CrossMark

SPECIAL ISSUE PAPER

Two-stage approach to extracting visual objects from paper documents

Paweł Forczmański[1] · Andrzej Markiewicz[1]

Abstract In the paper we present an approach to the automatic detection and identification of important elements in paper documents. This includes stamps, logos, printed text blocks, signatures and tables. Presented approach consists of two stages. The first one includes object detection by means of AdaBoost cascade of weak classifiers and Haarlike features. Resulting image blocks are, at the second stage, subjected to verification based on selected features calculated from recently proposed low-level descriptors combined with certain classifiers representing current machine-learning approaches. The training phase, for both stages, uses bootstrapping, i.e., integrative process, aiming at increasing the accuracy. Experiments performed on large set of digitized paper documents showed that adopted strategy is useful and efficient.

Keywords Document segmentation · Object detection · Classifiers cascade · Visual descriptors · Low-level features

1 Introduction

From the historical point of view, paper document has been one of the basic means of human communication across ages.

✉ Paweł Forczmański
  pforczmanski@wi.zut.edu.pl

  Andrzej Markiewicz
  sowizz@gmail.com

[1] Faculty of Computer Science and Information Technology, West Pomeranian University of Technology, Szczecin, Żołnierska Str. 52, 71–210 Szczecin, Poland

Although the information in such documents is represented in different languages, structures and forms, they often contain common elements such as stamps, signatures, tables, logos, blocks of text and background. It can be seen that in order to prevent document accumulation, most of valuable pieces are digitally scanned and kept as digital copies. Storing data this way makes process of document organizing, accessing and exchange easier but, even then without a managing system it is difficult to keep things in order. In the paper we present an approach to extract characteristic visual objects from paper document. According to [1] such an approach that is able to recognize digitized paper document may be used to transform it into hierarchical representation in terms of structure and content, which would allow for an easier exchange, editing, browsing, indexing, filling and retrieval.

Our algorithm can be a part of a document managing system, whose main purpose is to determine parts of the document that should be processed further (e.g., text [2]) or to be a subject of enhancement and denoising (e.g., graphics, pictures, charts [3]). It could be an integral part of any content-based image retrieval system, or simply a filter that would select only documents containing specific elements [4], segregate them in terms of importance (colored documents containing stamps and signatures are more valuable than monochromatic ones, which suggest a copy [5,10]), etc. Presented approach is document-type independent; hence, it can be applied to any formal documents, diploma, newspapers, postcards, envelopes, bank checks, etc.

The paper is organized as follows: first we review related works and point out their characteristic features; then, we demonstrate both stages of the algorithm and finally, we present selected experimental results. We conclude the paper with an in-depth discussion.

2 Previous works

Literature survey indicates that the problem examined in this paper has been a subject of study for about three decades (a Google Scholar search reveals that the first paper containing phrase "page segmentation" dates back to 1985). The first, extensive survey of page segmentation and zone classification methods, which are the closest problems, has been done by Okun et al. [6] and covers papers from 1990 to 1999. In recent years, many more ideas have been further developed. Hence, in the following sections global (multi-class element detection and classification) and individual (class-specific detection and classification) approaches are discussed, as the most popular. We provide also a short review of other two-stage approaches as a general idea of computer vision.

2.1 Global approach

According to Okun et al. [6] so-called global approaches can be divided into three categories of methods: bottom-up, top-down and heuristic. Top-down methods can be useful when it comes to documents of initially known structure. Whole document constitutes as an input to top-down algorithms. It is then decomposed into smaller elements such as blocks and lines of text, single words and characters. Bottom-up strategy starts with a pixel-level analysis, and then pixels with common properties are grouped into bigger structures. Bottom-up techniques show their advantages when dealing with documents of various structure, but due to their complexity are often slower. Heuristic procedures attempt to combine robustness of top-down approaches and accuracy of bottom-up methods.

Connected component analysis is the most popular approach among bottom-up methods. Small groups of pixels are aggregated into bigger regions based on their proximity, localization and size. This process is accompaniment by smearing, nearest neighbor search and Voronoi diagram techniques for component grouping. Described algorithms are quite robust to skew, but depending on selected measures the processing cost may vary [6].

Bottom-up strategy is shown in [7], where documents are segmented into three classes (background, graphics and text). A sliding window technique is used to segment input image into blocks. Each block is subjected to feature extraction stage. After an extensive analysis, Sauvola et al. [7] formulated a number of rules that act as a classifier (extending the rule set increases the number of classes). Blocks of the same label are grouped, and final bounding box is defined in iterative masking procedure. Reported accuracy of text detection stands at high 99 %. Unfortunately, the results for other classes were not provided. Very similar approach is presented in [8], but it uses different set of features that are calculated from gray-level co-occurrence matrix (GLCM),

as well as k-means algorithm for grouping. Mean accuracy equals to 94 %.

In the same survey, a list of top-down strategies was provided. Most of them rely on run-length analysis performed on binarized, skew-corrected documents. As an example vertical and horizontal histogram (of run-length) profiles are examined in terms of valley occurrence, which represents white space between blocks. Other solutions include usage of Gaussian pyramid in combination with low-level features or Gabor-filtered pixel clustering.

Heuristic methods combine bottom-up and top-down strategies. Usage of XY-cuts algorithm for joining components of the same label, which were obtained through classification performed on run-length matrix statistics, is a perfect example of such combination. Another approach makes use of quad-tree adaptive split-and-merge operations [6] to group or divide regions of high and low homogeneity accordingly. An analysis of fractal signature value, which is lower for background than other elements, proves its usefulness while processing documents of high complexity.

When we consider zone classification as a separate issue, it will allow us to put more focus onto multi-class discrimination problem. Keysers et al. [3] proposed a discrimination into eight different classes. The paper provides a comparative analysis of commonly used features. Among them, Tamura's histogram achieved the highest accuracy, but due to its computation complexity it was discarded in favor to less complex feature vectors. Reported error rate is equal to 2.1 %, but 72.7 % of logos and 31.4 % of tables were misclassified. Wang et al. [1] proposed 69-element feature vector, which was reduced to 25 elements during feature selection stage, which allowed to achieve mean accuracy of 98.45 %; however 84.64 % of logos and 72.73 % of "other" elements were misclassified.

2.2 Individual approach

Individual approach focuses on single class detection and recognition. It is based on classification of characteristic features, often in a scheme "one versus all". In our previous works [5,9] a similar problem of stamp detection and recognition was described in detail. It applies Hough line and circle transforms, color segmentation and heuristic techniques. As it was stated in above-mentioned literature survey, logo detection is a very similar problem and can be solved with a little tweak to our previously presented solution [5]. Other authors propose to use key-point analyzing algorithms like Scale-Invariant Feature Transform (SIFT), Speeded-Up Robust Features (SURF) and Features from Accelerated Segment Test (FAST) or Angular Radial Transform (ART). Two-step approaches similar to methods described in previous subsection are also highly popular.

Detection of text blocks can be realized by means of statistical analysis [11], edge extraction [12], texture analysis [12,13]. Other authors made use of stroke filters [14–16], cosine transform [17] and LBP algorithm [18].

It should be noted that the intraclass variance of table objects is a huge problem, since they can be very complex. Typical table usually consists of a header and cells forming rows and columns. The number of cells, rows and columns depends on the volume of information contained. Moreover, font, ruling and background can be styled differently. In [19] Hu et al. focused on different kinds of mistakes that could be made during table detection. They also made major assumption that the input document contains only one column of text with easily separable, non-overlapping lines [19]. Sameer et al. [20] proposed a solution based on line detection algorithm. Although their aim was to reconstruct tables, information on outermost line intersections could be used to determine table coordinates as well.

Signature and autograph detection methods may be derived from handwriting detection algorithms, but direct application of those methods is hampered by high intraclass variance caused by individual characteristic style of signatures [21]. When it comes to signatures recognition, much more effort was put into biometric aspects such as recognition carried on beforehand, manually extracted images of signatures. Zhu et al. [21] proposed an algorithm consisting of extensive pre-processing, multi-scale signature saliency measure calculation for each connected component and area merging based on proximity and curvilinear constraints. High accuracy (92.8 %) was achieved on popular *Tobacco-800* database.

Keypoint-based algorithms are also popular in terms of signature segmentation. In [22] SUFR algorithm was used to determine keypoint location on images containing results of connected component analysis performed on image with erased text (only signature is visible) and with erased signature. For each keypoint a feature vector is extracted and stored in appropriate database. Components of query document are labeled according to the closest example from both databases. Text tagged component is erased; thus, a segmented signature is revealed. Connected component analysis is crucial part of the solution presented in [23]. The paper provides a comparative analysis of HOG, SIFT, gradient-based features, Local Ternary Patterns (LTP) and global low-level features. Classification is performed by SVM classifier. Experiments performed on *Tobacco-800* database proved that the set containing gradient and low-level features was the best, achieving 95% accuracy.

Since in this paper only a selection of the most interesting methods was described, for a broad and recent literature survey on page segmentation and zone classification a reader is directed to the paper mentioned in the beginning [1].

2.3 Two-stage processing concept

We apply a two-stage approach to the page segmentation. This concept is definitely not novel in the computer vision field; however, in this particular task is rarely used. Similar ideas have been applied mostly to the problems of object detection, extraction and classification in other classes of digital images [24]. In most of them, the idea comes from the assumption that the first processing stage performs a rough detection of objects of interest, while the second one applies more precise means to improve the identification accuracy [25]. In many papers, the two-stage approach is related to the integration of features (e.g., appearance and spatio-temporal HOGs [26], difference-of-Gaussians and accumulated gradient projection vector [27], entropy of local histograms and heuristic features [28], edge information and SIFT features [29]), combining classifiers (e.g., SVM and random sample consensus—RANSAC [30], two stages of mean-shift clustering [31]), mixed approaches (e.g., Hough transform joined with DBSCAN clustering [32], edge map and SVM [33], HOG and SVM [34], two variants of snakes [35], particle swarm optimization and fuzzy classifier [36]).

The analysis of the literature shows that most of the algorithms often use image pre-processing techniques (e.g., document rectification), deal with restricted forms of analyzed documents (e.g., to checks) and employ sophisticated features together with multi-tier approaches. The other observation is that there is hardly any method aimed at the detection of all possible classes of visual objects in paper documents. It may be caused by non-trivial nature of the problem and different characteristics of analyzed graphical elements.

In the proposed approach, we do not apply any pre-processing and employ very efficient AdaBoost cascade which is implemented using integral image, hence giving very high processing speed. It should be stressed that we analyze probably most of all possible object types that can be found in documents, which has got no significant representation in the literature.

3 Algorithm description

In our approach we adopted an assumption that a successful extraction of visual objects from a paper document can be performed using a sequence of rather simple means. Hence, the developed algorithm consists of two subsequent stages. The first one is a rough detection of candidates, while the second one is a verification of found objects. The first stage is based on fast and simple approach, namely AdaBoost cascade of classifiers (employing Haar-like features). Since it results in significantly high number of false positives, it is supported with a verification stage using an additional classification employing a set of more complex features.
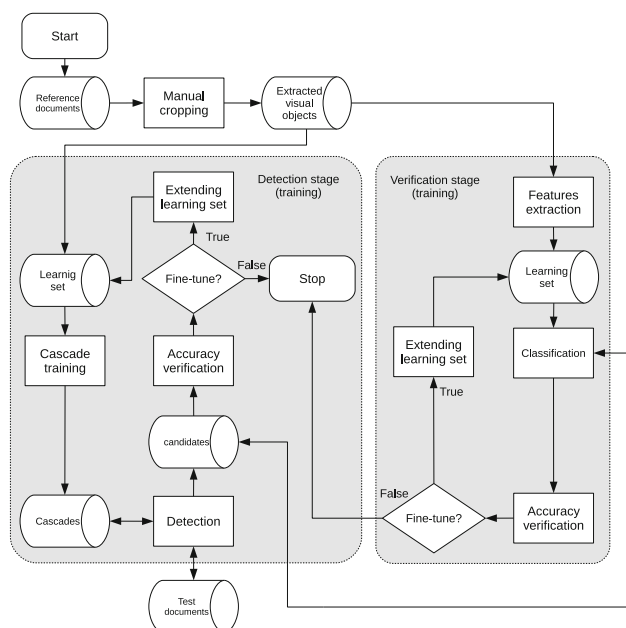
Fig. 1 Scheme of processing at the learning stage

Fig. 2 Exemplary objects belonging to the following classes (*in rows*): logo, stamp, signature, text, table, background

The training of the algorithm (see Fig. 1) in terms of detection and verification employs working in a iterative manner, which yields improved accuracy, depending on the quality and volume of learning sets. As it can be seen from Fig. 1 the reference documents dataset is subjected to manual cropping of interesting visual objects. This is an initialization of detector and verifier. Then, in each step (either detection or verification) the training involves fine-tuning and extending the learning sets. After that, the algorithm stops. In each iteration the learning set is being extended based on the results of accuracy verification.

The detector accuracy is evaluated on the set of testing documents, while the verifier is tested on objects extracted by cascade detector.

3.1 Cascade training and detection

Candidates detection is performed by AdaBoost-based cascades of weak classifiers [37,38]. At the training stage we learned five individual cascades for specific types of objects, namely: stamps, logos/ornaments, texts, tables and signatures. Exemplary objects are presented in Fig. 2. Background blocks, being an additional class, were taken further as negative examples for training other cascades. The detection was performed using a sliding window of 24×24 elements on a pyramid of scales where in each iteration we downscaled an input image by 10 %. Such size and downscale step are a compromise between complexity, memory overhead and discriminative properties.

The training procedure is performed iteratively with bootstrapping. The first, preliminary training, is to initialize the classifier. For this stage we used manually selected positive and negative samples for each class, marked in images collected from Internet and from *SigComp2009* [39]. The number of objects was limited in order to lower the processing time, having in mind the assumption, that after this iteration, positive and negative samples will be determined automatically.

Since the selection may by imperfect, in order to increase the detection accuracy we performed second iteration, in which the learning database was extended with objects resulted from previous iteration. We call it fine-tuning the detector (see Fig. 1). The positive results were added to the positive samples collection, while the negative to negative ones, respectively. It is a general rule that all samples from all classes except the selected one are put into negative part. The numbers of objects per class (in two iterations) are presented in Table 1. The class "background" was added in order to accumulate samples that were classified as other objects in the preliminary investigations. In the second iteration we removed background class since it gave very ambiguous results and it seems that detecting background using AdaBoost is not very accurate.

The effect of such fine-tuning is a removal of many false detections while retaining positive ones. Two examples of such situations are presented in Fig. 3. The first row presents the results of stamp detection after the first and the second iterations of training. The same applies to the second row in the same figure, yet it shows the results of signature detection. In both cases, the number of false detections has been reduced (however, not all of them have been eliminated). It

Table 1 Number of samples used at the cascade training stage

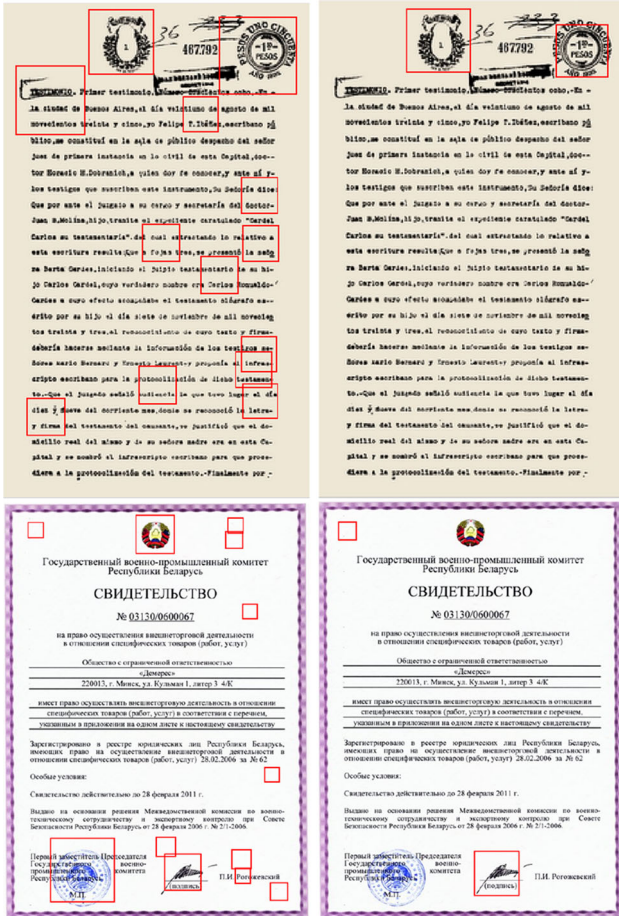| Class	1st iteration		2nd iteration	
	Positive	Negative	Positive	Negative
Stamps	150	740	324	2808
Logos	150	740	594	2538
Texts	150	740	707	2425
Signatures	150	740	224	2908
Tables	140	750	1283	1849
Background	150	740	n/a	n/a

Fig. 3 The effect of fine-tuning the detector (for stamps, in the *first row*, and signatures, in the *second row*, respectively)

is possible that repeating above-presented stage again will increase the quality of a learning set further. We stopped at two iterations as a compromise between accuracy and computational overhead.

3.2 Verification stage

Detected candidates are verified using a set of low-level features. The initial learning set, upon which reference features were calculated, consists of manually extracted 219 logos, 452 text blocks, 251 signatures, 1590 stamps, 140 tables and 719 background areas. As in case of detection, background blocks are used as negative examples and we do not verify background detection accuracy. After the initial investigations and the analysis of confusion matrices, in the second iteration of verification, we extended the learning set using extra 60 tables, 120 signatures and 50 text areas. Logotype and stamp classes together are quite numerous, and since their verification accuracy was acceptable, they were not extended.

It is a partial solution to the main observed problem; namely, many true-positive samples in signature and table classes were misclassified during the verification.

During our studies we selected eight feature sets, representing different approaches to low-level image description. They are presented in the following sections. Most of them (except binary version of LBP—LBPB) work on single-channel intensity images and do not relay on color information, which is an advantage.

3.2.1 First-order statistics (FOS)

We propose to use low-dimensional FOS as a base for further comparisons. Employed feature vector consists of six, direct, low-level attributes calculated from histogram of pixel intensities. These features are: mean pixel intensity, second (variance), third (skewness), fourth (kurtosis) central moment and entropy. They provide information about global characteristic of input image. A visualization of averaged FOS vectors over the whole learning database is presented in Fig. 4.
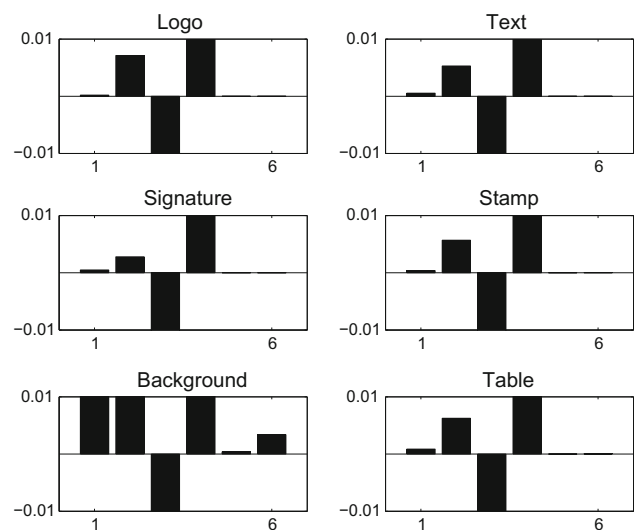

Fig. 4 Mean values of feature vectors (FOS) calculated for all classes in the learning set
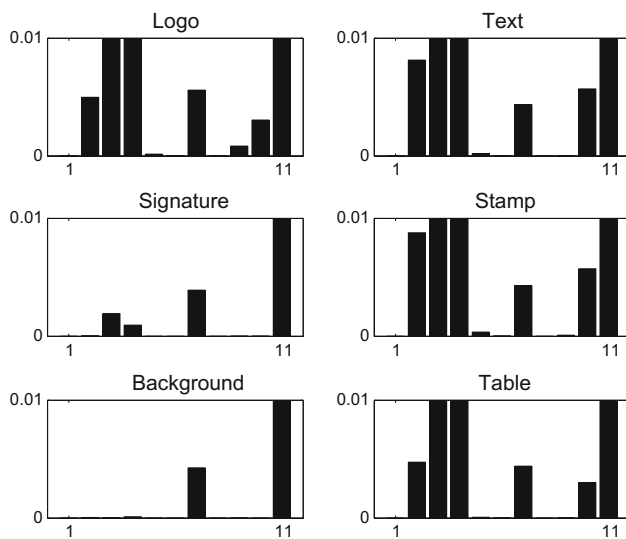
Fig. 5 Mean values of feature vectors (GLRLS) calculated for all classes in the learning set

Fig. 6 Mean values of feature vectors (HS) calculated for all classes in the learning set

3.2.2 Gray-level run-length statistics (GLRLS)

This feature vector consists of eleven attributes calculated from run-length matrix: short-run emphasis, long-run emphasis, gray-level non-uniformity, run-length non-uniformity, run-length non-uniformity, run percentage, low gray-level run emphasis, high gray-level run emphasis, short-run low gray-level emphasis, short-run high gray-level emphasis, long-run low gray-level emphasis, long-run high gray-level emphasis. Those features provide information about texture coarseness and/or fineness. Algorithm for GLRLM matrix calculation and a respective equations are presented in [40–42]. A visualization of averaged GLRLS vectors over the whole learning database is presented in Fig. 5.

3.2.3 Haralick's statistics (HS)

Well-known Haralick's properties are created from a set of 22 features calculated from gray-level co-occurrence matrix. A list of features used in our approach consists of: autocorrelation, contrast, correlation, cluster shade, cluster prominence, dissimilarity, energy, entropy, homogeneity, maximum probability, sum of squares: variance, sum average, sum variance, sum entropy, difference variance, difference entropy, information measures of correlation, inverse difference, inverse difference normalized, inverse difference moment normalized. Appropriate algorithms are available in [43–45]. A visualization of averaged HS vectors over the whole learning database is presented in Fig. 6.
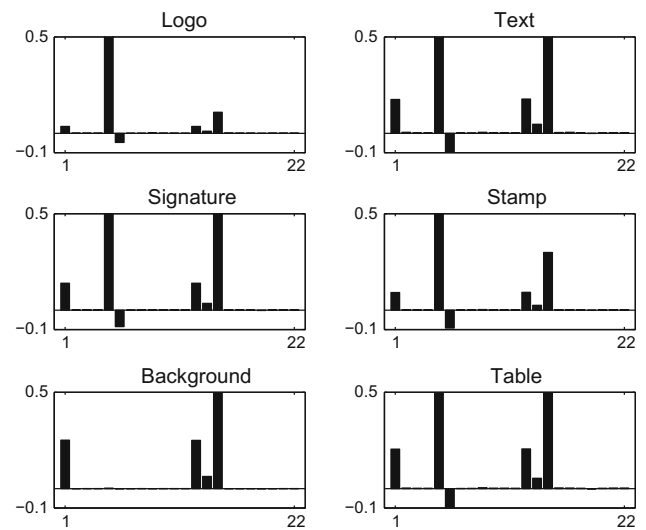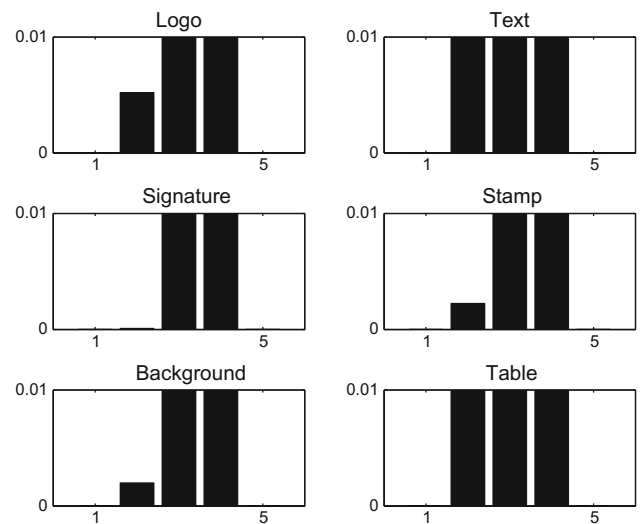

Fig. 7 Mean values of feature vectors (NGLDS) calculated for all classes in the learning set

3.2.4 Neighboring gray-level dependence statistics (NGLDS)

A very low-dimensional vector employing neighboring gray-level dependence statistics contains five values derived from NGLDM matrix, namely small number emphasis, large number emphasis, number non-uniformity, second moment and entropy. Element and their value distribution inside NGLDM matrix provide information about the level of texture coarseness. Algorithm for matrix calculations and respective equations are presented in [46]. A visualization of averaged NGLDS vectors over the whole learning database is presented in Fig. 7.
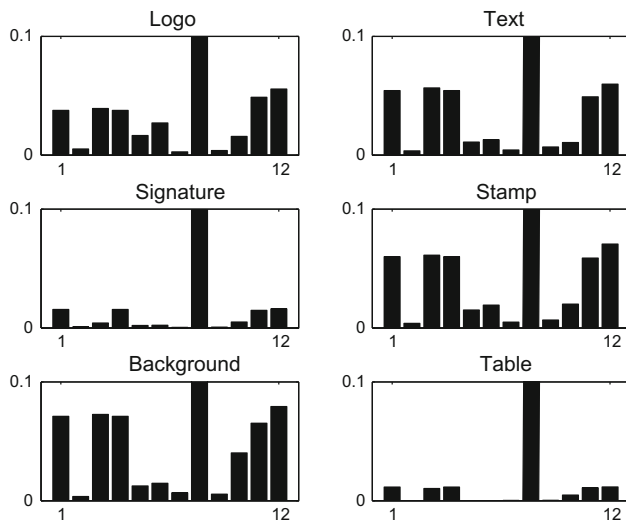
Fig. 8 Mean values of feature vectors (LLF) calculated for all classes in the learning set

Fig. 9 Mean values of feature vectors (HOG) calculated for all classes in the learning set

3.2.5 Low-level features (LLF)

So-called low-level features are a result of our previous research on stamp detection and recognition [5,9]. This approach shares common features with measures proposed by Haralic et al. Created feature vector contains eleven values, namely contrast, correlation, energy and homogeneity calculated in the same way as in case of GLCM matrix. Other attributes include: average pixel intensity, standard deviation of intensity, median intensity, contrast, mean intensity to contrast ratio, intensity of edges and mean intensity to edges intensity ratio. A visualization of averaged LLF vectors over the whole learning database is presented in Fig. 8.

3.2.6 Histograms of oriented gradients (HOG)

In order to investigate the state-of-the-art methods aimed at object detection we added to the comparison the histogram of oriented gradients approach. It is a method proposed by Dalal and Triggs in [47] and proved to be effective in human detection in digital images, but as it was mentioned in the paper, the algorithm is also capable of distinguishing between objects of different types. Feature vector of HOG descriptor is 256-element long. A visualization of averaged HOG vectors over the whole learning database is presented in Fig. 9.

3.2.7 Local binary patterns (LBP)

The last of the discussed features are local binary patterns. It was introduced in [48] as a universal, fine-scale texture descriptor [49]. Similarly to HOG the output vector consists of 256 elements. In our case, local binary patterns come in two different variants. The first one is calculated on
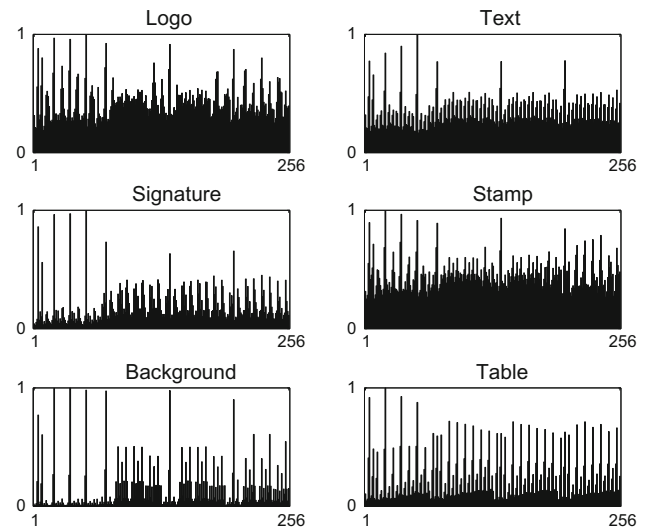

Fig. 10 Mean values of feature vectors (LBP) calculated for all classes in the learning set

monochromatic image, for the second binarized image was supplied (LBPB). A visualization of averaged LBP vectors over the whole learning database is presented in Fig. 10.

3.3 Dimensionality reduction

As it can be seen from Figs. 6, 7, 8, 9 and 10 many feature vectors have values that are common for all distinguished classes. It is probable that by eliminating them we can reduce the dimensionality of feature space while retaining recognition accuracy. That is why in the experiments we employed a substage of dimensionality reduction/feature selection, namely: principal component analysis (PCA) [50], linear discriminant analysis (LDA) [51], information gain (IG) [52] and least

Fig. 11 Eigenvalues contribution in PCA method applied to analyzed features

Fig. 12 Eigenvalues contribution in LDA method applied to analyzed features

Fig. 13 Distribution of information value in IG method applied to analyzed features
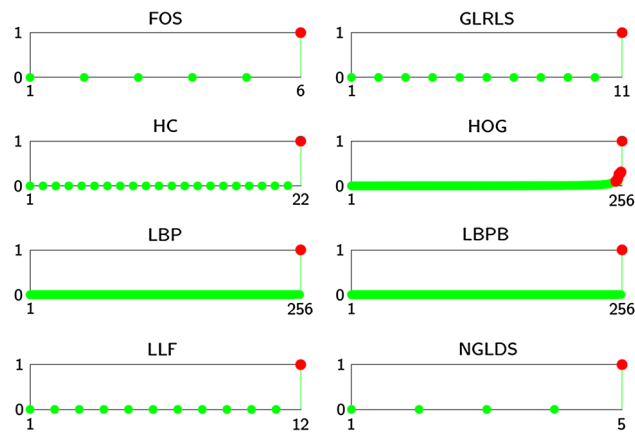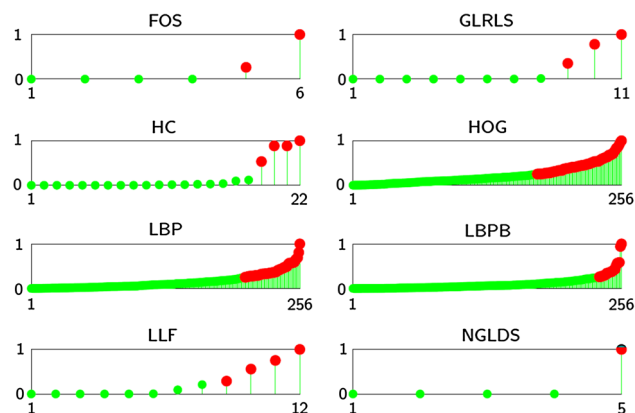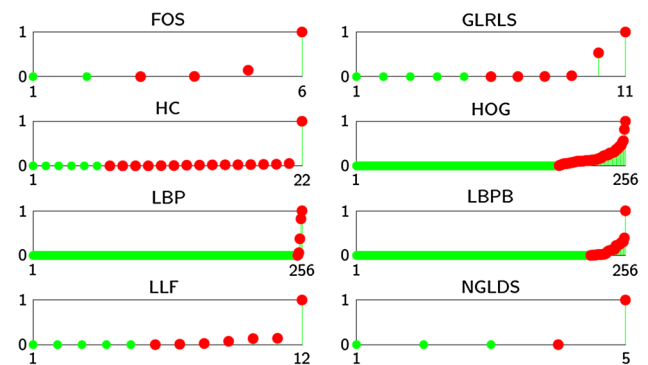

Fig. 14 Distribution of energy coefficients in LASSO method applied to analyzed features

absolute shrinkage and selection operator (LASSO) [53]. It is an improvement over a recent work [54].

In order to select the most discriminative information part in reduced feature spaces (after applying above algorithms) we performed an analysis of the distribution of energy (or importance levels) of reduced components. The visualizations of normalized components for each method of feature extraction are provided in Figs. 11, 12, 13 and 14. Selected components for further classification are marked.

As it can be seen, in some cases, only a fraction of calculated attributes were left (PCA), while in other cases the reduction algorithm selected more of them (less than half in case of LDA, more than half in case of IG and LASSO).

4 Experiments

The experiments were performed on our own database consisting of 719 digitized documents of various origin gathered, among other, from the Internet. It is the same database as one used in our previous work [5]. It contains scanned copies of

diverse diplomas, letters, invoices, postcards, envelopes and other official and unofficial documents written in different languages, with varying background and quality. The spatial resolution varies from 188×269 to 1366×620 pixels. Exemplary documents are shown in Fig. 15. First, an evaluation of the detection stage was performed. In the next step all generated examples were divided into two categories: positive and negative detections. This allowed us to calculate confusion matrices for each combination of classifier and feature set.

4.1 Detection stage

The decision whether the result should be considered positive or negative was made based on its bounding box area. Objects that are covered by approximately 75 % of resulting bounding box were classified positively. The results for both iterations are provided in Table 2. The mean detection accuracy after first iteration was equal to 54 % (with highest 80 % for text and lowest 14 % for signatures). Observed low accuracy is caused by high resemblance between classes, e.g., many logos were classified as stamps, large number of tables (which according to [6] should be considered as graph-

Fig. 15 Exemplary documents used in the experimental part

Table 2 Detection results

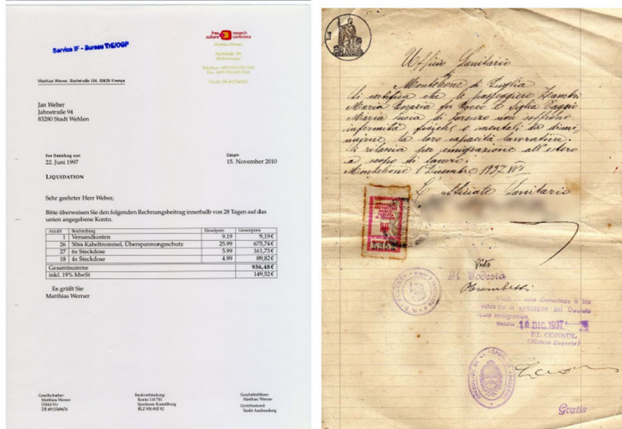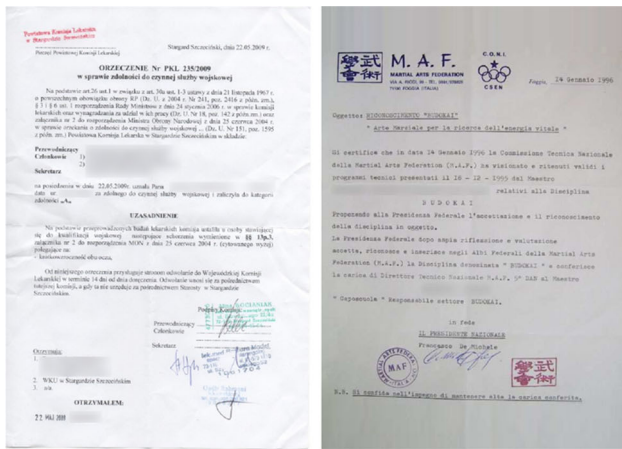| Class	1st iteration			2nd iteration		
	TP	FP	Acc. [%]	TP	FP	Acc. [%]
Stamps	174	281	42.54	235	187	60.04
Logos	444	394	65.29	236	75	84.01
Texts	557	919	80.38	136	84	91.63
Signatures	75	38	13.99	461	92	29.23
Tables	1133	1209	67.48	546	67	94.75

Fig. 16 Ambiguous objects: overlapped signatures, stamps and tables containing text

documents (uneven background and ink color, often overlapping with other elements). Those observations were taken into account when preparing data for the second training iteration.

Analyzing the results in Table 2 one can see a significant increase in detection accuracy using a learning set obtained by two iterations of training procedure. After that, there is a significantly lower number of false detections, yet also slightly lower number of positive detections. A clearly visible significant increase in signatures detection rate is still far from ideal. It is caused by the fact that in most cases signatures are overlapped with other elements, such as stamps, text and signature lines.

4.2 Verification stage

Experiments described below were aimed at determining a combination of a classifier and a feature vector (from the selection presented in Sect. 3.2) that gives the highest possible verification accuracy, depending on the quality of input samples. The selection of classifiers we investigated consists of: 1-nearest neighbor (1NN), Naïve Bayes (NBayes), binary decision tree (CTree), support vector machine (SVM), general linear model regression (GLM) and classification and regression trees (CART). There were also two iterations of processing provided for comparison. In the first iteration, the learning set was composed of initial features calculated for manually selected samples. The verification at this stage involved a selected pair of feature vector and classifier employed on objects returned in the first iteration of detection stage (see Sect. 3.1). The second iteration of verification process employed an extended learning set (see Sect. 3.2) and a feature vector/classifier fed with an output returned after the second iteration of detection.

The following figures (Figs.17 and 18) show examples of correct detections/verifications and failed ones, respectively.

ics) as printed text. The low accuracy for signatures comes from the lack of signatures in input documents; hence, we included the samples from *SigComp2009*, which are quite different in character. Examples of objects difficult to detect are presented in Fig. 16.

Lowest accuracy of signature detector results from different characteristics of examples used to train cascade (high resolution, bright and noise-free background, clear strokes, contrast ink) and the ones that are actually located on test

Fig. 17 Exemplary objects representing correct verification

Fig. 18 Exemplary objects representing failed verification

In each figure the objects are grouped in classes, as follows: stamps, logos, texts, signatures, tables.

As it can be seen from above figures, logotypes are often classified as stamps. Similar confusion applies to tables which are sometimes classified as text areas. What is more, the most problematic are tables which contain or are overlapped with graphical elements (e.g., logotypes or stamps).

In Tables 3, 4, 5, 6 and 7 verification accuracy for each class is presented (there are two columns of results for each classifier, each for subsequent iterations, respectively). The highest accuracy in the first iteration is underlined, while the highest accuracy in the second iteration is double underlined, respectively. Sometimes, more than one accuracy has the highest value; hence, more results are underlined.

4.3 Dimensionality reduction

In the experiments devoted to dimensionality reduction we employed k-nearest neighbor classifier and tenfold cross-validation. We tried to decide whether the reduction is necessary, since selected features (especially LBP and HOG) have rather high-dimensional feature space. As it was mentioned, we used PCA, LDA, IG and LASSO methods, since they are well-known, general purpose methods of high efficiency. The results of this experiment are presented in Table 8. Bold values indicate reflect the highest verification rate among methods involving dimensionality reduction.

As it can be seen, in most cases LASSO gives the highest accuracy; however, it is still lower than classification performed on a non-reduced features. Although the difference is not high, introducing these kinds of reductions may not be justified mainly because of additional computation overhead. The only exception is the case when we should conserve memory space, but nowadays it is not always crucial. The results of above experiment show that this substage may be omitted without loss of accuracy.

4.4 Discussion

As it was shown in Table 4 verification accuracy of logo-detecting cascade after second iteration had decreased. Large number of detected samples were misclassified as negative instead of positive. This is due to quite rigorous character of classifiers used. Taking into account the accuracy of the detection process (which is also done through classification) a cascade could be assigned a higher decision weight than the best pair of feature set and classifier used in verification to compensate for low precision in verification stage. Similar situation occurs in case of tables—again high detection accuracy is combined with low verification result. This is caused mostly by fuzzy boundary separating tables containing text from pure text class.

Average accuracies achieved at both stages of stamps and texts processing mean that equal decision weight could be assigned to both cascade and best combination of feature set and classifier. In both cases high precision of detection is coupled with high verification result. It is important to note that tables filled with text were classified as text. Otherwise, the results would be much lower.

Table 3 Stamps verification accuracy [%]

	1NN		NBayes		CTree		SVM		GLM		CART	
FOS	52.57	53.42	48.66	41.24	61.86	64.96	41.56	39.96	39.85	52.99	64.30	55.77
GLRLS	45.48	48.93	44.01	35.26	52.57	60.04	48.66	40.81	52.32	40.81	58.68	38.89
HS	58.19	60.68	55.75	50.21	74.57	63.25	79.46	71.37	<u>81.17</u>	71.79	73.35	35.47
HOG	73.84	<u>79.06</u>	73.84	69.44	60.39	63.46	65.53	51.07	69.44	66.45	69.19	41.45
LBP	55.01	61.54	48.17	36.97	55.99	56.84	73.84	43.38	73.35	38.46	54.77	39.53
LBPB	56.97	63.46	69.93	39.96	69.44	64.74	76.77	62.39	78.24	65.17	64.30	42.52
LLF	63.81	60.47	62.35	63.89	71.15	67.95	57.46	39.96	64.30	63.46	75.55	68.80
NGLDS	46.94	53.63	52.57	37.82	45.97	53.63	40.10	57.05	37.90	44.02	47.92	39.32

Table 4 Logos verification accuracy [%]

	1NN		NBayes		CTree		SVM		GLM		CART	
FOS	40.44	23.03	42.35	19.19	49.12	23.03	34.71	15.99	38.68	16.63	45.88	20.26
GLRLS	42.79	26.44	54.12	39.45	53.97	37.10	<u>67.94</u>	<u>51.60</u>	42.35	19.40	45.88	26.65
HS	42.35	18.98	59.12	32.41	54.12	33.48	61.18	39.23	49.12	23.67	44.26	20.26
HOG	53.82	30.28	55.88	39.23	51.32	35.61	60.74	43.07	58.53	49.12	47.21	31.34
LBP	40.00	18.76	47.65	17.70	37.50	21.54	46.76	22.17	43.24	21.32	38.53	18.76
LBPB	41.47	40.51	61.91	20.04	43.68	32.62	62.06	50.75	48.53	32.84	42.65	30.92
LLF	50.44	25.16	58.82	37.10	61.62	43.07	34.71	15.99	61.18	38.38	40.44	48.83
NGLDS	39.12	18.98	32.21	25.59	51.62	25.37	58.38	34.12	34.71	15.99	59.85	21.96

Table 5 Texts verification accuracy [%]

	1NN		NBayes		CTree		SVM		GLM		CART	
FOS	38.67	33.30	39.11	49.25	43.97	44.27	19.62	08.37	26.84	28.71	31.17	31.70
GLRLS	48.92	38.19	<u>85.71</u>	83.55	66.67	61.02	84.42	08.37	47.76	33.80	49.35	45.26
HS	39.83	36.39	69.70	74.88	67.82	55.83	59.31	86.64	64.79	67.80	38.67	45.36
HOG	71.28	79.56	84.70	<u>87.64</u>	54.69	66.00	55.41	87.04	77.63	85.34	61.76	58.23
LBP	62.91	60.42	77.34	69.69	67.53	68.79	80.52	83.25	72.87	79.36	46.75	47.16
LBPB	49.35	40.68	64.36	72.88	55.56	55.03	73.74	08.37	74.17	77.87	54.40	52.14
LLF	53.82	50.05	59.60	56.33	61.18	62.91	19.62	08.37	43.43	40.28	61.04	57.13
NGLDS	39.39	33.80	42.86	59.82	44.16	42.47	77.20	81.26	30.74	19.44	24.39	19.34

Table 6 Signatures verification accuracy [%]

	1NN		NBayes		CTree		SVM		GLM		CART	
FOS	84.51	69.23	85.45	52.31	85.63	64.62	77.43	70.77	80.97	67.69	84.33	65.38
GLRLS	85.45	71.54	85.82	70.77	<u>86.01</u>	70.77	85.82	57.69	79.85	66.15	85.63	66.92
HS	84.70	70.00	<u>86.01</u>	56.92	85.82	71.54	85.82	71.54	84.70	70.77	81.34	70.00
HOG	80.41	<u>80.00</u>	78.92	73.08	71.08	67.69	79.85	71.54	83.77	64.62	82.84	64.62
LBP	<u>86.01</u>	72.31	<u>86.01</u>	70.77	<u>86.01</u>	70.77	<u>86.01</u>	71.54	<u>86.01</u>	76.15	85.63	69.23
LBPB	<u>86.01</u>	72.31	80.22	63.85	<u>86.01</u>	69.23	<u>86.01</u>	75.38	<u>86.01</u>	75.38	85.82	72.31
LLF	83.96	70.00	85.63	63.08	85.63	70.00	<u>86.01</u>	70.77	82.46	56.92	76.31	66.15
NGLDS	<u>86.01</u>	71.54	<u>86.01</u>	70.77	<u>86.01</u>	70.77	<u>86.01</u>	70.77	<u>86.01</u>	70.77	85.07	72.31

Table 7 Tables verification accuracy [%]

	1NN		NBayes		CTree		SVM		GLM		CART	
FOS	32.70	05.02	31.98	05.09	32.28	05.17	31.63	05.09	32.28	05.25	32.76	05.64
GLRLS	23.47	06.27	31.51	05.02	32.34	05.09	32.04	04.86	32.28	05.09	33.53	05.09
HS	32.46	05.17	32.10	05.25	32.22	05.02	31.21	05.17	32.58	05.17	32.34	06.50
HOG	27.99	05.02	32.28	05.02	29.30	04.78	22.45	04.70	20.49	05.49	29.18	07.68
LBP	24.54	21.87	32.94	06.90	32.88	07.21	29.60	04.86	28.23	08.54	31.80	13.40
LBPB	47.41	33.07	41.16	11.29	32.58	12.77	27.34	04.62	32.04	05.49	41.27	19.91
LLF	58.13	54.78	<u>69.62</u>	<u>57.60</u>	60.21	54.39	32.52	05.25	55.39	50.63	51.58	40.20
NGLDS	27.04	21.94	18.76	12.07	25.25	08.86	23.05	12.07	22.57	05.72	27.34	06.82

Table 8 Verification rate comparison for different dimensionality reduction methods [%]

Reduct.	Features						
	FOS	GLRLS	HS	HOG	LBP	LLF	NGLDS
No red.	77.41	84.07	81.85	89.90	89.67	82.21	67.14
PCA	69.17	79.13	**83.82**	80.66	51.47	81.35	68.47
LDA	65.89	78.99	64.48	54.60	44.11	67.00	**69.31**
IG	69.50	74.11	77.05	86.12	81.04	75.74	63.40
LASSO	**74.80**	**82.24**	81.77	**88.65**	**82.07**	**82.13**	66.84

As it was noted, signature class causes most of the problems. Higher detection accuracy is only a result of much lower FP rate. This is caused by the extension of the learning set (both in training of cascade and at the verification stage). Further increase, especially in case of positive samples number, would be beneficial.

The analysis of presented verification results shows that all of discussed object classes should be considered separately. Unfortunately, it is impossible to point out a single pair of classifier/feature vector that wins in all cases. There seems to be no one rule that is behind above results.

In case of stamp class, the most accurate pair consists of GLM classifier and HC features set and a pair of 1NN classifier and HOG descriptor comes at second. Those pairs alternate between iterations. Analogous observations were made in case of the worst pair. In the first iteration, GLM classifier and NGLDS features were worst and NBayes + GLRS were second worst. Reverse relationship occurred in the second iteration. The average accuracy across all sets is equal to 60.17 and 53.3 % in first and second iterations, respectively. HS is the most accurate descriptor (average accuracy of 70.42 %) in the first iteration and HOG (with 61.82 % average accuracy) in the second. An accuracy of 63.51 % places CART classifier as the best in the first iteration, and 61.86 % places CTree classier at the top in the second iteration. Results for remaining classes were described in similar manner—first percentage value always corresponds to the result achieved in the first iteration and so on.

In both iterations of logo verification SVM classifier and GLRLS features set proved to be the best. There was no recurrence in case of the worst pair. Average accuracy is equal to 48.6 and 29.04 %. The highest average score was achieved by SVM classifier (53.31, 34.12 %) and HOG descriptor (54.58, 38.11 %).

Bayes-based classifiers, namely NBayes+GLRLS and NBayes+HOG, achieved the highest accuracies in the first and the second iteration of text verification process, respectively. Analogous switch in terms of the best and the second best as in case of stamp occurred. Overall accuracy stands at 55.52 and 52.99 %. The LBP and HOG descriptors proved to be the most accurate (67.99, 77.3 %). In both cases NBayes was selected as the best (65.42, 69.26 %).

The analysis of signature verification results is shown that GLM+LBP achieved high scores at both stages, only to be defeated by 1NN+HOG pair in the second iteration. Overall accuracy equals to 84.02 and 68.94 %. In both iterations the same feature set and classifier produced the highest scores: LBP (85.95, 71.8 %) and 1NN (84.63, 72.12 %).

Only in case of tables verification there is a significant domination of one classifier and feature set pair (NBayes+LLF) over all other combinations. Although the average accuracy is low (33.47 and 12.66 %), its value achieved by the best pair is satisfactory. NBayes classier paired with LLF feature set reached 69.62 and 54.17 %. Average classification accuracy of NBayes classifier is equal to 36.29 and 19.14 %, and for LLF features stands at 54.58, 43.81 % in the first and the second iterations, respectively.

4.5 Comparison with state-of-the-art methods

It is not easy to directly compare obtained results with other state-of-the-art methods, since the benchmark sets are very different. Moreover, the comparison with individual methods may not be justified because such methods employ class-specific approaches, which are tuned for particular object types. Hence, below, a not entirely meaningful comparison with certain, selected global approaches is provided. Taking into consideration average values, the detection accuracy in case of our algorithm is equal to 71.93 % and the verification accuracy (calculated for the best individual pairs) is equal to 78.48 %. When we exclude the most problematic class (in terms of detection), namely signatures, the detection accuracy rises to 82.61 % and the verification slightly drops to 76.59 %. It is because signatures are detected with a relatively low accuracy, yet their verification accuracy is quite high. In [3], the authors obtained an average detection accuracy equal to 81.84 %; however, when we consider only classes, that are similar to our case (however, without stamp class), the accuracy drops to 72.95 %. The main problem with that approach is a high number of misclassifications in case of tables. In our algorithm, tables are detected and verified with a very high accuracy. In [1] the mean accuracy for 9 classes is equal to 84.38 %. When we restrict the set in order to be similar to the one in our case (also without stamp class), it is equal to 89.11 %. The best result was obtained for printed text class, and again, the most problematic class is logotypes.

As it can be seen, our approach is comparable to the state-of-the-art approaches, while it features very intuitive processing flow and a significantly lower computational overhead. It also takes into consideration classes that are not analyzed in above-mentioned approaches, namely stamps and signatures. Having in mind increasing the learning datasets and introducing extra training iterations (at the detection stage), the accuracy may be even higher.

5 Summary

We have presented a novel approach to the extraction of visual objects from digitized paper documents. Its main contribution is a two-stage detection/verification idea based on iterative training and multiple features–classifiers pairs. As opposite to other known methods, the whole framework is common for various classes of objects. It also features classes that are not considered by other scientists in the global approaches, namely signatures and stamps. Performed extensive experiments showed that the whole idea proved to be valid. High accuracies achieved in in-depth analysis performed on large, real document set prove this fact further. Results from the second iteration (see Table 2) are particularly encouraging. Although there is a high similarity

between some classes and numerous challenging examples throughout image database (see Fig. 16), the detection is successful. The signatures class is an exception, and the lower accuracy of detection/verification can be put down to the poor representation across databases. Increasing the size of learning set for signatures detection with high degree of probability would boost results as shown in case of the first and the second iteration.

High accuracies for certain classes in particular could lead to dropping the verification stage as it is redundant if cascade looks as like what it really is—a classifier itself. However, as long as there is more than a few of misclassified samples the use of this stage is justified. If we decide to use the verification stage, it is important to examine each class separately, as shown in previous section. It is well illustrated in Table 7. While overall accuracy is really low, accuracy for LLF feature set is several times higher than in case of any other feature set. As it was shown, the dimensionality reduction substage is not necessary, since it does not improve the classification accuracy.

References

1. Wang, Y., Phillips, T.I., Haralick, M.R.: Document zone content classification and its performance evaluation. Pattern Recogn. **39**(1), 57–73 (2006)
2. Lech, P., Okarma, K.: Fast histogram based image binarization using the Monte Carlo threshold estimation. ICCVG'2014. LNCS vol. **8671**, pp. 382–390 (2014)
3. Keysers, D., Shafait, F., Breuel, M.T.: Document image zone classification - a simple high-performance approach. 2nd International Conference on Computer Vision Theory and Applications. pp. 44–51 (2007)
4. Marchewka, A., Pasela, A.: Extraction of data from Limnigraf chart images. Adv. Intell. Syst. Comput. **233**, 263–269 (2014)
5. Forczmański, P., Markiewicz, A.: Stamps detection and classification using simple features ensemble. Math. Probl. Eng. Article ID 367879 (2015)
6. Okun, O., Doermann, D., Pietikäinen, M.: Page Segmentation and Zone Classification: The State of the Art. Technical Report: LAMP-TR-036/CAR-TR-927/CS-TR-4079, University of Maryland, College Park (1999)
7. Sauvola, J., Pietikäinen, M.: Page Segmentation and Classification Using Fast Feature Extraction and Connectivity Analysis. ICDAR, 1995, Proceedings of 3rd International Conference on Document Analysis and Recognition, pp. 1127–1131 (1995)
8. Lin, M.-W., Tapamo, J.-R., Ndovie, B.: A texture-based method for document segmentation and classification. S. Afr. Comput. J. **36**, 49–56 (2006)
9. Forczmański, P., Markiewicz, A.: Low-Level Image Features for Stamps Detection and Classification. 8th International Conference

on Computer Recognition Systems (CORES), Advances in Intelligent Systems and Computing 226, pp. 383–392 (2013)

10. Forczmański, P., Frejlichowski, D.: Robust Stamps Detection and Classification by Means of General Shape Analysis. International Conference on Computer Vision and Graphics (ICCVG). LNCS vol. 6374, pp. 360–367 (2010)

11. Su, C., Haralick, M.R., Ihsin, T.P.: Extraction of text lines and text blocks on document images based on statistical modeling. Int. J. Imaging Syst. Technol. **7**(4), 343–356 (1996)

12. Pietikäinen, M., Okun, O.: Edge-based method for text detection from complex document images. Proceedings. Sixth International Conference on Document Analysis and Recognition, pp. 286–291 (2001)

13. Jain, A.K., Zhong, Y.: Page segmentation using texture analysis. Pattern Recogn. **29**(5), 743–770 (1996)

14. Jung, C., Liu, Q., Kim, J.: A stroke filter and its application to text localization. Pattern Recogn. Lett. **30**(2), 114–122 (2009)

15. Liu, Q., Jung, C., Kim, S., Moon, Y., Kim, J.: Stroke Filter for Text Localization in Video Images. IEEE International Conference on Image Processing, pp. 1473 – 1476 (2006)

16. Li, X., Wang, W., Jiang, S., Huang, Q., Gao, W.: Fast and effective text detection. 15th IEEE International Conference on Image Processing, pp. 969–972 (2008)

17. Zhong, Y., Zhang, H., Jain, A.K.: Automatic caption localization in compressed video. IEEE TPAMI **22**(4), 385–392 (2000)

18. Ojala, T., Pietikäinen, M., Mäenpää, T.: Gray scale and rotation invariant texture classification with local binary patterns. In Proceedings of the 6th European Conference on Computer Vision, pp. 404–420 (2000)

19. Hu, J., Kashi, R., Lopresti, D., Wilfong, G.: Evaluating the performance of table processing algorithms. Int. J. Doc. Anal. Recogn. **4**(3), 140–153 (2002)

20. Gatos, B., Danatsas, D., Pratikakis, I., Perantonis, S.J.: Automatic Table Detection in Document Images. Pattern Recogn. Data Min. LNCS **3686**, 609–618 (2005)

21. Zhu, G., Zheng, Y., Doermann, D., Jaeger, S.: Signature detection and matching for document image retrieval. IEEE TPAMI **31**(11), 2015–2031 (2009)

22. Ahmed, S., Malik, M.I., Liwicki, M., Dengel, A.: Signature segmentation from document images International Conference on Frontiers in Handwriting Recognition (ICFHR), pp. 425–429 (2012)

23. Cüceloğlu, İ., Oğul, H.: Detecting handwritten signatures in scanned documents. Proceedings of the 19th Computer Vision Winter Workshop, pp. 89–94 (2014)

24. Li, S.Z., Hornegger, J.: A two-stage probabilistic approach for object recognition. Computer Vision – ECCV'98, LNCS 1407, pp. 733–747 (1998)

25. Gerdes, R., Otterbach, R., Kammüller, R.: Fast and robust recognition and localization of 2-D objects. Mach. Vis. Appl. **8**(6), 365–374 (1995)

26. Mitsui, T, Fujiyoshi, H.: Object detection by joint features based on two-stage boosting. Computer Vision Workshops (ICCV Workshops), 2009 IEEE 12th International Conference on, pp. 1169–1176 (2009)

27. Chen, Y.-P., Yeh, T.-D.: A Method for Extraction and Recognition of Isolated License Plate Characters. International Journal of Computer Science and Information Security, Vol. 5, No.1 (2009)

28. Sikdar, A., Roy, P., Mukherjee, S., Das, M., Banerjee, S.: Two Stage Method for Bengali Text Extraction from Still Images Containing Text International Conference of Advanced Computer Science & Information Technology (ACSIT-2012), pp. 14 –15 (2012)

29. Jauregi, E., Lazkano, E., Sierra, B.: Object recognition using region detection and feature extraction. Proceedings of 10th International Conference Towards Autonomous Robotic Systems TAROS 2009, pp. 104–111 (2009)

30. Kuo, C.-H., Lee, J.-D.: A two-stage classifier using SVM and RANSAC for face recognition. TENCON IEEE Region 10 Conference, pp. 1–4 (2007)

31. Papić, V., Turić, H., Dujmić, H.: Two-stage segmentation for detection of suspicious objects in aerial and long- range surveillance applications. Proceedings of the 10th WSEAS International Conference on Automation & Information, pp. 152–156 (2009)

32. Niua, J., Lua, J., Xub, M., Lvb, P., Zhaoa, X.: Robust Lane Detection using Two-stage Feature Extraction with Curve Fitting. Pattern Recognition, in press, doi:10.1016/j.patcog.2015.12.010 (2015)

33. Anthimopoulos, M., Gatos, B., Pratikakis, I.: A two-stage scheme for text detection in video images. Image Vis. Comput. **28**, 1413–1426 (2010)

34. Han, F., Shan, Y., Cek, R., Sawhney, H.S., Kumar, R.: A two-stage approach to people and vehicle detection with HOG-based SVM. Performance Metrics for Intelligent Systems (PerMIS'06), 133–140 (2006)

35. Pribošek, J., Gregorčič, P., Diači, J.: Automating shockwave segmentation in low-contrast coherent shadowgraphy. Mach. Vis. Appl. **26**(4), 485–494 (2015)

36. Setayesh, M., Zhang, M., Johnston, M.: Feature Extraction and Detection of Simple Objects Using Particle Swarm Optimisation. Wellington Victoria University Technical Report Series no. 09-15 (2009)

37. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2001, pp. 511–518 (2001)

38. Burduk, R.: The AdaBoost Algorithm with the Imprecision Determine the Weights of the Observations, 6th Asian Conference Intelligent Information and Database Systems ACIIDS. LNCS **8398**, pp. 110–116 (2014)

39. Liwicki, M.: ICDAR 2009 Signature Verification Competition. http://www.iapr-tc11.org/mediawiki/index.php/ICDAR_2009_Signature_Verification_Competition_(SigComp2009). Accessed 24 Feb 2015 (2009)

40. Galloway, M.M.: Texture analysis using gray level run lengths. Comput. Graph. Image Process. **4**(2), 172–179 (1975)

41. Tang, X.: Texture information in run-length matrices. Image Process. IEEE Trans. **7**(11), 1602–1609 (1998)

42. Dasarathy, R.B., Holder, B.E.: Image characterizations based on joint gray-level run-length distributions. Pattern Recogn. Lett. **12**, 497–502 (1991)

43. Haralick, M.R., Shanmugam, K., Dinstein, I.: Textural features of image classification. IEEE Trans. Syst. Man Cybern. **SMC–3**(6), 610–621 (1973)

44. Soh, L., Tsatsoulis, C.: Texture analysis of SAR sea ice imagery using gray level co-occurrence matrices. IEEE Trans. Geosci. Remote Sens. **37**(2), 780–795 (1999)

45. Clausi, A.D.: An analysis of co-occurrence texture statistics as a function of grey level quantization. Can. J. Remote Sens. **28**(1), 45–62 (2002)

46. Siew, L.H., Hodgson, R.M., Wood, E.J.: Texture measures for carpet wear assessment. IEEE TPAMI **10**(1), 92–105 (1988)

47. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. International Conference on Computer Vision & Pattern Recognition, vol. **2**, pp. 886–893 (2005)

48. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE TPAMI **24**(7), 971–987 (2002)

49. Maturana, D., Mery, D., Soto, Á.: Face Recognition with Local Binary Patterns, Spatial Pyramid Histograms and Naive Bayes Nearest Neighbor Classification. Proceedings of the 2009 International Conference of the Chilean Computer Science Society, pp. 125–132 (2009)

50. Abdi, H., Williams, L.J.: Principal component analysis. Wiley Interdiscip. Rev Comput. Stat. **2**(4), 433–459 (2010)
51. McLachlan, G.J.: Discriminant Analysis and Statistical Pattern Recognition. Wiley Interscience, New York (2004)
52. Battiti, R.: Using mutual information for selecting features in supervised neural net learning. Neural Netw. IEEE Trans. **5**(4), 537–550 (1994)
53. Tibshirani, R., Regression shrinkage and selection via the lasso, J. R. Stat. Soc Ser. B Methodol. **58**(1), 267–288 (1996)
54. Markiewicz, A., Forczmański, P.: Detection and classification of interesting parts in scanned documents by means of adaBoost classification and low-level features verification. Comput. Anal. Images Patterns LNCS **9257**, 529–540 (2015)

Paweł Forczmański received MSc title (1998) and PhD title (2002) from Faculty of Computer Science and Information Technology, Technical University of Szczecin (at present West Pomeranian University of Technology, Szczecin). His scientific interests include computer vision algorithms, methods of dimensionality reduction and image classification in biometrical applications, content-based image retrieval and visual surveillance. He is the author of over 70 scientific articles, chapters in books and conference papers.

Andrzej Markiewicz received BSc title (2013) and MSc title (2015) from Faculty of Computer Science and Information Technology, West Pomeranian University of Technology, Szczecin. His scientific interests include image processing, pattern recognition and computer vision algorithms.