Document Image Classification Via AdaBoost and ECOC Strategies Based on SVM Learners

Mehmet Ahat[1,2], Cagdas Ulas[1] and Onur Agin[1]
[1]*R&D and Special Projects Department, Yapi Kredi Bank, Gebze, Kocaeli, Turkey*
[2]*Faculty of Engineering and Natural Sciences, Sabanci University, Tuzla, Istanbul, Turkey*

Keywords: Document Image Retrieval and Classification, SVM, One-Versus-All, AdaBoost, ECOC, BoVW Model.

Abstract: In this paper, we describe easily extractable features and an approach for document image retrieval and classification at spatial level. The approach is based on the content of the image and utilizing visual similarity, it provides high speed classification of noisy text document images without optical character recognition (OCR). Our method involves a bag-of-visual words (BoVW) model on the designed descriptors and a Random-Window (RW) technique to capture the structural relationships of the spatial layout. Using the features based on these information, we analyze different multiclass classification methods as well as ensemble classifiers method with Support Vector Machine (SVM) as a base learner. The results demonstrate that the proposed method for obtaining structural relations is competitive for noisy document image categorization.

1 INTRODUCTION

Document Image Retrieval is a crucial research area dealing with the problem of retrieving structurally similar document images from a large heterogeneous collection given a *relevant* image, which is useful for document database management, information extraction and document routing (Hu et al., 1999). In today's world, large quantities of paper documents are converted into electronic form and stored as document images in digital libraries. Storing scanned document images alone does not suffice, instead, it is only beneficial when the process of retrieving relevant documents should be done in an efficient manner. However, the number of *relevant* documents provided for retrieval is usually much lower than the number of *irrelevant* documents, resulting in imbalance problem in the data and thus, the retrieval problem becomes more challenging (Zheng et al., 2004).

Several methods have been practised for document image retrieval and categorization. Most of these methods are mainly based on layout (structure) or content of the documents. Content based approaches are highly dependent on the quality of OCR and attaining OCR result on the entire document is excessively expensive in terms of time (Kumar et al., 2012). As shown in Figure 1, images from different types of documents often have quite distinct spatial layout styles. At very low resolutions, these distinctions are also identifiable which allows us to develop faster algorithms than that of content based techniques for document classification (Hu et al., 1999). One of the well known methods on layout similarity is based on the block segmentation where the image is divided into several structural blocks (Fan et al., 2001) and these blocks are then compared to their analogous for the given type of documents. Another popular method proposes the creation of spatial-pyramid features (Lazebnik et al., 2006) by partitioning the image into smaller grids and computing the density of features in each region. Finding efficient ways to capture the structural relationships at local level is an important research problem, and several methods (Yang and Newsam, 2011; Kumar and Doermann, 2013) have been proposed before on this issue.

In this work, we propose a method for the classification of noisy document images. All the documents that we consider in classification are extracted from our real-world bank dataset, consisting various types of forms (see Figure 1) mostly used in loan applications and scanned by bank branch employee to be converted to digital format. Our approach in this study employs different multiclass classification methods and AdaBoost based ensemble classifiers with SVM as a base learner to make predictions on document type by utilizing a set of features that represent the structural information at spatial layout level. Our work differs from previous approaches in sev-

Ahat M., Ulas . and Agin O..
Document Image Classification Via AdaBoost and ECOC Strategies Based on SVM Learners.
DOI: 10.5220/0005131502500255
In *Proceedings of the International Conference on Neural Computation Theory and Applications* (NCTA-2014), pages 250-255
ISBN: 978-989-758-054-3

eral ways: (1) We represent each small region of the image with a very small number of feature descriptors, which results in high speed for feature extraction procedure (2) Images are represented with less number of visual codewords compared with other methods using SIFT features (Smith and Harvey, 2011) as the key descriptor (3) Without any use of *irrelevant* images in training phase, the proposed method can achieve high recall rate for correct detection of *irrelevant* images (4) We compare the performance of different multiclass classification strategies as well as ensemble classifiers and demonstrate that the results are especially competitive when SVM based ensemble classifiers are used for this kind of imbalanced dataset problem.

The remainder of this paper is structured as follows: In Section 2, the proposed noisy document image classification method is presented. We provide a description of the experimental setup and demonstrate the classification results in Section 3 and finally we give concluding remarks in Section 4.

2 PROPOSED METHOD

The proposed method in this work is composed of following steps: the description of feature descriptors, the utilization of BoVW model with Random-Window approach and the analysis of SVM based multiclass classification strategies and ensemble classifiers. Following sections will provide the details of these steps.

2.1 Feature Descriptors

The document images in our business form database are binary (monochrome) images that the pixel values can take only one of two values (0,1), suffering from lack of information as compared to color and gray-scale images. Hence, specifically designed features should be found for this type of images.

In this work, we divide each training and test image into small square patches to determine a 60×40 image layout with 2400 image patches and represent each patch with 4 feature descriptors based on structural variations in small local areas.

If P_C denotes the width (column number) of the image patch; P_R denotes the height (row number) of the patch and $w(i,j)$ represents the pixel value of the pixel at ith row and jth column, the feature descriptors are calculated as follows:

1. **Column Standard Deviation (σ_c)**

$$\sigma_c = \sqrt{Variance(\mathbf{X_C})} \qquad (1)$$

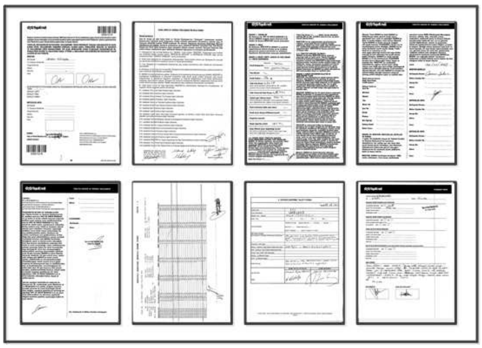

Figure 1: Sample business forms in the dataset. Each form is an example from one of the *relevant* classes.

where $\mathbf{X_C} = [X_1 X_2 ... X_{P_C}]$ and each $X_i = \sum_{j=1}^{P_R} w(j,i)$

2. **Row Standard Deviation (σ_r)**

$$\sigma_r = \sqrt{Variance(\mathbf{X_R})} \qquad (2)$$

where $\mathbf{X_R} = [X_1 X_2 ... X_{P_R}]$ and each $X_i = \sum_{j=1}^{P_C} w(i,j)$

3. **Patch Mean Value (m_p)**

$$m_p = \frac{1}{P_R P_C} \left(\sum_{j=1}^{P_C} \sum_{i=1}^{P_R} w(i,j) \right) \qquad (3)$$

4. **Pixel Transition Intensity (t_s)**

$$t_s = \frac{1}{P_R(P_C - 1)} \left(\sum_{j=1}^{P_R} \sum_{i=1}^{P_C} trans(j,i) \right) \qquad (4)$$

where $trans(j,i)$ is defined as follow:

$$trans(j,i) = \begin{cases} 1 & \text{if } w(j,i-1) = 1 \ \& \ w(j,i) = 0 \\ 0 & \text{otherwise} \end{cases}$$

2.2 Bag of Visual Words (BoVW) Model

In computer vision, the BoVW model (Csurka et al., 2004) can be applied to image classification and related tasks by treating image descriptors as words. A bag of visual words is a sparse vector of mostly occurrence counts or presence of the visual words from a vocabulary of local image features. A vocabulary (or codebook) of visual models is obtained by clustering local image descriptors extracted from training images, which is also described as vector quantization of image features into visual words. The vector quantization process is generally done by a hard or soft assignment (clustering) and a codebook of visual words is obtained. Visual words (codewords) are then defined as the centers of learned clusters.

In this work, we use k-means clustering (Winn et al., 2005) to determine the codebook of visual words. The number of cluster is empirically set as 4 due to the intuition that the structural variations inside the local patches of text-document images are small. After obtaining the visual words, each train and test images are represented with a sequence of visual words in a 60×40 layout. This layout is given as the input to "Random Window" generator where a pre-defined number of windows inside the layout is selected and each window is represented with a set of features based on structural relations.

2.3 Random Window (RW) Approach

As it was mentioned in Section 2.2, we represent our document images by visual words. However, visual words are not enough to discriminate the structure of different documents. Usually each type contains particular image patterns inside sub-images whose coordinates and sizes are unknown. In order to capture spatial relationships, after converting all document images into a sequence of visual words in a 60×40 layout, we randomly select rectangular windows inside the layout and extract layout features by using the following approach:

Let RW is the set of randomly selected windows' coordinates represented as

$$RW_i = \left\{ \left(m_i^{'}, m_i^{''}\right), \left(n_i^{'}, n_i^{''}\right) \right\}$$

where $0 \leq m_i^{'} \leq m_i^{''} \leq 60$ and $0 \leq n_i^{'} \leq n_i^{''} \leq 40$.

Let V is the available number of visual words in vocabulary and S is the set of occurrence counts of the visual words for given RW's. Hence, $S_i = \left\{s_1^i, s_2^i, ..., s_V^i\right\}$ where s_j^i is the occurrence count of the jth visual word in RW_i. For given RW_i, the feature vector $\mathbf{F_i}$ is defined as; $\mathbf{F_i} = \left[s_1^i/\eta, s_2^i/\eta, ..., s_V^i/\eta\right]$ where η is the normalization constant calculated as $\eta = \sqrt{(s_1^i)^2 + (s_2^i)^2 + ... + (s_V^i)^2}$ for the set of S_i.

2.4 Support Vector Machine (SVM)

SVM is basically conceived for binary classification. The idea is to separate two classes by calculating the maximum margin hyperplane between the training examples (Vapnik, 1998). The decision function of SVM for a binary classification problem is

$$f(x) = \langle \mathbf{w}, \phi(\mathbf{x}) \rangle + b \qquad (5)$$

where $\phi(\mathbf{x})$ is a mapping of sample $\mathbf{x}$ from the input space to a high dimensional feature space. $\langle .,. \rangle$ denotes the dot product in the feature space. The optimal values of $\mathbf{w}$ and b can be determined by solving the following optimization problem:

$$\begin{aligned} &\text{minimize} \quad g(\mathbf{w}, \xi) = \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{N} \xi_i \\ &\text{subject to} \quad y_i(\langle \mathbf{w}, \phi(\mathbf{x}_i) \rangle + b) \geq 1 - \xi_i,\ \xi_i \geq 0 \end{aligned} \qquad (6)$$

where ξ_i is the ith slack variable and C is the regularization parameter. The minimization problem in (6) can be written as

$$\begin{aligned} &\text{minimize} \quad W(\alpha) = -\sum_{i=1}^{N} \alpha_i + \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} y_i y_j \alpha_i \alpha_j k(\mathbf{x}_i, \mathbf{x}_j) \\ &\text{subject to} \quad \sum_{i=1}^{N} y_i \alpha_i = 0,\ \forall i : 0 \leq \alpha_i \leq C \end{aligned} \qquad (7)$$

where α_i is a Lagrange multiplier corresponding to sample $\mathbf{x}_i$, $k(.,.)$ is a kernel function which implicitly maps the input vectors into a suitable feature space. In this space, an optimal separating hyperplane is constructed by the support vectors.

$$k(\mathbf{x}_i, \mathbf{x}_j) = \langle \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) \rangle \qquad (8)$$

In this work, we use the RBF kernel, $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\|\mathbf{x}_i - \mathbf{x}_j\|)^2/2\sigma^2)$. The performance of RBF-SVM is mainly affected by the kernel parameters, for example, σ, and the regularization parameter, C. By using model selection techniques such as k-fold or leave-one-out cross-validation (CV), a single best σ and C can be found (Li et al., 2008).

The following briefly describes several notations used in this paper:

- $T = \{(\mathbf{x}_1, y_1), (\mathbf{x}_2, y_2), ..., (\mathbf{x}_N, y_N)\}$: A training set; where $\mathbf{x}_i \in R^n$; each label, y_i is an integer value belongs to $Y = \{l_1, l_2, ..., l_{N_c}\}$, where N_c is the number of classes. $h = \{h_1, h_2, ..., h_n\}$: A set of n binary classifiers.

2.5 Multiclass SVM Classifiers

2.5.1 One-Versus-All (OVA)

The one-versus-all method constructs n binary classifiers, one for each class. The ith classifier, h_i, is trained with the data from class i as positive instances and all data from other classes as negative instances to discriminate among the patterns of the class and the patterns of the remaining (Bagheri et al., 2012). A new instance is classified as the class whose corresponding classifier output has the largest value (probability). Hence, the ensemble decision function, h, is defined as:

$$y = \underset{i \in \{1,2,...,n\}}{\text{argmax}}\ h_i(x) \qquad (9)$$

2.5.2 Error Correcting Output Codes (ECOC)

The ECOC framework is widely used for multiclass classification problems. It is based on combining binary classifiers and designing a codeword for each class. Since each class is coded by different codewords, it may exhibit error-correcting capabilities, which increases accuracy of the multiclass problem (Dietterich and Bakiri, 1995). Let M is a coding matrix with a dimension of $N_c \times L$ whose elements $m_{i,j}$ can be $\{-1,+1\}$ for dense ECOC models. L is the length of codewords which is used for class assignments. Each column of M is a map of binary classifier that separates each class from the others. Later on, the ECOC method was extended and elements can be $\{-1,0,+1\}$ where "0" labeled elements in the coding matrix is not considered during the training phase of a particular binary classifier (Allwein et al., 2001).

$$M_{N_c,L} = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,L} \\ m_{2,1} & m_{2,2} & \cdots & m_{2,L} \\ \vdots & \vdots & \ddots & \vdots \\ m_{N_c,1} & m_{N_c,2} & \cdots & m_{N_c,L} \end{bmatrix}$$

When an instance x is tested, each binary classifier predicts -1 or 1 for x and these predictions creates a L long output code vector. x is labeled as the class whose codeword has minimum distance. The distance is usually hamming distance between the output code vector and class codewords.

2.6 AdaBoost SVM

Although the use of SVM as a component (base) classifier in AdaBoost may not seem to be in accordance with Boosting principle since SVM itself is not a weak classifier, the proposed *AdaBoostSVM* in (Li et al., 2008) demonstrates that it can show better generalization performance than SVM on imbalanced classification problems that we consider in this study. The key idea of *AdaBoostSVM* is that for a sequence of trained SVM component classifiers, starting with large σ values (implies week learning), the σ values are reduced progressively as the Boosting iteration proceeds which effectively produces a set of component classifiers whose model parameters are adaptively different resulting in better generalization as compared to using a fixed (optimal) σ value.

The *AdaBoostSVM* method (Li et al., 2008) which was proposed for binary classification problem can be easily modified for OVA multiclass approach. The pseudo-code of the *OVA-AdaBoostSVM* is provided in Algorithm 1.

Algorithm 1: *OVA-AdaBoostSVM.*

Input: T, Y, number of training samples, N; the initial σ, σ_{ini}; the minimum σ, σ_{min}; the step of σ, σ_{step}.
for each $l_n \in Y$ **do**
  Apply the class binarization for class l_n on T.
  Initialize: the weights of training samples: $w_i^1 = \frac{1}{N}$.
  while $\sigma > \sigma_{min}$ **do**
    (1) Train a RBF-SVM component classifier, h_t, on weighted T.
    (2) Calculate the training error of h_t : $\varepsilon_t = \sum_{i=1}^{N} w_i^t,\ y_i \neq h_t(\mathbf{x}_i)$.
    (3) If $\varepsilon_t > 0.5$, decrease σ value by σ_{step} and go to step (1).
    (4) Set the weight of h_t : $\alpha_t = \frac{1}{2}\ln(\frac{1-\varepsilon_t}{\varepsilon_t})$.
    (5) Update the weights of training samples: $w_i^{t+1} = \frac{w_i^t \exp\{-\alpha_t y_i h_t(\mathbf{x}_i)\}}{C_t}$ where C_t is a normalization constant, and $\sum_{i=1}^{N} w_i^{t+1} = 1$.
  end while
  $f_{l_n}(\mathbf{x}) = \text{sign}(\sum_{t=1}^{T} \alpha_t h_t(\mathbf{x}))$
end for
Output: $y(\mathbf{x}) = l_n \in Y$, where $f_{l_n}(\mathbf{x}) = 1$.

3 EXPERIMENTAL RESULTS

In this work, we consider the classification problem of document images from 8 types (classes) of business forms as shown in Figure 1. We use our own data set to evaluate the performance of proposed classification approach. Our training data consists of 50 image samples for each class. The test data consists of 2066 image samples. 1267 of these images are *irrelevant* images whereas 799 of them are *relevant* images. Most of the images in the training and test dataset were contaminated with marginal (clutter) noise and salt&pepper noise during scanning, transmission or conversion to digital form.

Three different methods for classification analysis of this multiclass classification problem are compared in this study: OVA-SVM, OVA-AdaBoost-SVM, Sparse ECOC-SVM. For OVA and ECOC based SVM modeling, 5-fold CV conducted for both parameter tuning and generalization capability. For AdaBoost-SVM, the regularization parameter C is empirically set as 10 for all experiments. The σ_{min} is computed as the average minimum distance between any two training samples inside the subset of training data and the σ_{ini} is set as the L^2 norm of the average of the training samples in the input space. Lastly, σ_{step} is determined to be 2. For ECOC-SVM, we use Sparse ECOC models with Hamming Decoding. We initialize the codebooks (coding matrix) with OVA class binarization and generate randomly rest of the codebooks each of whose row (codeword) can have

Table 1: Performance values for each method when window number is 400. (Irrelevant HR = Irrelevant class avg. hit rate (recall), Relevant HR = Relevant classes avg. hit rate).

Method	Precision	Accuracy	F_1 score	Irrelevant HR	Relevant HR
OVA	0.9293	0.9433	0.9176	0.9335	**0.9414**
AdaBoost	**0.9351**	**0.9520**	**0.9261**	**0.9962**	0.9144
Sparse ECOC	0.9209	0.9347	0.9062	0.9089	0.9176

zero elements with 0.3 probability. $15 \times \log(N_c)$ binary SVMs are trained. Moreover, a fresh codebook is generated with the same settings in each run.

Due to the variations in the results, we run each method several times using the train and test data. The final performance of each algorithm on the data set is the average of the results over all runs. The decision procedure of each method is as follows: If the predicted label of a test image is "-1" for all binary classifiers, then the test image class is assigned to be "0" class which is also called as *irrelevant* class. If only one of the OVA classifiers returns the label "+1", the class label of the test image is determined to be the class holding that OVA classifier. Lastly, If more than one of the OVA classifiers return "+1", then the class satisfying the highest probability estimate or smallest distance (for ECOC-SVM) is chosen as classification decision.

The performance evaluation of the proposed algorithm on 3 different SVM based approaches depends on the following important criteria: precision, recall, accuracy, F_1 score, irrelevant class hit rate (recall) and relevant classes hit rate. The macro-weighted average precision, recall, accuracy and F_1 score values are calculated as in (Sokolova and Lapalme, 2009). The irrelevant class hit rate is calculated by dividing the number of *irrelevant* images which are correctly classified by the total number of *irrelevant* image in test data using the aforementioned decision procedure. The relevant classes hit rate is determined as in the same way when assuming that there is not any *irrelevant* class and as opposed to aforementioned decision procedure, the class decision for *relevant* images is made only according to the highest probability estimate or smallest distance.

The performance values for each method when randomly generated window number is fixed to 400 are shown in Table 1. The best results are highlighted with bold fonts. Results in Table 1 demonstrates that AdaBoost-SVM method outperforms others on our own data set in terms of all criteria except for relevant classes hit rate. This method achieves a classification accuracy of 95.2% and 99.6% irrelevant hit rate, which almost implies perfect detection of irrelevant images. Sparse ECOC-SVM based classification method exhibits the worst performance among all 3 classification approaches with F_1 score = 0.906. One possible reason of this is that the performance of ECOC method is drastically affected by the association between class label and its codeword representation (Crammer and Singer, 2000). Generating random code matrix with fixed settings can create sub-optimal codeword representations that lead to the lowest performance. Figure 2 shows the average recall values of all runs versus the number of randomly selected windows considered as in the range of $[50, 600]$. Generally, the recall value has an increasing trend with the number of window due to having more information on visual words' statistics. The best value is achieved by the case when the window number is 400 and AdaBoost performs the best with 92.8 % recall rate. Next, the McNemar's statistical test (Li et al., 2008) is employed to determine the significance of the results presented in Table 1. Table 2 shows the McNemar's statistical test results for each method pair. The results are obtained by averaging the test statistics among all runs for each method pair. The test results illustrate that the performance of AdaBoost significantly differs from that of other two methods on this data set ($\{7.46, 21.69\} > 3.8414$). The underlying reason for this is the Boosting mechanism that forces several SVM component classifiers on imbalanced data sets to focus on the misclassified samples from the minority class, and to prevent them from being wrongly classified (Li et al., 2008).

Our results in Table 1 indicates that none of the three methods can achieve the best performance in terms of Irrelevant HR and Relevant HR at the same time. A good future direction of this work can be using an another strong classifier such as Random Decision Forest (RDF) (Yao et al., 2011) combined with SVM for better prediction of both *irrelevant* and *relevant* classes.

Table 2: McNemar's statistics between all method pairs when window number is fixed to 400.

Method Pairs	McNemar's statistic (χ^2)
OVA - AdaBoost	7.46
AdaBoost - Sparse ECOC	21.69
OVA - Sparse ECOC	21.02

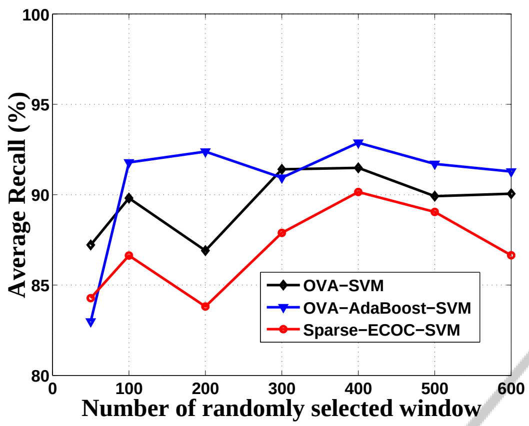

Figure 2: Average Recall values versus the number of randomly selected window for each method.

4 CONCLUSION

In this paper, we have proposed a method for the classification and retrieval of business form type document images. In our method, we incorporate BoVW model using a set of features based on structural variations in local image patches and present an approach to learn the visual words' histogram at layout level. Using a real-world bank data, we perform the analysis of different multiclass classification strategies and ensemble classifiers (Boosting) method with SVM as a base learner. Although initial results in this study seem to be promising, we believe that the proposed document image classification approach should also be investigated on real benchmark datasets. Furthermore, the effectiveness of the proposed local feature descriptors in this work should also be compared with that of the existing descriptors in literature, e.g., SIFT and SURF. Both of these issues remain as a future work to validate the robustness of the proposed approach.

ACKNOWLEDGEMENTS

This work was partially supported by the Scientific and Technological Research Council of Turkey under Grant 3120918 and by Yapi Kredi Bank under Grant 62609.

REFERENCES

Allwein, E., Schapire, R., and Singer, Y. (2001). Reducing multiclass to binary: A unifying approach for margin classifiers. *J. Mach. Learn. Res.*, 1:113–141.

Bagheri, M., Montezar, G., and Escalera, S. (2012). Error correcting output codes for multiclass classification: Application to two image vision problems. In *CSI Internatioanal Symposium on Artificial Intelligence and Signal Processing (AISP)*, pages 508–513.

Crammer, K. and Singer, Y. (2000). On the learnability and design of output codes for multiclass problems. In *Proceedings of the Thirteenth Annual Conference on Computational Learning Theory*, pages 35–46.

Csurka, G., Dance, C., Fan, F., Willamowski, F., and Bray, C. (2004). Visual categorization with bags of keypoints. In *Workshop on Statistical Learning in Computer Vision, ECCV*, pages 1–22.

Dietterich, T. and Bakiri, G. (1995). Solving multiclass learning problems via error-correcting output codes. *J. Artif. Int. Res.*, 2(1):263–286.

Fan, K., Wang, Y., and Chang, M. (2001). Form document identification using line structure based features. In *Proc.of the 6th Int. Conf. on Document Anal. & Recognition*, pages 704–708.

Hu, J., Kashi, R., and Wilfong, G. (1999). Document image layout comparision and classification. In *Proc. of the 6th Int. Conf. on Document Anal. & Recognition*, pages 285–288.

Kumar, J. and Doermann, D. (2013). Unsupervised classification of structurally similar document images. In *Proc. of the 12th Int. Conf. on Document Anal. & Recognition*, pages 1225–1229.

Kumar, J., Ye, P., and Doermann, D. (2012). Learning Document Structure for Retrieval and Classification. In *International Conference on Pattern Recognition (ICPR 2012)*, pages 1558–1561.

Lazebnik, S., Schmid, C., and Ponse, J. (2006). Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *Computer Vision and Pattern Recognition, IEEE Conf*, pages 2169–2178.

Li, X., Wang, L., and Sung, E. (2008). Adaboost with svm-based component classifiers. *Eng. Appl. Artif. Intell.*, 21(5):785–795.

Smith, D. and Harvey, R. (2011). Document retrieval using sift image features. 17(1):3–15.

Sokolova, M. and Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Inf. Process. Manage.*, 45(4):427–437.

Vapnik, V. (1998). *Statictal Learning Theory*. John Wiley and Sons, Inc.,New York.

Winn, J., Criminisi, A., and Minka, T. (2005). Object categorization by learned universal visual dictionary. In *ICCV*, pages 1800–1807.

Yang, Y. and Newsam, S. (2011). Spatial pyramid co-occurrence for image classification. In *Computer Vision (ICCV),International Conference on*, pages 1465–1472.

Yao, B., Khaslo, A., and Fei-Fei, L. (2011). Combining randomization and discrimination for fine-grained image categorization. In *Proc. CVPR*.

Zheng, Z., Wu, X., and Srihari, R. (2004). Feature selection for text categorization on imbalanced data. *SIGKDD Explor. Newsl.*, 6(1):80–89.