08ER25848, NSF HECURA CCF-0621443, NSF SDCI OCI-0724599, CNS-0830927, and NSF ST-HEC CCF-0444405.

References

1. Sophos Plc:
`http://www.sophos.com/pressoffice/news/articles/2008/07/dirtydozj-ul08.html`

2. McAfee: http://www.avertlabs.com/research/blog/?p=170
3. Gao, Y., Yang, M., Zhao, X., Pardo, B., Wu, Y., Pappas, T., Choudhary, A.: Image spam hunter. In: Proc. of the 33th IEEE International Conference on Acoustics, Speech, and Signal Processing, Las Vegas, NV, USA (2008)
4. Dredze, M., Gevaryahu, R., Elias-Bachrach, A.: Learning fast classifiers for image spam. In: Proc. the 4th Conference on Email and Anti-Spam (CEAS), California, USA (2007)
5. Wang, Z., Josephson, W., Lv, Q., Charikar, M., Li, K.: Filtering image spam with near-duplicate detection. In: Proc. the 4th Conference on Email and Anti-Spam (CEAS), California, USA (2007)
6. Freund, Y., Seung, H.S., Shamir, E., Tishby, N.: Selective sampling using the query by committee algorithm. Machine Learning 28, 133–168 (1997)
7. Tong, S., Koller, D., Kaelbling, P.: Support vector machine active learning with applications to text classification. Journal of Machine Learning Research, 999–1006 (2001)
8. Goh, K.S., Chang, E.Y., Lai, W.C.: Multimodal concept-dependent active learning for image retrieval. In: Proceedings of the 12th annual ACM international conference on Multimedia. ACM, New York (2004)
9. Lawrence, N.D., Seeger, M., Herbrich, R.: Fast sparse gaussian process methods: The informative vector machine. In: Advances in Neural Information Processing Systems, vol. 15, pp. 609–616. MIT Press, Cambridge (2003)
10. MacKay, D.J.C.: Information-based objective functions for active data selection. Neural Computation 4, 590–604 (1992)
11. Madevska-Bogdanovaa, A., Nikolikb, D., Curfsc, L.: Probabilistic svm outputs for pattern recognition using analytical geometry. Neurocomputing 62, 293–303 (2004)
12. Cortes, C., Vapnik, V.: Support-vector networks. Machine Learning 20, 273–297 (1995)
13. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. MIT Press, Cambridge (2006)
14. Kapoor, A., Grauman, K., Urtasun, R., Darrell, T.: Active learning with gaussian processes for object categorization. In: Eleventh IEEE International Conference on Computer Vision, Rio de Janeiro, Brazil (2007)
15. Burges, C.J.C.: A tutorial on support vector machines for pattern recognition. Data Mining and Knowledge Discovery 2, 121–167 (1998)
16. Aizerman, A., Braverman, E.M., Rozoner, L.I.: Theoretical foundations of the potential function method in pattern recognition learning. Automation and Remote Control 25, 821–837 (1964)
17. Ng, T.T., Chang, S.F., Tsui, M.P.: Lessons learned from online classification of photo-realistic computer graphics and photographs. In: IEEE Workshop on Signal Processing Applications for Public Security and Forensics, SAFE (2007)
18. Mäenpä, T.: The local binary pattern approach to texture analysis extensions and applications. PhD thesis, Infotech Oulu, University of Oulu, Finland (2003)
19. Canny, J.: A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 8, 679–698 (1986)
20. Huang, J., Kumar, S.R., Mitra, M., Zhu, W.J., Zabih, R.: Image indexing using color correlograms. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Los Alamitos, CA, USA (1997)